

Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна

РОЗРОБНИК БАЗ ДАНИХ

Теоретично-тестовий практикум

Харків – 2025

Рецензенти:

Ю. О. Скоб – доктор технічних наук, професор кафедри математичного моделювання та штучного інтелекту Національного аерокосмічного університету імені М. Є. Жуковського «Харківський авіаційний інститут»;

М. А. Мірошник – доктор технічних наук, професор, професор кафедри комп'ютерних систем та робототехніки Харківського національного університету імені В. Н. Каразіна.

*Затверджено до друку рішенням Науково-методичної ради
Харківського національного університету імені В. Н. Каразіна
(протокол 11 від 25 червня 2025 року)*

Ч-92 Розробник баз даних : теоретично-тестовий практикум / уклад. О. І. Чуб,
М. М. Хруслов. – Харків : ХНУ імені В. Н. Каразіна, 2025. – 96 с.

Видання «Розробник баз даних» створене для підготовки та перевірки знань фахівців, які працюють із базами даних. Особливістю цього навчального посібника є прив'язка змісту до європейської класифікації навичок, компетенцій, кваліфікацій та професій (ESCO), що є стандартом для опису професійної діяльності у країнах Європейського Союзу. У практикумі детально розглянуто професію «Розробник баз даних»: функціональні обов'язки, основні навички, компетенції та вимоги відповідно до міжнародних стандартів.

Тестова частина включає 325 питань із чотирма варіантами відповідей, із яких лише один є правильним. Такий формат забезпечує об'єктивну оцінку знань, охоплюючи як теоретичну підготовку, так і практичні навички. У кінці посібника наведено правильні відповіді для самоперевірки та рекомендовані джерела для подальшого вивчення. Посібник орієнтований на студентів, які проходять навчання за спеціальностями F7 «Комп'ютерна інженерія» та G7 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка».

УДК 004.65

© Харківський національний університет
імені В. Н. Каразіна
© Чуб О. І., Хруслов М. М., уклад., 2025

ЗМІСТ



Вступ.....	4
Розділ 1. Теоретична частина.....	6
1.1. Стандарт ESCO.....	6
1.2. Роль ESCO у сучасному цифровому ринку праці(виступу)	7
1.3. Еволюція та вдосконалення ESCO	8
1.4. Довідник для користувачів ESCOpedia.....	10
1.5. Компоненти ESCO	11
1.5.1. Професійний компонент.....	11
1.5.2. Компонент навичок, компетенцій та знань	12
1.6. Маркування цифрових навичок в ESCO.....	13
1.7. Характеристика та вимоги до професії «Розробник баз даних»	16
1.7.1. Компетенції професії «Розробник баз даних»	17
1.7.2. Розподіл навичок та компетенцій у професії «Розробник баз даних»	28
Висновки за Розділом I.....	30
Розділ II. Тестова частина.....	31
2.1. Формат і зміст тестування.....	31
2.2. Тестування компетенцій.....	33
Компетенція № 1. Застосовувати політику інформаційної безпеки	33
Компетенція № 2. Збалансовувати ресурси бази даних.....	38
Компетенція № 3. Збирати відгуки клієнтів про додатки.....	42
Компетенція № 4. Створювати моделі даних	46
Компетенція № 5. Оцінювати тривалість робіт	50
Компетенція № 6. Визначати вимоги замовника.....	55
Компетенція № 7. Перекладати технічні тексти.....	59
Компетенція № 8. Створювати резервні копії	64
Компетенція № 9. Звітувати про результати аналізу	69
Компетенція № 10. Тестувати ІКТ-запити	74
Компетенція № 11. Використовувати інтерфейс конкретного додатка	79
Компетенція № 12. Використовувати бази даних	82
Компетенція № 13. Писати документацію до баз даних.....	86
Компетенція 2.3. Відповіді на тести.....	90
Висновки за Розділом II.....	92
Рекомендовані джерела для підготовки.....	94

ВСТУП



Навчальний посібник «Бази даних. Теоретично-тестовий курс» створено для комплексної підготовки студентів, які в майбутньому будуть працювати з базами даними. Особливістю видання є інтеграція матеріалів із європейським стандартом European Skills, Competences, Qualifications and Occupations. ESCO забезпечує систему опису професій, навичок та кваліфікацій у межах країн Європейського Союзу, що робить посібник корисним у контексті орієнтації навчального процесу на міжнародні стандарти професійної діяльності.

Основні задачі, які ставили перед собою автори навчального посібника:

1) Адаптація офіційної англomовної інформації про стандарт ESCO до освітнього контексту для забезпечення доступності, зрозумілості та практичної цінності матеріалів у підготовці фахівців у сфері баз даних;

2) Формування у студентів практичних навичок, необхідних для ефективного виконання професійних обов'язків розробника баз даних, з акцентом на моделювання структури даних та забезпечення надійної роботи баз даних;

3) Надання структурованого навчального матеріалу, що охоплює як теоретичну частину, так і тестові завдання, для об'єктивної оцінки рівня знань і подальшого розвитку професійних компетенцій.

У першому розділі посібника розглядається теоретичне підґрунтя ESCO. Тут докладно описано ключові аспекти стандарту, його функціональність, структура та значення для цифрового ринку праці. Підкреслено роль ESCO в уніфікації професій, гармонізації освітніх програм і забезпеченні прозорості взаємодії між ринком праці та освітніми установами. Особливу увагу приділено професії «Розробник баз даних», її характеристикам, вимогам до компетенцій і навичок, а також значенню цієї професії у забезпеченні ефективного функціонування сучасних інформаційних систем. Включені також історичні аспекти розвитку ESCO, його впровадження у цифрові платформи та способи адаптації до динамічних змін на ринку праці.

Другий розділ посібника присвячено практичному застосуванню теоретичних знань, зокрема тестуванню компетенцій фахівців. Тестова частина включає 325 запитань із чотирма варіантами відповідей, із яких лише один є правильним. Завдання розподілені на 13 ключових тем, які відповідають компетенціям, визначеним у стандарті ESCO для професії «Розробник баз даних». Ці теми охоплюють усі аспекти професійної діяльності, починаючи

від інформаційної безпеки та моделювання даних і закінчуючи створенням звітності та документації до баз даних. Такий формат тестування забезпечує об'єктивну оцінку як теоретичних знань, так і практичних навичок, що є важливими для професійного розвитку.

Окрім теоретичних матеріалів і тестових завдань, у посібнику наведено список правильних відповідей, що дає змогу користувачам самостійно перевіряти свої результати. Також включено рекомендовані джерела для подальшого вивчення тем, що сприяє глибшому опануванню матеріалів. Це робить посібник ефективним інструментом для самостійного навчання та підготовки до професійної діяльності.

Посібник рекомендований для студентів спеціальностей галузі знань 12 «Інформаційні технології», а також широкого кола читачів. Зміст навчального видання сприяє розвитку як базових, так і спеціалізованих навичок, забезпечуючи інтеграцію знань у практичну діяльність.

РОЗДІЛ І

ТЕОРЕТИЧНА ЧАСТИНА



1.1. Стандарт ESCO

Стандарт ESCO (European Skills, Competences, Qualifications and Occupations) – це багатомовна європейська класифікація, що створена з метою гармонізації системи опису професій і навичок у межах Європейського Союзу. Цей стандарт є важливим інструментом для інтеграції ринку праці з освітньою та професійно-тренувальною сферами, сприяючи їхній узгодженій взаємодії.

Станом на 1 січня 2025 року ESCO описує та класифікує 3039 професій та 13939 навичок, доступних 28 мовами, серед яких усі офіційні мови ЄС, а також ісландська, норвезька, українська та арабська. Ця унікальна особливість робить класифікацію універсальною, полегшуючи її використання у різних соціально-економічних і культурних контекстах.

Відкрита база даних ESCO доступна на офіційному онлайн-порталі¹, де користувачі можуть безплатно ознайомлюватися із класифікаціями, завантажувати їх і використовувати у своїй діяльності. Завдяки цьому ESCO стає не лише інструментом стандартизації, а й джерелом для розвитку інноваційних підходів у сфері управління трудовими ресурсами, освіти та підготовки кадрів.

Однією з ключових характеристик ESCO є її здатність систематизувати взаємозв'язки між професіями та навичками, створюючи структуровану ієрархію, що є водночас гнучкою та адаптивною до змін ринку праці. Ці взаємозв'язки інтегруються у формати, зрозумілі електронним системам, що дозволяє використовувати їх для широкого спектра інноваційних рішень. Завдяки такій інтеграції ESCO стає основою для автоматизації багатьох процесів на онлайн-платформах, зокрема в галузях зайнятості, освіти та професійної підготовки.

Це відкриває широкий спектр можливостей, таких як:

- *пошук роботи з урахуванням компетенцій кандидатів*, що дозволяє точніше відповідати навичкам працівників вимогам роботодавців;

¹ Офіційний сайт ESCO: <https://esco.ec.europa.eu/en>

- *розробка індивідуальних програм перепідготовки та підвищення кваліфікації, які відповідають сучасним вимогам ринку, допомагаючи зберігати конкурентоспроможність кадрів;*
- *пропозиції щодо розвитку професійних траєкторій, що сприяє адаптації фахівців до нових економічних умов та вдосконалення їхніх кар'єрних перспектив.*

Ці функції не лише покращують якість взаємодії між роботодавцями, працівниками та освітніми установами, а й сприяють довгостроковій трансформації ринку праці, роблячи його більш прозорим, інклюзивним і динамічним.

Розробка ESCO здійснюється під егідою Європейської Комісії, зокрема Генерального директорату з питань зайнятості, соціальних питань та інклюзії. Перша офіційна версія класифікації (ESCO v1) була презентована 28 липня 2017 року, і відтоді ESCO стала загальнодоступною через спеціалізований портал, де її можна безплатно інтегрувати через API.

ESCO активно сприяє розробці новітніх цифрових рішень у сферах освіти, професійної підготовки та управління трудовими ресурсами. Її формат дозволяє розробникам, дослідникам та іншим зацікавленим сторонам інтегрувати класифікацію в сучасні інформаційні системи, розширюючи можливості для аналізу даних та прогнозування розвитку ринку праці.

ESCO відіграє ключову роль у побудові інклюзивного ринку праці, де кожна особа має можливість реалізувати свій потенціал. Вона сприяє розвитку освіти та професійного навчання, адаптуючи їх до потреб роботодавців і сучасних глобальних викликів.

Загалом, ESCO виступає стратегічним інструментом для інтеграції та оптимізації ринку праці, що дозволяє ефективно поєднувати потреби роботодавців, працівників і освітніх інституцій. У час швидких змін і глобальних викликів ESCO допомагає створювати умови для гармонійного розвитку, сприяючи зміцненню економічної стабільності та соціальної згуртованості в межах Європи та за її межами.

1.2. Роль ESCO у сучасному цифровому ринку праці

У сучасному світі, що швидко змінюється, ринок праці стикається з численними викликами та трансформаціями. Зростання професійної та географічної мобільності, збільшення частоти змін роботи та роботодавців, а також постійна поява нових навичок створюють нові вимоги до управління талантами. Цифрові платформи для пошуку роботи, такі як дошки вакансій, професійні соціальні мережі та інструменти для оцінювання навичок, докорінно змінюють процеси рекрутингу, навчання та кар'єрного розвитку.

Роботодавці, претенденти, а також освітні й навчальні установи дедалі більше покладаються на цифрові інструменти для публікації вакансій, подачі заяв, пошуку талантів та організації навчання. У цих умовах критично важливо мати чітку, актуальну та стандартизовану інформацію про навички, компетенції та кваліфікації. Саме в цьому контексті ESCO виступає стратегічним інструментом, що сприяє ефективності функціонування сучасного ринку праці.

Концепції та описи ESCO допомагають людям та організаціям глибше зрозуміти ключові аспекти професійного розвитку та управління талантами:

- *які знання та навички є необхідними для конкретної професії*

Завдяки ESCO люди можуть краще усвідомлювати вимоги ринку праці, що дозволяє їм адаптувати свої кар'єрні плани до актуальних умов.

- *які знання, навички та компетенції здобуваються в результаті отримання певної кваліфікації*

Це сприяє узгодженню освітніх програм із практичними потребами роботодавців.

- *які кваліфікації вимагаються для виконання певної професії*

Це полегшує процес рекрутингу, забезпечуючи чіткі критерії для пошуку фахівців.

ESCO також виконує важливу роль у заповненні прогалин у кваліфікаціях і навчальних програмах, надаючи компаніям і навчальним закладам можливість адаптувати свої підходи до сучасних вимог.

Використання концепцій ESCO полегшує перехід до нового формату взаємодії на ринку праці, сприяючи:

- *уніфікації термінології*

ESCO забезпечує спільну «мову» для опису професій, навичок і кваліфікацій, що дозволяє цифровим інструментам краще взаємодіяти та інтегруватися.

- *підтримці цифрових платформ*

Стандарти ESCO стають базою для розробки інструментів, які автоматизують процеси оцінювання, пошуку талантів і навчання.

- *інноваційності у сфері навчання та розвитку*

Інтеграція ESCO в освітні програми дозволяє гнучко реагувати на зміни у попиті на ринку праці.

1.3. Еволюція та вдосконалення ESCO

У сучасних умовах стрімкої трансформації ринку праці та освітніх систем класифікація ESCO загальноновизнано є важливим інструментом для забезпечення гармонійної взаємодії між цими двома сферами. Однак для збереження своєї релевантності ESCO потребує постійного вдосконалення.

Швидка поява нових професій і навичок, динамічні зміни в навчальних програмах і термінології вимагають регулярного оновлення класифікації, що можливо лише за умови тісного співробітництва з усіма зацікавленими сторонами. Регулярний обмін відгуками, пропозиціями та рекомендаціями з організаціями, які використовують ESCO, є фундаментальним для забезпечення її актуальності. Вдосконалення класифікації включає оновлення описів професій, навичок і кваліфікацій, інтеграцію нових концепцій, а також покращення її термінологічної бази та технічної сумісності з сучасними цифровими інструментами.

Основні етапи еволюції ESCO:

- *2013–2016: Демо-версії ESCO*

Розробка перших демонстраційних версій класифікації (ESCO v0 – ESCO v0.9) стала основою для пілотування та тестування ключових концепцій. Цей період був присвячений збору даних і перевірці функціональності системи.

- *2017 (липень): Офіційний запуск ESCO v1*

Перша повна версія ESCO включала три основні складники:

- ✓ навички та компетенції;
- ✓ професії;
- ✓ кваліфікації.

Випуск версії ESCO v1 після п'яти етапів пілотування та тестування знаменував значний прорив, забезпечивши підтримку 27 мов і інтеграцію на онлайн-платформі.

- *2017 (жовтень): Перша конференція з запуску ESCO*

У першій конференції взяли участь 300 делегатів з усієї Європи – представник урядових установ, бізнесу, освіти та ІТ-галузі. Захід включав 15 воркшопів, ключові промови, панельні дискусії, презентації найкращих практик використання ESCO та відзначення переможців європейського хакатону з аналізу великих даних. Конференція стала важливим кроком у популяризації ESCO як інструменту для ефективної роботи з ринком праці та освітнім сектором.

- *2018–2021: Підготовка масштабного оновлення*

Цей період був присвячений аналізу та вдосконаленню класифікації. Оновлення включали:

- ✓ розширення описів професій, навичок і компетенцій;
- ✓ удосконалення термінології;
- ✓ інтеграцію нових вимог до ІТ-застосунків.

- *2022: ESCO v1.1 та подальші оновлення*

✓ січень: Випущена версія ESCO v1.1 із новими професіями, міжгалузевими навичками та маркуванням для зелених і цифрових навичок (DigComp);

✓ лютий: Онлайн-конференція з запуску версії v1.1, яка зібрала понад 350 зацікавлених сторін;

✓ вересень: Оновлення ESCO v1.1.1, яке включало нові переклади, вдосконалену ієрархію та відповідність стандартам ISCO-08.

- *2024: Перспективи розвитку*

✓ лютий: Випуск ESCO v1.1.2 із повним перекладом українською мовою, виправленнями та покращенням описів професій.

✓ травень: Запуск ESCO v1.2, що включає:

- о нові професії, навички та концепції знань;
- о інтеграцію національних мов жестів;
- о проведення 18 типів покращень якості.

На підтримку запуску версії v1.2 було організовано дводенну онлайн-конференцію за участю понад 450 представників різних країн. Основний акцент зроблено на практичному використанні ESCO, а також на інтеграції класифікації в державах-членах ЄС та за їхніми межами.

1.4. Довідник для користувачів ESCOmedia

ESCOmedia – це довідковий інструмент ESCO, розроблений для полегшення розуміння ключових концепцій, термінів і принципів. Цей посібник спрямований на надання чітких, структурованих і деталізованих пояснень для ефективного використання ESCO у сфері зайнятості, освіти та професійної підготовки.

ESCOmedia виступає важливим джерелом знань як для новачків, які лише починають знайомитися з ESCO, так і для досвідчених користувачів, які шукають поглиблену інформацію про певні аспекти.

Основні функції ESCOmedia:

- *огляд за тематичними групами*

ESCOmedia організована за тематичними групами, які допомагають систематизувати знання та полегшують пошук потрібної інформації.

Наприклад:

✓ «Вступ до ESCO»: базова інформація про концепцію класифікації, її мету та застосування.

✓ «Розробка ESCO»: детальна історія створення класифікації, її ключові етапи та методологічні підходи.

Тематичний підхід дозволяє користувачам ефективно навігувати серед великої кількості матеріалів, зосереджуючись на конкретних аспектах, що їх цікавлять.

- *алфавітний індекс*

Для швидкого доступу до потрібної інформації ESCOmedia пропонує алфавітний перелік усіх доступних тем. Цей інструмент спрощує навігацію,

дозволяючи користувачам миттєво знаходити визначення, концепції чи пояснення, які їх цікавлять.

Функція «Ієрархічний перегляд» надає можливість побачити інформацію в структурованому алфавітному форматі, що сприяє глибшому розумінню взаємозв'язків між поняттями.

- *Інструмент пошуку*

Для тих, хто має конкретний запит або шукає специфічний термін чи концепцію, ESCOmedia пропонує функцію пошуку. Цей інструмент дозволяє:

- ✓ швидко знаходити необхідну інформацію, скорочуючи час на навігацію;

- ✓ переглядати тематично пов'язані ресурси;

- ✓ бачити додаткові матеріали, пов'язані з темою, що розглядається.

- *зворотний зв'язок і додаткова інформація*

ESCOmedia надає можливість користувачам залишати відгуки та звертатися до команди ESCO за додатковими роз'ясненнями чи матеріалами. Для цього створено спеціальну форму зворотного зв'язку, яка дозволяє ділитися пропозиціями, запитаннями чи побажаннями щодо покращення роботи ESCOmedia.

1.5. Компоненти ESCO

Європейська система класифікації навичок, компетенцій, кваліфікацій та професій складається з двох основних компонентів:

- *професійний компонент* – відповідає за опис та класифікацію професій;

- *компонент навичок, компетенцій і знань* – визначає, які саме знання, навички та компетенції є важливими для конкретних професій.

Разом ці компоненти створюють інтегровану структуру, яка забезпечує взаємозв'язок між освітніми програмами, кваліфікаціями та вимогами ринку праці. Вони сприяють гармонізації професійної термінології в Європі, полегшуючи міжнародне співробітництво, порівняння кваліфікацій і підвищення мобільності працівників.

1.5.1. Професійний компонент

Професійний компонент ESCO створено для опису, класифікації та впорядкування професій у різних галузях економіки. У чинній версії ESCO цей компонент містить 3039 професійних профілів, кожен з яких відповідає конкретній професії та структурований відповідно до Міжнародної стандартної класифікації професій (ISCO-08).

Кожен професійний профіль містить такі ключові характеристики:

- *основна та додаткова термінологія професії:*

- ✓ основна термінологія відображає офіційну назву професії;
- ✓ додаткова термінологія охоплює синоніми, альтернативні назви або менш поширені терміни.

- *формальне визначення, опис і примітки про сферу застосування:*

- ✓ визначення дає чітке розуміння сутності професії;
- ✓ опис деталізує завдання, функції та основні характеристики;
- ✓ примітки про сферу застосування окреслюють межі використання професії або специфіку її виконання в різних умовах.

- *ієрархічні зв'язки між професіями:*

Професії класифікуються за принципом ієрархії:

- ✓ загальні професії – ширші категорії, які включають підкатегорії спеціалізованих професій;

- ✓ спеціалізовані професії – вузькі напрями, які є частинами загальних професій.

- *зв'язок із компонентом навичок і компетенцій:*

- ✓ кожен професійний профіль визначає перелік знань, навичок і компетенцій, необхідних для виконання роботи;

- ✓ цей зв'язок забезпечує інтеграцію між професіями та вимогами ринку праці.

Професійний компонент забезпечує такі ключові функції:

- *упорядкування професій* – створює єдину класифікацію професій на основі європейських та міжнародних стандартів.

- *полегшення порівняння професій* – уніфікує професійні профілі, що полегшує їхню інтерпретацію у різних країнах Європи.

- *забезпечення прозорості* – чітко визначає вимоги до кожної професії, включно з навичками, знаннями та кваліфікаціями.

- *взаємодія з іншими компонентами* – інтегрується з компонентом навичок і компетенцій, створюючи цілісну систему для опису професійної діяльності.

1.5.2. Компонент навичок, компетенцій і знань

Компонент навичок, компетенцій і знань є ключовим елементом системи ESCO, який забезпечує деталізацію вимог до професійної діяльності. Він містить структурований перелік умінь, знань та компетенцій, що є актуальними для ринку праці. Цей компонент відповідає за визначення необхідних навичок і знань для конкретних професій і допомагає встановити чіткий зв'язок між професійною діяльністю, кваліфікаціями та освітніми програмами.

В чинній версії ESCO цей компонент містить 13939 навичок.

Навички, компетенції та знання поділяються на дві основні категорії:

- *обов'язкові (essential)* – ті, що є критично важливими для виконання основних робочих завдань.

Відсутність таких навичок або знань унеможливило б ефективне виконання професійної діяльності. Наприклад, для розробника баз даних це знання мови SQL та принципів проєктування баз даних.

- *додаткові (optional)* – ті, що можуть покращити результати роботи, але не є ключовими.

Вони забезпечують додаткову перевагу або полегшують виконання певних завдань. Наприклад, для розробника баз даних це можуть бути навички роботи з хмарними платформами управління базами даних.

Компонент охоплює широкий спектр умінь і знань, які можна поділити на дві основні групи:

- *загальні навички та компетенції*

Ці навички є універсальними та застосовуються у багатьох професіях. Приклади: аналітичне мислення, вміння розв'язувати технічні проблеми, управління проєктами, робота в міждисциплінарних командах.

- *спеціалізовані знання та компетенції*

Вони є специфічними для певних професій або галузей. Приклади: знання нормативно-правової бази, знання специфікацій галузевих стандартів.

1.6. Маркування цифрових навичок в ESCO

У межах Європейської програми навичок та ініціатив Європейської Комісії, спрямованих на подолання цифрового розриву, у жовтні 2022 року було впроваджено маркування цифрових навичок і знань у системі ESCO. Ця інновація дозволяє ідентифікувати цифрові компетенції, які є невіддільною частиною класифікації ESCO.

У системі ESCO цифрові навички визначаються відповідно до Рамки цифрових компетенцій для громадян (DigComp). Ця рамка створена для стандартизації термінології та опису ключових аспектів цифрової компетентності. DigComp базується на визначенні, затвердженому в Рекомендації Ради ЄС щодо ключових компетенцій для навчання протягом життя.

Цифрова компетенція включає впевнене, критичне та відповідальне використання цифрових технологій, а також активну участь у цифровому середовищі для навчання, роботи та соціальної активності. До її складників входять:

- інформаційна та дата-грамотність;
- комунікація та співпраця;
- медіаграмотність;

- створення цифрового контенту (включно з програмуванням);
- безпека (кібербезпека);
- питання інтелектуальної власності;
- розв'язання проблем і критичне мислення.

Процес маркування концепцій цифрових навичок та знань у системі ESCO базується на комбінації методів машинного навчання (ML) та ручного маркування, що реалізовано у п'ять етапів:

1. Ручне маркування концепцій навичок і знань ESCO відповідно до DigComp 2.2;
2. Ручне маркування вакансій, зібраних із бази даних EURES, для визначення цифрових компетенцій, які потрібні роботодавцям;
3. Розробка моделі машинного навчання, яка класифікує концепції ESCO як цифрові чи нецифрові;
4. Використання моделі для розрахунку ймовірності цифрового характеру кожної концепції ESCO;
5. Порівняння результатів автоматичного та ручного маркування для уточнення класифікації.

Цей підхід дозволив не лише вдосконалити точність класифікації, але й виявити концепції, що потребували додаткового аналізу. Таким чином було мінімізовано вплив людських помилок і недоліків навчальних даних.

У результаті виконаного процесу 1201 концепція знань, навичок і компетенцій у системі ESCO була класифікована як цифрова, зокрема:

- 718 навичок;
- 475 концепцій знань;
- 7 трансверсальних навичок.

Цифрові концепції навичок та знань ESCO доступні у розділі для завантажень. Для отримання повного списку необхідно завантажити класифікацію у бажаній версії. У пакеті завантаження буде присутній файл «digitalSkillsCollection».

На порталі ESCO також можна використовувати фільтри. Для доступу до цифрових концепцій ESCO потрібно активувати фільтр Digital Label у відповідному розділі.

Добірка прикладів цифрових навичок, концепцій знань та трансверсальних навичок, представлених у класифікації ESCO:

- *цифрові навички:*

1. програмування на Python: уміння розробляти програми мовою Python для різних застосувань;
2. вебдизайн: здатність створювати естетично привабливі та функціональні вебсайти;
3. аналіз великих даних (Big Data): навички обробки та інтерпретації великих обсягів даних для прийняття рішень;

4. кібербезпека: знання методів захисту інформаційних систем від несанкціонованого доступу;

5. розробка мобільних додатків: уміння створювати програми для мобільних платформ, таких як iOS та Android;

6. управління соціальними мережами: здатність ефективно використовувати платформи соціальних медіа для просування контенту;

7. 3D-моделювання: навички створення тривимірних моделей для різних галузей, включно з дизайном та інженерією;

8. хмарні обчислення (Cloud Computing): знання роботи з хмарними сервісами для зберігання та обробки даних;

9. SEO-оптимізація: уміння покращувати видимість вебсайтів у пошукових системах;

10. віртуальна та доповнена реальність (VR/AR): здатність розробляти та впроваджувати VR/AR-технології в різних сферах.

- *цифрові концепції знань:*

1. алгоритми та структури даних: розуміння основних алгоритмів та способів організації даних у програмуванні;

2. принципи об'єктноорієнтованого програмування (ООП): знання концепцій ООП, таких як класи, об'єкти та наслідування;

3. архітектура комп'ютерних систем: розуміння будови та функціонування апаратного та програмного забезпечення комп'ютерів;

4. теорія баз даних: знання принципів проєктування, створення та управління базами даних;

5. мережеві технології: розуміння принципів функціонування комп'ютерних мереж та протоколів передачі даних;

6. штучний інтелект та машинне навчання: знання методів розробки інтелектуальних систем, здатних до навчання та прийняття рішень;

7. криптографія: розуміння методів захисту інформації шляхом шифрування та інших криптографічних технік;

8. принципи розробки інтерфейсів користувача (UI/UX): знання підходів до створення зручних та інтуїтивно зрозумілих інтерфейсів;

9. методології розробки програмного забезпечення: розуміння різних підходів до організації процесу розробки, таких як Agile чи Scrum;

10. етика в інформаційних технологіях: знання етичних аспектів використання та розробки цифрових технологій.

- *7 трансверсальних навичок:*

1. критичне мислення: здатність аналізувати інформацію та робити обґрунтовані висновки.

2. комунікативні навички: ефективне спілкування як усно, так і письмово.

3. робота в команді: здатність співпрацювати з іншими для досягнення спільних цілей.

4. адаптивність: гнучкість та здатність пристосовуватися до змін.
5. керування часом: ефективне планування та розподіл часу.
6. креативність: здатність генерувати нові та оригінальні ідеї.
7. розв'язання проблем: уміння знаходити рішення в складних ситуаціях.

1.7. Характеристика та вимоги до професії «Розробник баз даних»

Розробник баз даних – це фахівець, який створює, налаштовує та підтримує бази даних для ефективного зберігання, обробки й аналізу інформації. У сучасну епоху цифровізації ця професія має ключове значення, адже якісна робота з даними лежить в основі прийняття рішень у бізнесі, науці, державному управлінні та інших сферах.

Згідно з класифікацією ESCO, професія «Розробник баз даних» має код ISCO-08: 2521.3 та охоплює завдання, які потребують технічних знань, аналітичного мислення та стратегічного підходу. Основна мета роботи – проектування та підтримка баз даних для забезпечення їхньої доступності, зручності й безпеки для користувачів.

Вимоги до кваліфікації розробників баз даних визначаються очікуваннями ключових зацікавлених сторін:

- *роботодавці*, які потребують фахівців, здатних створювати надійні, продуктивні та безпечні бази даних для підтримки бізнесу;
- *освітні установи*, що забезпечують підготовку кадрів відповідно до актуальних потреб ринку праці;
- *професійні асоціації*, які формують стандарти та критерії оцінювання компетенцій фахівців.

Ці сторони визначають ключові компетенції, необхідні для успішної реалізації професійних обов'язків у сфері проектування, впровадження, налаштування та підтримки баз даних.

Числовий код 2521.3 в описі професії «Розробник баз даних» має таке значення:

- **252** – *тризначний код, який визначає загальну групу професій*

У цьому випадку це професії, пов'язані з інформаційними та комунікаційними технологіями (ІКТ), а саме «Фахівці з розробки програмного забезпечення та прикладних систем»;

- **2521** – *четверта цифра деталізує підгрупу професій*

Підгрупа 2521 – «Розробників прикладного програмного забезпечення та баз даних»;

- **.3** – *підрівень, який уточнює спеціалізацію всередині підгрупи*

В цьому випадку це «Розробник баз даних», що наголошує на вузькій спеціалізації розробки, впровадження та управління базами даних.

Для успішної реалізації професійних обов'язків у сфері проектування, впровадження, налаштування та підтримки баз даних визначено 13 ключових компетенцій, які широко використовуються та регулярно перевіряються.

1.7.1. Компетенції професії «Розробник баз даних»

Затверджені компетенції є критично важливими для фахівців, які займаються проєктуванням, впровадженням, налаштуванням та підтримкою баз даних, забезпечуючи їхню ефективну та безпечну роботу.

1. Застосовувати політику інформаційної безпеки

Успішне застосування політики інформаційної безпеки вимагає розробки, впровадження та підтримки ефективної системи захисту даних у базах даних. Цей процес передбачає всебічний підхід, що включає технічну експертизу, аналітичні здібності та стратегічне мислення.

На першому етапі спеціаліст визначає потенційні загрози, що можуть вплинути на безпеку даних, такі як хакерські атаки, витік інформації, фішингові схеми чи вірусні програми. Аналізуючи ці ризики, розробляються політики безпеки, які регламентують правила роботи з даними, наприклад, порядок доступу до бази даних, використання надійних паролів, шифрування інформації та дотримання протоколів передачі даних.

Ключовою частиною цього процесу є впровадження технічних рішень, таких як шифрування даних, багаторівнева автентифікація та системи моніторингу доступу. Шифрування забезпечує конфіденційність інформації, навіть якщо дані потрапляють у небажані руки, тоді як автентифікація додає рівень захисту, перевіряючи ідентичність користувачів. Системи моніторингу доступу дозволяють відстежувати, хто і коли отримує доступ до даних, що допомагає швидко виявляти підозрілу активність.

Не менш важливою є відповідність політики безпеки законодавчим вимогам, таким як GDPR² або інші регламенти, що регулюють захист персональних даних. Спеціаліст має забезпечувати прозорість у роботі з персональними даними, гарантувати їхнє зберігання відповідно до вимог законодавства, а також надавати користувачам можливість керувати своїми даними.

Особливу увагу слід приділяти людському фактору, адже навіть найкращі технічні системи можуть бути вразливими через помилки персоналу. Для цього організовуються регулярні тренінги, які підвищують обізнаність співробітників щодо сучасних кіберзагроз та процедур захисту даних.

Впроваджена політика безпеки потребує регулярного оновлення та перевірки. Система повинна адаптуватися до нових викликів, зокрема до по-

² General Data Protection Regulation (Загальний регламент захисту даних) – <https://gdpr-info.eu/>

яви нових видів загроз. Регулярне тестування безпеки, наприклад, проведення симуляцій атак, дозволяє оцінити готовність системи до реальних ризиків.

Інтеграція баз даних із зовнішніми системами також має бути захищеною. Підключення через API, наприклад, повинно виконуватися з чіткими правилами доступу, щоб уникнути потенційних вразливостей.

2. Збалансовувати ресурси бази даних

Балансування ресурсів бази даних – це комплексний процес, спрямований на оптимізацію використання ключових системних ресурсів, таких як пам'ять, процесорний час, мережевий трафік та інші елементи інфраструктури. Ефективне балансування забезпечує швидкодію бази даних, її стабільність та доступність для користувачів, навіть за умов високих навантажень.

Спеціаліст із баз даних виконує регулярний аналіз продуктивності системи, щоб виявити так звані «вузькі місця». Це можуть бути повільно виконувані запити, перевантаження серверів або нерівномірне розподілення ресурсів між різними компонентами системи. Виявивши проблему, фахівець впроваджує заходи для її усунення. Наприклад, одним з ефективних підходів є індексація таблиць, яка дозволяє значно прискорити виконання запитів шляхом створення структури даних, що забезпечує швидкий доступ до потрібної інформації.

Для великих систем із розподіленим навантаженням фахівець може використовувати методи кластеризації, коли база даних розділяється на декілька серверів, які працюють спільно для обробки запитів. Це дозволяє розподілити навантаження між кількома вузлами системи, забезпечуючи високу продуктивність навіть у пікові періоди. Інший підхід – впровадження розподіленого зберігання даних, що дозволяє розміщувати різні частини бази даних у різних фізичних чи віртуальних місцях для оптимізації використання ресурсів.

Ще однією ключовою технологією є налаштування пулів з'єднань. Це дозволяє ефективніше використовувати наявні з'єднання з базою даних, мінімізуючи час, потрібний для їхнього створення та закриття. Завдяки цьому підходу підвищується загальна продуктивність системи, особливо для додатків із високою частотою звернень до бази даних.

Особлива увага приділяється моніторингу та підтримці системи в реальному часі. Це включає використання спеціальних інструментів для аналізу навантаження, таких як метрики використання процесора, оперативної пам'яті, обсягу зчитуваних і записуваних даних. На основі отриманих даних спеціаліст може вносити корективи до конфігурації системи, що дозволяє запобігти перевантаженням до того, як вони стануть критичними.

Кінцева мета балансування ресурсів бази даних – досягнення гармонійного поєднання швидкодії, стабільності та надійності системи. Це забезпе-

чує позитивний користувацький досвід, адже користувачі отримують доступ до даних швидко та без збоїв. Крім того, оптимізація використання ресурсів допомагає зменшити витрати на інфраструктуру, що є важливим для бізнесу.

3. Збирати відгуки клієнтів про додатки

Збір зворотного зв'язку від користувачів охоплює створення інструментів для отримання даних від користувачів, таких як опитування, інтерв'ю, фокус-групи або аналіз поведінки під час роботи з додатками. Завдяки цьому спеціаліст може отримати цінну інформацію про досвід користувачів, виявляти проблеми у функціонуванні системи, незручності в інтерфейсі або недоліки у функціональності програмного забезпечення.

Одним із ключових аспектів збору зворотного зв'язку є вміння формулювати запитання, які максимально точно відображають потреби та враження користувачів. Аналіз поведінки користувачів, що ґрунтується на автоматично зібраних даних, таких як частота використання певних функцій чи виникнення помилок, дозволяє доповнити цю картину об'єктивними спостереженнями.

Після збору інформації фахівець повинен її ретельно проаналізувати. Це вимагає не лише технічних знань, але й розуміння потреб кінцевих користувачів. Аналіз дозволяє виявити системні проблеми, які можуть негативно впливати на продуктивність чи задоволення користувачів. Наприклад, якщо користувачі скаржаться на повільне завантаження даних, це може свідчити про проблеми з оптимізацією запитів або недостатні ресурси серверів.

На основі отриманих даних спеціаліст формулює конкретні рекомендації для вдосконалення системи. Це може включати технічні рішення, такі як оптимізація індексів бази даних, розширення серверних потужностей або зміни у функціоналі додатків, наприклад, спрощення інтерфейсу чи впровадження нових функцій. Усі рекомендації мають бути орієнтовані на те, щоб забезпечити максимальну відповідність продукту потребам користувачів.

Зворотний зв'язок є не лише засобом виявлення недоліків, а й важливим інструментом для підвищення конкурентоспроможності продукту. Розуміння потреб користувачів і швидке реагування на їхні запити сприяють покращенню якості роботи додатків, підвищенню задоволення користувачів і, відповідно, зростанню їхньої лояльності.

4. Створювати моделі даних

Моделювання починається з аналізу вимог до даних, що визначають потреби бізнесу або організації. На основі цих вимог спеціаліст із баз даних формує концептуальну модель, яка відображає бізнес-логіку та визначає основні об'єкти системи і взаємозв'язки між ними. Концептуальна модель слугує високорівневим представленням даних, яке є зрозумілим як для технічних фахівців, так і для представників бізнесу.

Наступним кроком є створення логічної моделі даних. Логічна модель деталізує структуру даних, формуючи чіткі уявлення про атрибути об'єктів, типи даних, які будуть використовуватися, та зв'язки між таблицями. Логічна модель є більш технічним етапом, оскільки вона забезпечує основу для подальшої розробки фізичної моделі.

Фізична модель даних є кінцевим етапом моделювання і враховує специфіку конкретної системи управління базами даних (СУБД), яка буде використовуватися. На цьому етапі спеціаліст вирішує, як саме дані будуть зберігатися на фізичному рівні, включно з вибором формату зберігання, індексів, стратегій шардінгу та розподілу даних між серверами. Фізична модель має враховувати технічні особливості інфраструктури, такі як продуктивність апаратного забезпечення та оптимізація ресурсів.

Однією з ключових задач під час моделювання є нормалізація даних. Цей процес спрямований на усунення надлишкових даних, що може знижувати продуктивність бази даних та ускладнювати її оновлення. Нормалізація допомагає структурувати дані так, щоб уникнути дублювання та забезпечити їхню цілісність. Водночас фахівець визначає зв'язки між таблицями, що включають залежності типу «один до одного», «один до багатьох» та «багато до багатьох».

Якісна модель даних є критично важливою, оскільки вона забезпечує ефективне зберігання та обробку інформації. Вона дозволяє мінімізувати час виконання запитів, зменшити використання ресурсів системи та гарантує цілісність даних. Крім того, добре продумана модель спрощує подальше обслуговування бази даних, робить її масштабованою і здатною адаптуватися до змін у вимогах бізнесу.

Моделювання даних – це не одноразовий процес, а постійний цикл аналізу, проєктування та вдосконалення. З часом, у міру змін потреб організації чи зростання обсягів даних, модель може потребувати перегляду для забезпечення її відповідності новим умовам. Завдяки ретельному підходу до моделювання даних створюється міцна основа для роботи баз даних, яка підтримує їхню продуктивність, надійність і масштабованість у довгостроковій перспективі.

5. Оцінювати тривалість робіт

Основна мета оцінки тривалості робіт – забезпечити реалістичне планування, що дозволяє уникнути затримок, перевитрати бюджету і неефективного використання ресурсів.

Фахівець починає з визначення основних завдань, які необхідно виконати, та їхнього розбиття на дрібніші підзавдання. Це дозволяє глибше зрозуміти обсяг робіт і врахувати всі деталі, які можуть вплинути на тривалість виконання. Кожне підзавдання оцінюється окремо, що дозволяє точніше розрахувати загальний час, необхідний для реалізації проєкту.

Одним з ефективних методів оцінки є використання аналогій із попередніми проєктами. Якщо подібні завдання вже виконувалися раніше, можна спиратися на досвід і врахувати час, витрачений на аналогічні етапи. Це особливо корисно для прогнозування тривалості типових процесів, таких як налаштування бази даних, тестування чи впровадження нових функцій.

Ще одним поширеним методом є розрахунок критичного шляху, який допомагає визначити послідовність завдань, що мають найбільший вплив на загальний термін проєкту. Цей підхід дозволяє сконцентруватися на критичних етапах, запобігти їхній затримці та оцінити, як затримки в інших завданнях можуть вплинути на кінцевий результат.

У процесі оцінки враховуються також технічна складність завдань і можливі ризики. Наприклад, інтеграція бази даних із зовнішніми системами чи налаштування безпеки можуть потребувати більше часу через непередбачувані труднощі. Аналіз ризиків дозволяє заздалегідь передбачити додатковий час на розв'язання потенційних проблем, тим самим мінімізуючи їхній вплив на загальний термін виконання проєкту.

Оцінка тривалості робіт не є разовим процесом. У міру реалізації проєкту можуть виникати нові завдання, змінюватися обставини або виявлятися непередбачувані фактори. Тому фахівець має регулярно переглядати та коригувати оцінки, щоб забезпечити їхню актуальність і точність.

6. Визначати вимоги замовника

Цей етап передбачає тісну співпрацю зі сторонами, зацікавленими в проєкті, з метою повного розуміння їхніх потреб і очікувань. Для досягнення цієї мети спеціаліст використовує різні підходи, такі як проведення інтерв'ю, воркшопів, аналіз наявних систем та вивчення наявної документації.

Інтерв'ю дозволяють зібрати індивідуальні думки та вимоги кожного ключового учасника проєкту. Це можуть бути представники бізнесу, кінцеві користувачі або інші зацікавлені сторони. Завдяки персоналізованому підходу спеціаліст отримує глибше розуміння специфічних потреб, які можуть не бути очевидними на перший погляд. Воркшопи, у свою чергу, надають можливість зібрати різні групи зацікавлених сторін для спільного обговорення вимог, узгодження пріоритетів і вирішення потенційних конфліктів у їхніх очікуваннях.

Аналіз наявних систем і документації є не менш важливим елементом цього процесу. Він дозволяє вивчити наявні інфраструктури та знайти способи їхньої оптимізації або інтеграції з новою базою даних. Наприклад, аналіз поточних потоків даних може виявити проблеми, які необхідно усунути, або можливості, які слід використати.

У процесі збору вимог спеціаліст має враховувати як функціональні, так і нефункціональні аспекти. Функціональні вимоги стосуються того, які саме дані необхідно зберігати, як вони будуть організовані та які запити повинні підтримуватися. Наприклад, клієнт може вимагати зберігати інформацію про користувачів, транзакції або історію покупок. Нефункціональні вимоги, навпаки, зосереджені на характеристиках роботи бази даних, таких як швидкодія, надійність, безпека та масштабованість. Ці аспекти визначають, наскільки добре база даних виконуватиме свої завдання в реальних умовах експлуатації.

Особливу увагу слід приділити майбутнім потребам бізнесу. Технологічне середовище та бізнес-вимоги постійно змінюються, тому база даних повинна бути спроектована так, щоб забезпечити легку адаптацію до нових умов. Це може включати можливість збільшення обсягів даних, підтримку нових функцій або інтеграцію з іншими системами в майбутньому.

Збір вимог – постійний процес взаємодії з клієнтом протягом усього циклу розробки. У міру уточнення вимог можуть виникати нові деталі, які потрібно враховувати. Тісна комунікація з клієнтом допомагає уникнути непорозумінь, забезпечити реалізацію ключових функцій і створити продукт, який максимально відповідає очікуванням замовника.

7. Перекладати технічні тексти

Основним завданням перекладача є забезпечення точності передання інформації, одночасно роблячи її зрозумілою для цільової аудиторії. Це потребує не лише досконалого володіння мовою оригіналу і мовою перекладу, але й глибокого розуміння технічної термінології та специфіки галузі.

Процес перекладу технічної документації охоплює широкий спектр завдань, таких як переклад інструкцій, технічних специфікацій, звітів, посібників із використання, протоколів і навіть технічних креслень. Кожен із цих типів документів вимагає особливого підходу. Наприклад, переклад інструкцій потребує чіткості та послідовності, щоб користувач міг легко виконати необхідні дії. У технічних специфікаціях важливо дотримуватися максимальної точності, адже будь-яка помилка може призвести до серйозних наслідків під час реалізації проєкту або виготовлення продукту.

Перекладач повинен бути готовий адаптувати текст відповідно до знань і потреб цільової аудиторії. У багатьох випадках користувачі технічної документації не мають глибоких технічних знань, тому перекладач має знаходити баланс між використанням точної термінології та спрощенням складних концепцій для їхнього легшого розуміння. Це включає пояснення складних термінів, надання прикладів або додавання контексту там, де це необхідно.

Окрім того, перекладач повинен враховувати культурні й мовні особливості регіону, для якого виконується переклад. Наприклад, одиниці

вимірювання, стилі дат або навіть формати документів можуть відрізнятися залежно від країни. Усі ці нюанси слід враховувати, щоб переклад відповідав очікуванням аудиторії.

Процес перекладу технічної документації також включає ретельне вичитування та перевірку точності. Це необхідно, щоб гарантувати відповідність перекладеного тексту оригінальному, зокрема при роботі зі складними формулами, графіками або таблицями. Використання спеціального програмного забезпечення для перекладу, таких як системи комп'ютерної трансляції або інструменти для роботи з термінологією, може значно підвищити ефективність і точність перекладу.

8. Створювати резервні копії

Створення резервних копій спрямоване на мінімізацію ризиків втрати інформації, що може статися через технічні збої, вплив вірусів, помилки користувачів або навіть зовнішні загрози, такі як кібератаки. Завдяки резервному копіюванню дані можуть бути відновлені в разі непередбачених обставин, забезпечуючи безперервність роботи системи та збереження важливої інформації.

Процес резервного копіювання починається з вибору типу резервування, який залежить від обсягу даних, частоти їхнього оновлення та вимог до відновлення. Повне резервне копіювання створює копію всієї бази даних, що забезпечує максимальну безпеку, але вимагає більше часу і ресурсів. Інкрементальне резервування фіксує лише зміни, що сталися після останнього копіювання, зменшуючи використання ресурсів, але ускладнюючи процес відновлення. Диференційне резервування копіює всі зміни з моменту останнього повного резервування, забезпечуючи баланс між ефективністю зберігання і швидкістю відновлення.

Частота створення резервних копій також є критично важливою. Спеціаліст має враховувати частоту оновлення даних, важливість цих даних і рівень ризику їхньої втрати. Наприклад, для фінансових систем або баз даних з високою активністю резервні копії можуть створюватися щогодини або навіть частіше. Для менш критичних систем підходить резервування раз на день або на тиждень. Вибір часу для виконання процесу резервування також має значення, щоб мінімізувати вплив на продуктивність системи під час її активного використання.

Ще одним важливим аспектом є тестування можливості відновлення даних із резервних копій. Створення копії саме по собі не гарантує безпеки, якщо ці дані неможливо відновити в разі потреби. Регулярне тестування процесу відновлення дозволяє переконатися в надійності резервних копій, а також виявити та усунути потенційні проблеми до того, як вони стануть критичними.

Автоматизація процесу резервного копіювання є ключовим завданням для спеціаліста. Завдяки використанню спеціального програмного забезпечення процес створення резервних копій може бути виконаний без участі людини, за заздалегідь визначеним графіком. Це не лише підвищує ефективність, але й мінімізує ймовірність помилок, викликаних людським фактором.

9. Звітувати про результати аналізу

Підготовка звітів включає кілька важливих етапів, починаючи зі збору необхідних даних і закінчуючи їхнім поданням у формі, що дозволяє легко зрозуміти висновки та прийняти відповідні рішення.

Першим кроком у підготовці звіту є збір даних. Спеціаліст аналізує доступні джерела, визначає релевантні показники та відбирає необхідну інформацію, яка відповідає меті звіту. Це може включати дані про фінансові результати, поведінку користувачів, стан ринку або ефективність внутрішніх процесів. Важливо, щоб дані були актуальними, точними та відповідали критеріям, визначеним на початку аналізу.

Наступний етап – це обробка та візуалізація даних. Спеціаліст використовує сучасні інструменти бізнес-аналітики, такі як Power BI або Tableau, для створення графіків, таблиць, дашбордів та інших візуальних елементів. Ці інструменти дозволяють перетворити великі обсяги даних на наочну та зрозумілу інформацію. Графіки допомагають виявляти тренди, таблиці надають детальний аналіз, а дашборди забезпечують інтерактивний доступ до ключових показників.

Після візуалізації спеціаліст формує звіт, додаючи текстовий опис, який пояснює результати аналізу. Опис має бути чітким і зрозумілим для аудиторії, незалежно від рівня її технічної підготовки. Важливо не лише представити результати, але й зробити акцент на висновках, які можна з них зробити. Наприклад, якщо аналіз даних показав зниження продажів у певному регіоні, звіт має включати потенційні причини цього та пропозиції щодо розв'язання проблеми.

Звіти повинні відповідати конкретним запитам аудиторії. Для керівництва компанії важливо представити ключові показники, які впливають на стратегічні рішення, тоді як для операційної команди можуть бути корисні детальні дані про робочі процеси. Крім того, стиль і структура звіту мають бути адаптовані до аудиторії, щоб інформація була зрозумілою та легко сприймалася.

Готовий звіт стає важливим інструментом для прийняття рішень, орієнтованих на дані. Він допомагає виявити слабкі місця, визначити нові можливості, прогнозувати майбутні тенденції та оцінювати ефективність попередніх дій.

10. Тестувати ІКТ-запити

Тестування запитів до баз даних: процес охоплює комплекс перевірок, спрямованих на оцінку коректності, ефективності та захищеності виконуваних запитів, щоб гарантувати їхню відповідність очікуванням і вимогам.

Перший етап тестування полягає у перевірці коректності запитів. Спеціаліст аналізує, чи запит виконує правильні операції над даними, повертає очікувані результати та враховує всі передбачені умови. Наприклад, якщо запит має вибирати дані для певного періоду чи групи користувачів, необхідно переконатися, що повернуті результати відповідають заданим критеріям. Ця перевірка є ключовою для забезпечення точності та надійності роботи бази даних.

Наступним аспектом є оцінка ефективності виконання запитів. Тестування включає вимірювання часу виконання запиту, аналіз споживання ресурсів системи, таких як пам'ять та процесорний час, і виявлення потенційних вузьких місць у роботі бази даних. Якщо запит виконується занадто повільно, спеціаліст може оптимізувати його, використовуючи індекси, переписуючи структуру запиту або змінюючи спосіб обробки даних. Ефективність запитів є критично важливою для великих систем, де навіть невелика затримка у виконанні може призвести до значного зниження продуктивності загальної системи.

Безпека запитів – ще один важливий аспект тестування. Спеціаліст повинен переконатися, що запити не створюють вразливостей у системі, які можуть бути використані зловмисниками. Особливу увагу приділяють запобіганню SQL-ін'єкціям, які є одним із найпоширеніших способів атак на бази даних. Для цього використовуються такі заходи, як параметризовані запити, використання ORM (Object-Relational Mapping) інструментів і перевірка введених даних на наявність шкідливих елементів.

Окрім того, тестування включає перевірку відповідності запитів вимогам конфіденційності. Це передбачає перевірку, чи запити дотримуються правил доступу до даних і чи не надають користувачам можливість отримувати інформацію, яку вони не мають права бачити. Наприклад, звітні запити повинні забезпечувати анонімність даних, якщо це передбачено політиками безпеки, або обмежувати доступ до певних полів чи записів на основі рівня доступу користувача.

Тестування запитів є ітеративним процесом, що передбачає постійне вдосконалення. У міру розвитку системи або зміни бізнес-вимог запити можуть потребувати оновлення, тому їх необхідно повторно перевіряти на коректність, ефективність і безпеку. Використання сучасних інструментів моніторингу та аналізу запитів дозволяє автоматизувати частину процесу тестування, зменшуючи навантаження на спеціаліста та підвищуючи точність перевірок.

11. Використовувати інтерфейс конкретного додатка

Спеціаліст повинен мати чітке уявлення про всі можливості, які надає інтерфейс. Це включає вивчення доступних інструментів для керування даними, таких як запити, інструменти адміністрування, функції аналізу та звітування. Знання цих можливостей дозволяє обирати найефективніші підходи до виконання різних завдань, зокрема оптимізацію запитів, моніторинг продуктивності або інтеграцію з іншими системами.

Одним із ключових аспектів використання інтерфейсу є налаштування прав доступу. Спеціаліст повинен уміти створювати та налаштовувати ролі, визначати рівні доступу для різних категорій користувачів та забезпечувати відповідність цим налаштуванням політик безпеки. Це дозволяє забезпечити захист даних від несанкціонованого доступу, водночас надаючи користувачам можливість виконувати свої завдання без перешкод.

Інтерфейс бази даних також часто пропонує функції для автоматизації рутинних процесів. Спеціаліст має використовувати ці можливості для спрощення управління даними, зокрема налаштування автоматичного резервного копіювання, регулярного очищення даних або створення розкладів для виконання специфічних запитів. Автоматизація допомагає мінімізувати ризики помилок, пов'язаних із людським фактором, і підвищити загальну ефективність роботи з базою даних.

Крім того, спеціаліст повинен володіти інструментами для імпорту та експорту даних. Це включає вміння інтегрувати базу даних із зовнішніми системами, переносити дані між різними форматами та забезпечувати цілісність і точність інформації під час цих операцій. Наприклад, у проєктах з великими обсягами даних може знадобитися імпорт даних із зовнішніх джерел або експорт для подальшого аналізу за допомогою спеціалізованого програмного забезпечення.

Ще одним важливим аспектом є можливість розробки кастомних рішень для специфічних завдань. Інтерфейси баз даних часто дозволяють використовувати скрипти чи інтеграцію з мовами програмування, такими як Python або SQL, для створення індивідуальних рішень. Це може включати автоматизовану обробку даних, налаштування спеціальних звітів або інтеграцію з іншими бізнес-додатками.

12. Використовувати бази даних

Фахівець по роботі з базами даних повинен мати глибокі знання про такі популярні СУБД, як MySQL, PostgreSQL, Oracle, MongoDB та інші, щоб бути готовим працювати з різними технологічними рішеннями залежно від вимог проєкту чи організації.

Однією з основних навичок є знання мови SQL, яка є стандартом для роботи з реляційними базами даних. Спеціаліст повинен вміти створювати

запити для отримання, оновлення, видалення та вставки даних, а також використовувати складніші конструкції, такі як об'єднання таблиць, вкладені запити та функції агрегації. Крім того, знання особливостей мови SQL у різних СУБД дозволяє адаптувати запити до конкретної системи, враховуючи її синтаксис і функціональні можливості.

Робота із СУБД також передбачає розуміння принципів проектування баз даних. Фахівець повинен уміти створювати структури, які забезпечують ефективне зберігання, доступ і обробку даних. Це включає нормалізацію таблиць, оптимізацію зв'язків між ними та врахування вимог до продуктивності й масштабованості. Проектування бази даних має враховувати як поточні, так і майбутні потреби бізнесу, щоб уникнути необхідності радикальних змін у структурі в майбутньому.

Ще одним важливим аспектом є навички налаштування та оптимізації баз даних. Спеціаліст повинен уміти конфігурувати параметри бази даних відповідно до вимог проєкту, наприклад, налаштовувати кешування, індекси чи стратегії зберігання даних. Оптимізація бази даних включає аналіз продуктивності запитів, усунення вузьких місць у системі та впровадження технологій, які забезпечують швидший доступ до даних.

Розуміння архітектури бази даних є ще одним важливим елементом роботи із СУБД. Це знання дозволяє спеціалісту оцінювати, як дані зберігаються, як забезпечується їхня цілісність і які механізми використовуються для підтримання стабільної роботи системи. Наприклад, у випадку MongoDB важливо розуміти, як працюють документи та колекції, тоді як в Oracle спеціаліст має враховувати специфіку таблиць, індексів і процедур.

13. Писати документацію до баз даних

Створення документації для бази даних є важливим завданням, яке забезпечує зрозуміле й структуроване представлення інформації про її структуру, функціональність та використання. Цей процес охоплює підготовку матеріалів, що детально пояснюють, як працює база даних, і надають користувачам та адміністраторам необхідні інструкції для її ефективного використання та обслуговування.

Перш за все, документація повинна містити опис структури бази даних. Це включає схеми таблиць, зв'язки між ними, ключі та індекси, а також пояснення логіки їхньої організації. Такі матеріали допомагають зрозуміти, як дані зберігаються, як вони пов'язані між собою та які механізми забезпечують їхню цілісність. Це особливо важливо для розробників, які працюють із базою даних, оскільки дозволяє швидко орієнтуватися у її структурі.

Окрім опису структури, документація має пояснювати принципи роботи бази даних. Це включає опис функціональних можливостей, таких як підтримка транзакцій, використання індексів для оптимізації запитів

або механізми забезпечення безпеки даних. Такі розділи допомагають користувачам і адміністраторам зрозуміти, як працює база даних і які її можливості можна використовувати для досягнення максимального ефекту.

Важливою частиною документації є інструкції для кінцевих користувачів та адміністраторів. Для користувачів це можуть бути покрокові керівництва, що пояснюють, як виконувати певні завдання, такі як створення запитів, імпорту або експорту даних, або робота з інтерфейсом бази даних. Для адміністраторів документація повинна охоплювати складніші аспекти, такі як налаштування системи, моніторинг продуктивності, резервне копіювання та відновлення, а також усунення можливих проблем.

Документація має бути чіткою, логічно структурованою та легкою для розуміння. Інформація в ній повинна бути подана у спосіб, який дозволяє швидко знаходити потрібні відомості. Використання заголовків, підзаголовків, списків, таблиць і графічних елементів, таких як схеми та діаграми, допомагає зробити матеріал більш наочним і зрозумілим.

1.7.2. Розподіл навичок і компетенцій у професії «Розробник баз даних»

На рис. 1 наведено розподіл навичок і компетенцій у професії «Розробник баз даних».

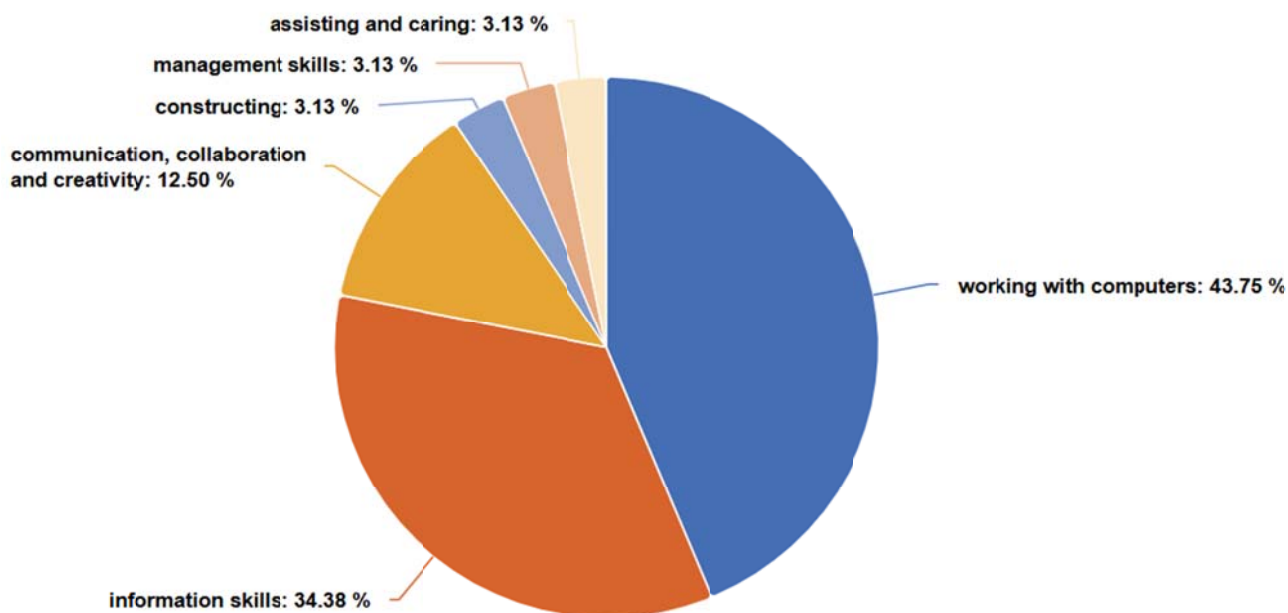


Рис. 1 – Розподіл навичок і компетенцій у професії «Розробник баз даних»

1. Робота з комп'ютерами (Working with computers) – 43.75 %

Це найбільша частка навичок у професії «Розробник баз даних». Вона включає глибоке розуміння програмного забезпечення, операційних систем, роботи з комп'ютерними мережами та автоматизованими систе-

мами. Розробник повинен володіти комп'ютерними навичками на високому рівні, адже вся їхня діяльність напряду пов'язана із програмуванням, налаштуванням та підтримкою баз даних.

2. Навички роботи з інформацією (Information skills) – 34.48 %

Цей аспект охоплює вміння аналізувати, зберігати, сортувати та обробляти інформацію. Розробники баз даних часто працюють з великими обсягами даних, забезпечують їхню цілісність та якість. Навички роботи з інформацією є критично важливими для створення ефективних баз даних, здатних відповідати бізнес-вимогам.

3. Комунікація, співпраця та креативність (Communication, collaboration and creativity) – 12.5 %

Ця категорія займає меншу частку, вона важлива для успішної взаємодії з клієнтами, командою розробників та іншими зацікавленими сторонами. Вміння спільно працювати над проєктами, презентувати ідеї та генерувати креативні рішення – важливі аспекти роботи.

4. Проєктування баз даних (Constructing) – 3.13 %

Цей напрямок охоплює розробку архітектури баз даних, структурування таблиць, зв'язків і правил, які забезпечують їхню ефективну роботу. Хоча аспект має невелику частку, він є основою для створення систем, що відповідають вимогам бізнесу.

5. Навички управління базами даних (Management skills) – 3.13 %

Управління базами даних передбачає забезпечення їхньої стабільної роботи, налаштування доступу, моніторинг продуктивності та впровадження оновлень. Цей аспект є важливим для підтримки високої надійності систем.

6. Підтримка та обслуговування баз даних (Assisting and caring) – 3.13 %

Ця категорія охоплює регулярне обслуговування баз даних, забезпечення їхньої безпеки та резервного копіювання, а також розв'язання проблем користувачів. Вона є критичною для забезпечення безперервної роботи баз даних.

Проаналізувавши розподіл навичок і компетенцій, можна зробити такі висновки:

- *пріоритетність навичок роботи з комп'ютерами та інформацією*

Разом ці категорії займають понад 78 % від загального розподілу навичок. Це свідчить про те, що професія «Розробник баз даних» вимагає від спеціалістів глибоких технічних знань та вмінь працювати з інформацією.

- *аспекти співпраці та креативності мають значно меншу вагу*

Такий дисбаланс між технічними та соціальними навичками є типовим для ІТ-сфери. Комунікація і креативність складають разом 12.5 %. Ці навички (soft skills) необхідні для розуміння потреб бізнесу, узгодження технічних рішень із командою та пропонування інноваційних ідей для покращення систем.

- *рівномірний розподіл менших компетенцій*

Проектування, управління та підтримка баз даних мають однаковий відсоток (3.13 % кожна). Це свідчить про те, що ці аспекти є рівноцінними і їхній розвиток має значення для виконання професійних обов'язків.

Розподіл навичок демонструє, що професія значною мірою зосереджена на роботі з технічними системами. Це робить професію актуальною для людей із технічним складом мислення та аналітичними здібностями.

ВИСНОВКИ ЗА РОЗДІЛОМ I

Розділ I навчального посібника присвячений теоретичним основам та практичній важливості стандарту ESCO (European Skills, Competences, Qualifications and Occupations) у контексті сучасного цифрового ринку праці. Стандарт ESCO слугує важливим інструментом для систематизації опису професій, навичок і кваліфікацій, забезпечуючи узгодженість між освітніми програмами, професійним навчанням та вимогами роботодавців.

Однією з визначальних характеристик ESCO є його здатність формувати структуру, яка інтегрує різні аспекти ринку праці. Система охоплює ключові функції гармонізації освітніх стандартів, підвищення адаптивності кадрів до економічних викликів та сприяння працевлаштуванню. Особливий акцент зроблено на багатомовності стандарту, що відкриває можливість його застосування у найрізноманітніших соціально-економічних умовах, забезпечуючи ефективність як на локальному, так і на міжнародному рівнях.

У посібнику акцентовано на ролі ESCO у впровадженні інноваційних підходів до управління трудовими ресурсами, розвитку програм перепідготовки кадрів та створення інструментів автоматизації процесів рекрутингу.

Поряд із загальним аналізом, у розділі наведено детальний опис професії 2521.3 «Розробник баз даних». Розглянуто основні характеристики професії, ключові вимоги до фахівців та їхні компетенції. Деталізовано необхідні технічні та аналітичні навички, серед яких знання принципів проектування баз даних, володіння мовою SQL, забезпечення інформаційної безпеки та використання сучасних платформ для управління даними. Висвітлюється значення цієї професії у контексті підтримки безперервного процесу цифрової трансформації, що є ключовим аспектом сучасної економіки.

РОЗДІЛ II

ТЕСТОВА ЧАСТИНА



2.1. Формат і зміст тестування

Формат тестування розроблений для комплексної оцінки знань і навичок фахівців, які працюють із базами даних. Тестові завдання охоплюють ключові аспекти професійної діяльності, що необхідні для успішного виконання практичних і аналітичних завдань.

Тестування містить 325 питань, кожне питання тесту пропонує чотири варіанти відповідей, серед яких лише один є правильним. Такий формат сприяє ефективному вимірюванню як теоретичної підготовки, так і практичних умінь. Структура забезпечує об'єктивність оцінювання та можливість автоматизованої перевірки результатів, що робить тестування зручним для використання в різних професійних і навчальних середовищах.

Тестові завдання адаптовані для широкої аудиторії: від початківців, які лише опановують основи роботи з базами даних, до досвідчених фахівців, зацікавлених у систематизації своїх знань і підтвердженні кваліфікації.

Тестування охоплює 13 тем, кожна з яких відповідає конкретній компетенції та зосереджується на окремих аспектах професійної діяльності, пов'язаній з базами даних.

Основні теми, які розглядаються в тестах:

- *інформаційна безпека баз даних:*
 - ✓ політики безпеки;
 - ✓ захист від SQL-ін'єкцій;
 - ✓ контроль доступу та шифрування;
 - ✓ аудит і моніторинг баз даних.
- *оптимізація ресурсів:*
 - ✓ розподіл обчислювальних ресурсів;
 - ✓ балансування навантаження у розподілених базах даних;
 - ✓ використання шардінгу та кешування;
 - ✓ інструменти для моніторингу продуктивності.
- *збір зворотного зв'язку:*
 - ✓ аналіз та обробка відгуків користувачів;
 - ✓ використання аналітичних інструментів;
 - ✓ методи підвищення точності та релевантності отриманих даних.

- *моделювання даних:*
 - ✓ розробка ER-діаграм;
 - ✓ нормалізація та денормалізація таблиць;
 - ✓ визначення ключів, зв'язків і кардинальності;
 - ✓ вибір оптимальної моделі для конкретного завдання.
- *резервне копіювання:*
 - ✓ типи резервного копіювання;
 - ✓ використання інструментів для створення резервних копій;
 - ✓ забезпечення цілісності даних та шифрування резервних копій.
- *визначення вимог замовника:*
 - ✓ формалізація бізнес-вимог;
 - ✓ функціональні та нефункціональні вимоги;
 - ✓ методи перевірки повноти вимог.
- *оцінка тривалості робіт:*
 - ✓ методи оцінювання часу;
 - ✓ врахування ризиків та можливих затримок;
 - ✓ планування і контроль виконання задач.
- *переклад технічних текстів:*
 - ✓ точність передачі технічної термінології;
 - ✓ робота з термінами, що відрізняються в різних системах;
 - ✓ локалізація і адаптація тексту до контексту.
- *робота з API баз даних:*
 - ✓ захист API від атак;
 - ✓ використання токенів та шифрування;
 - ✓ інтеграція баз даних з іншими системами.
- *моніторинг та аудит:*
 - ✓ використання інструментів для моніторингу;
 - ✓ аудит доступу та дій у базі даних;
 - ✓ визначення вузьких місць у продуктивності.
- *захист від DoS/DDoS атак:*
 - ✓ обмеження швидкості запитів;
 - ✓ захист серверів і оптимізація запитів.
- *створення звітності:*
 - ✓ аналіз даних для формування звітів;
 - ✓ інструменти для автоматизації звітності;
 - ✓ забезпечення візуалізації даних.
- *управління версіями баз даних:*
 - ✓ відстеження змін у структурах баз;
 - ✓ інструменти для версійного контролю;
 - ✓ міграція даних між версіями баз.

Результати тестування дозволяють оцінити рівень підготовки кандидата в кожній із зазначених компетенцій. Це сприяє ідентифікації сильних сторін та областей, які потребують вдосконалення.

Ключові результати тестування:

- *об'єктивна оцінка знань:*

Тести допомагають визначити, наскільки кандидат розуміє базові принципи та сучасні підходи до управління базами даних.

- *планування професійного розвитку:*

На основі отриманих результатів можна скласти індивідуальний план навчання для покращення компетенцій у слабких областях.

Наприкінці посібника знаходяться відповіді на тестові питання, що дозволяє учасникам перевірити правильність своїх відповідей.

2.2. Тестування компетенцій

Компетенція № 1 – Застосовувати політику інформаційної безпеки

1. Що є основною метою політики інформаційної безпеки у базах даних?

- 1) Збільшення швидкості виконання запитів
- 2) Захист даних від несанкціонованого доступу, втрати або модифікації
- 3) Створення резервних копій бази даних
- 4) Оптимізація структури бази даних

2. Що є ключовим фактором при впровадженні політики контролю доступу до баз даних?

- 1) Вибір операційної системи сервера
- 2) Використання принципу найменших привілеїв
- 3) Використання одного облікового запису для всіх користувачів
- 4) Перевірка цілісності SQL-запитів

3. Що з наведеного найкраще забезпечує захист даних під час передавання?

- 1) Використання захищених HTTP-з'єднань
- 2) Шифрування трафіку за допомогою протоколу TLS/SSL
- 3) Використання FTP для передачі файлів
- 4) Встановлення двофакторної аутентифікації отримувача

4. Що означає термін «SQL-ін'єкція»?

- 1) Вразливість бази даних, що дозволяє виконувати сторонні SQL-запити
- 2) Автоматична оптимізація SQL-запитів
- 3) Процес резервного копіювання бази даних
- 4) Технологія управління базами даних

5. Яка стратегія найкраще підходить для захисту від SQL-ін'єкцій?

- 1) Використання параметризованих запитів
- 2) Заборона використання SQL у додатку
- 3) Використання простого текстового введення у формах
- 4) Відключення всіх облікових записів з низьким рівнем доступу

6. Що є обов'язковим для забезпечення конфіденційності даних у базах даних?

- 1) Використання алгоритмів хешування для зберігання паролів
- 2) Налаштування зовнішніх індексів
- 3) Ручна модифікація таблиць
- 4) Доступ до бази даних через API

7. Що таке аудит баз даних у контексті інформаційної безпеки?

- 1) Процес перевірки даних на правильність
- 2) Моніторинг і запис дій, пов'язаних із доступом до бази даних
- 3) Автоматична оптимізація запитів до бази даних
- 4) Перевірка процесу міграції даних між базами

8. Який із методів допомагає запобігти несанкціонованому доступу до бази даних?

- 1) Встановлення брандмауера для бази даних
- 2) Шифрування системних журналів
- 3) Організація доступу до бази виключно для користувачів, які пройшли двофакторну ідентифікацію
- 4) Обмеження кількості IP-адрес, які мають доступ до бази даних

9. Що є основним ризиком при використанні слабких алгоритмів шифрування для захисту бази даних?

- 1) Зниження продуктивності бази даних
- 2) Ймовірність розшифрування даних зловмисником через відомі уразливості алгоритму
- 3) Ускладнення процесу резервного копіювання
- 4) Неможливість інтеграції з іншими системами

10. Чому ротація ключів шифрування є критично важливою для захисту бази даних?

- 1) Вона забезпечує відповідність системи міжнародним стандартам
- 2) Знижує вплив компрометації одного ключа на весь обсяг даних
- 3) Дозволяє уникнути проблем із продуктивністю системи
- 4) Виключає необхідність журналювання доступу до бази

11. Що є основним недоліком шифрування даних лише на рівні транспортного протоколу?

- 1) Складність налаштування протоколу
- 2) Відсутність захисту даних, які вже збережені в базі
- 3) Високе споживання обчислювальних ресурсів
- 4) Вимога постійного оновлення ключів

12. Чому критично важливо відокремлювати привілеї для читання, запису та адміністрування в базі даних?

- 1) Для зменшення затримок у виконанні SQL-запитів
- 2) Щоб обмежити потенційний вплив компрометації одного облікового запису
- 3) Для полегшення інтеграції з іншими системами
- 4) Щоб уникнути потреби в регулярному аудиті

13. Яка стратегія допомагає захистити дані від внутрішніх загроз?

- 1) Використання спільних облікових записів для всієї команди
- 2) Розподіл обов'язків та обмеження привілеїв
- 3) Моніторинг і аудит дій користувачів
- 4) Відокремлення робочого, тестового і розробницького середовищ

14. Яка стратегія найкраще підходить для запобігання атакам на API бази даних?

- 1) Використання змінних токенів для автентифікації без обмеження терміну дії
- 2) Використання токенів із обмеженим терміном дії для автентифікації
- 3) Заборона доступу до API для неавторизованих користувачів
- 4) Збереження токенів автентифікації у зашифрованому вигляді в захищених журналах

15. Що є ключовим фактором у захисті бази даних від атак типу DoS/DDoS?

- 1) Налаштування обмежень швидкості запитів до бази даних
- 2) Використання обмежень доступу для запитів із зовнішніх IP-адрес
- 3) Налаштування обмежень швидкості запитів для визначених типів транзакцій
- 4) Обмеження швидкості тільки для запитів із певних API-ключів

16. Який тип шифрування є найефективнішим для захисту конфіденційних даних у базі даних?

- 1) Симетричне шифрування із відкритими ключами
- 2) Асиметричне шифрування із довгими ключами

- 3) Симетричне шифрування з однаковими ключами для читання та запису
- 4) Шифрування зі змінними ключами без ротації

17. Що з наведеного є прикладом вразливості в системах керування базами даних?

- 1) Відсутність резервних копій
- 2) Неправильна конфігурація привілеїв користувачів
- 3) Використання непараметризованих запитів
- 4) Встановлення оновлень з затримкою

18. Що є головним ризиком використання довірених IP-адрес для контролю доступу?

- 1) Немоżliвість роботи з великою кількістю клієнтів
- 2) Компрометація внутрішніх мереж, які вважаються «довіреними»
- 3) Ускладнення процесу налаштування бази даних
- 4) Потреба у додаткових ресурсах для моніторингу

19. Що є критично важливим для забезпечення захисту журналів доступу до бази даних?

- 1) Шифрування журналів та обмеження доступу до них
- 2) Регулярна ротація журналів для обмеження доступу до них
- 3) Збереження журналів у зашифрованих архівах
- 4) Зберігання журналів у текстових файлах у хмарному сховищі

20. Що є основною метою періодичного оновлення політики інформаційної безпеки?

- 1) Врахування нових загроз і змін у технологіях
- 2) Підвищення продуктивності системи
- 3) Зменшення кількості авторизованих користувачів бази даних
- 4) Оптимізація витрат на адміністрування бази даних

21. Що є головною метою впровадження принципу «безпечного за замовчуванням» у базах даних?

- 1) Зменшення кількості авторизованих користувачів бази даних
- 2) Мінімізація ризику завдяки початковій конфігурації системи з високим рівнем безпеки
- 3) Оптимізація продуктивності бази даних
- 4) Автоматичне створення складних SQL-запитів для усунення людського фактору

22. Що є важливим для захисту від атаки «відмова в обслуговуванні»?

- 1) Встановлення обмежень на кількість запитів від одного джерела

- 2) Зашифроване журналювання доступу
- 3) Дозвіл з'єднань лише для користувачів з відомою IP-адресою
- 4) Додавання всіх запитів до однієї черги

23. Яка стратегія є ефективною для захисту бази даних від атак типу «експлуатація уразливостей ORM» (Object-Relational Mapping)?

- 1) Відмова від ORM та перехід до ручного написання SQL-запитів
- 2) Використання параметризованих запитів у ORM
- 3) Додавання фільтрів SQL на рівні клієнта
- 4) Зашифроване журналювання в ORM

24. Що означає термін «ізоляція середовищ» у контексті безпеки баз даних?

- 1) Розділення продуктивного, тестового та розробницького середовищ для мінімізації ризиків
- 2) Використання повнодискового шифрування
- 3) Використання одного сервера для всіх проєктів
- 4) Автоматичне оновлення даних між продуктивним і тестовим середовищами

25. Що є ключовим для забезпечення безпечного зберігання резервних копій бази даних?

- 1) Шифрування резервних копій перед їхнім збереженням
- 2) Розміщення копій бази в хмарному сховищі
- 3) Автоматичне видалення усіх старих копій після створення свіжої копії
- 4) Зашифроване журналювання в ORM під час створення резервних копій

Компетенція № 2 – Збалансовувати ресурси бази даних

1) Який підхід дозволяє оптимально розподілити обчислювальні ресурси у кластеризованій базі даних?

- 1) Використання одного вузла для всіх запитів
- 2) Впровадження розподілу навантаження
- 3) Архівування даних на зовнішніх носіях
- 4) Вимкнення кешування

2) Що є найефективнішим підходом для зменшення дискового вводу/виводу у базі даних?

- 1) Впровадження нормалізації таблиць
- 2) Налаштування пулу з'єднань
- 3) Використання кешування на рівні додатка
- 4) Увімкнення журналювання транзакцій

3) Як можна забезпечити стабільну роботу бази даних у середовищі зі змінними навантаженнями?

- 1) Налаштування обмежень на рівні таблиць
- 2) Автоматичне масштабування обчислювальних ресурсів
- 3) Ручна оптимізація запитів користувачів
- 4) Регулярна дефрагментація таблиць

4) Який метод найкраще підходить для зниження затримки у доступі до даних, які використовуються найчастіше?

- 1) Встановлення додаткових серверів
- 2) Використання кластеризованих індексів
- 3) Впровадження кешування у пам'яті
- 4) Агрегація великих таблиць

5) Що є ключовим фактором при балансуванні навантаження у розподіленій базі даних?

- 1) Максимізація транзакцій на секунду
- 2) Рівномірний розподіл запитів між вузлами
- 3) Зменшення розміру логів транзакцій
- 4) Використання резервних копій

6) Який інструмент найкраще підходить для моніторингу використання ресурсів бази даних?

- 1) Redis
- 2) Grafana
- 3) Microsoft Power BI
- 4) Google Analytics

7) Що таке шардінг (sharding) у контексті баз даних?

- 1) Процес створення резервних копій даних
- 2) Розподіл даних між кількома серверами
- 3) Використання агрегованих запитів для оптимізації
- 4) Оптимізація таблиць через нормалізацію

8) Який підхід дозволяє мінімізувати латентність у глобально розподілених базах даних?

- 1) Географічно близьке кешування
- 2) Зменшення кількості індексів
- 3) Резервування всіх даних на центральному сервері
- 4) Архівування старих даних

9) Що з наведеного найчастіше використовується для забезпечення високої доступності бази даних?

- 1) Використання реплікації даних
- 2) Нормалізація таблиць
- 3) Зменшення кількості запитів
- 4) Видалення старих даних

10) Що з переліченого найкраще підходить для балансування ресурсів у хмарних базах даних?

- 1) Використання вертикального масштабування
- 2) Розгортання локальних серверів
- 3) Впровадження горизонтального масштабування
- 4) Вимкнення механізму журналювання

11) Як можна уникнути конфліктів при одночасному доступі до даних?

- 1) Використання блокувань на рівні таблиць
- 2) Впровадження транзакцій з ізоляцією
- 3) Збільшення розміру бази даних
- 4) Регулярне резервне копіювання

12) Що дозволяє мінімізувати витрати пам'яті у базі даних?

- 1) Додавання вторинних індексів
- 2) Архівування старих даних
- 3) Використання матеріалізованих представлень
- 4) Перенесення бази на SSD-диск

13) Як знизити вплив інтенсивних читальних операцій на продуктивність?

- 1) Використання читацьких реплік
- 2) Оптимізація SQL-запитів

- 3) Додавання індексів
- 4) Зміна структури таблиць

14) Що з наведеного найкраще допоможе уникнути блокувань у транзакціях?

- 1) Використання коротких транзакцій
- 2) Використання каскадного видалення
- 3) Оптимізація структури таблиць
- 4) Видалення індексів

15) Який механізм дозволяє уникати дублювання даних при реплікації?

- 1) Використання унікальних індексів
- 2) Синхронна реплікація
- 3) Використання конфлікт-резолюції
- 4) Вимкнення журналювання

16) Що є найважливішим для підтримки продуктивності бази даних при великій кількості одночасних користувачів?

- 1) Підтримка паралелізму
- 2) Оптимізація агрегацій
- 3) Додавання індексів
- 4) Обмеження доступу

17) Що дозволяє зменшити обсяг дискового простору, зайнятого базою даних?

- 1) Використання компресії даних
- 2) Додавання індексів
- 3) Оптимізація мережевого з'єднання
- 4) Використання читацьких реплік

18) Як можна підвищити швидкість виконання складних аналітичних запитів?

- 1) Використання матеріалізованих представлень
- 2) Зменшення розміру таблиць
- 3) Впровадження нормалізації
- 4) Вимкнення кешування

19) Що дозволяє запобігти перевантаженню при високих обсягах записів у базі даних?

- 1) Використання черг повідомлень
- 2) Додавання індексів
- 3) Налаштування читацьких реплік
- 4) Архівування старих даних

20) Який інструмент найкраще підходить для виявлення «вузьких місць» у продуктивності бази даних?

- 1) EXPLAIN-план запиту
- 2) Використання матеріалізованих представлень
- 3) Реплікація даних
- 4) Впровадження нормалізації

21) Як найкраще уникнути проблем з конкуренцією при масштабуванні бази даних?

- 1) Використання горизонтального масштабування
- 2) Використання транзакційного ізолювання
- 3) Нормалізація таблиць
- 4) Видалення індексів

22) Як можна зменшити навантаження на дискову підсистему?

- 1) Використання in-мемоу баз даних
- 2) Архівування старих даних
- 3) Нормалізація таблиць
- 4) Використання читацьких реплік

23) Що дозволяє зменшити час виконання транзакцій?

- 1) Використання batch processing
- 2) Зменшення розміру бази даних
- 3) Відключення індексів
- 4) Нормалізація таблиць

24) Який механізм дозволяє уникнути перевантаження мережі у розподілених базах даних?

- 1) Використання географічно розподілених вузлів
- 2) Збільшення пропускної здатності
- 3) Архівування старих даних
- 4) Налаштування пулу з'єднань

25) Як можна зменшити латентність при глобальному використанні бази даних?

- 1) Використання географічно розподілених кешів
- 2) Оптимізація SQL-запитів
- 3) Видалення індексів
- 4) Архівування старих даних

Компетенція № 3 – Збирати відгуки клієнтів про додатки

1. Що є найкращим способом отримання зворотного зв'язку від користувачів мобільного додатку?

- 1) Телефонні опитування
- 2) Інтеграція анкети в інтерфейс додатку
- 3) Реклама в соціальних мережах
- 4) Надсилання SMS

2. Який інструмент є найбільш ефективним для аналізу відгуків у текстовій формі?

- 1) SQL
- 2) Sentiment Analysis
- 3) Google Forms
- 4) FTP

3. Який основний принцип слід враховувати при створенні анкети для збору зворотного зв'язку?

- 1) Робити анкети довгими для повнішого збору інформації
- 2) Використовувати технічні терміни для точності
- 3) Робити анкети короткими та простими
- 4) Додавати запитання, які стосуються особистості користувача, але не стосуються додатку

4. Який з інструментів збору відгуків підходить для аналізу активності користувачів у реальному часі?

- 1) Google Analytics
- 2) Canva
- 3) Oracle Database
- 4) Google Sheets

5. Що може вплинути на точність отриманих відгуків?

- 1) Мова опитування
- 2) Відсутність автоматизації
- 3) Час збору даних
- 4) Усі вищеперелічені

6. Яка метрика дозволяє оцінити загальну задоволеність клієнтів?

- 1) NPS (Net Promoter Score)
- 2) CTR (Click-Through Rate)
- 3) SLA (Service Level Agreement)
- 4) API (Application Programming Interface)

7. Яке з цих джерел може бути корисним для збору відгуків?

- 1) Магазины додатків (Google Play, App Store)
- 2) Форми для працівників компанії
- 3) Внутрішній Slack канал
- 4) Платіжні системи

8. Що найкраще мотивує користувачів залишати відгуки?

- 1) Низька ціна продукту
- 2) Гейміфікація
- 3) Відсутність технічної підтримки
- 4) Обов'язковий збір зворотного зв'язку

9. Який формат відгуків найкраще підходить для аналізу настроїв клієнтів?

- 1) Відгуки у форматі «п'ятизіркової шкали»
- 2) Текстові коментарі
- 3) Фото та відео контент
- 4) Лайки на платформі

10. Що потрібно врахувати при визначенні часу збору відгуків?

- 1) Географічну локацію користувачів
- 2) Найактивніший час використання додатку
- 3) Час роботи серверів
- 4) Часовий пояс команди, яка обробляє відгуки

11. Як можна підвищити релевантність отриманих відгуків?

- 1) Фільтрувати відгуки за допомогою спеціальних алгоритмів
- 2) Вимагати залишити контактну інформацію
- 3) Забороняти користувачам залишати анонімні відгуки
- 4) Примусово вимагати залишити відгук

12. Що є ключовим фактором у виборі платформи для збору зворотного зв'язку?

- 1) Простота використання
- 2) Інтеграція з іншими інструментами
- 3) Вартість підписки
- 4) Популярність бренду-розробника

13. Який метод найкраще підходить для збору відгуків у B2B сегменті?

- 1) Форми зворотного зв'язку на офіційних сайтах
- 2) Глибинні інтерв'ю з представниками компаній
- 3) Соціальні мережі
- 4) Публічні форуми

14. Яке програмне забезпечення підходить для інтеграції збору відгуків у додаток?

- 1) Google Forms
- 2) Appcues
- 3) Slack
- 4) Google Analysis

15. Що рекомендується робити з анонімними відгуками?

- 1) Не обробляти їх, бо вони не є достовірними
- 2) Аналізувати без упередження
- 3) Вимагати ідентифікації від користувача
- 4) Заборонити надсилання анонімних відгуків

16. Який із цих методів збору даних вимагає згоди користувача?

- 1) Аналіз історії переглядів
- 2) Використання Push-сповіщень для опитувань
- 3) Збір даних через API
- 4) Усі вищезгадані

17. Як найкраще налаштувати API для обробки відгуків?

- 1) Робити API відкритим для всіх
- 2) Використовувати JSON формат для передачі даних
- 3) Обмежувати кількість запитів від одного користувача
- 4) Зберігати дані лише в текстових файлах

18. Який із підходів є найбільш ефективним для збору відгуків від користувачів у режимі реального часу?

- 1) Створення веб-опитування
- 2) Вбудовані чати та форми у додатку
- 3) Надсилання електронних листів із запитом
- 4) Надсилання опитування в соціальні мережі

19. Що є важливим при виборі часу для надсилання запиту на відгук?

- 1) Надсилання запиту тільки у разі видалення додатку
- 2) Надсилання запиту після завершення важливої дії в додатку
- 3) Надсилання одразу після встановлення додатку
- 4) Надсилання тільки під час технічних збоїв

20. Як краще залучати користувачів до надання детальних відгуків?

- 1) Доповнювати питання картинками
- 2) Використовувати відкриті питання разом із рейтинговими шкалами

- 3) Запитувати відгуки лише у досвідчених користувачів
- 4) Пропонувати знижку лише за позитивний відгук

21. Що є перевагою інтеграції аналітичного модуля в додаток?

- 1) Зменшення розміру додатку
- 2) Миттєве збирання та аналіз даних про взаємодію користувачів
- 3) Зниження рівня анонімності користувачів
- 4) Полегшення роботи адміністратора

22. Який з форматів найменш ефективний для збору зворотного зв'язку?

- 1) Короткі опитування у додатку
- 2) Лонг-рід статті з формою в кінці
- 3) Автоматизовані поп-ап вікна із запитанням
- 4) Інтерактивні елементи, наприклад, емоджі-рейтинг

23. Як найкраще сегментувати відгуки для подальшого аналізу?

- 1) За віком користувачів
- 2) За типом відгуку (позитивний, негативний, нейтральний)
- 3) За датою написання відгуку
- 4) За кількістю символів у тексті

24. Який підхід допомагає перевірити достовірність отриманих відгуків?

- 1) Автоматична перевірка за унікальністю тексту
- 2) Перевірка локації користувача
- 3) Верифікація користувача через обліковий запис
- 4) Усі вищезгадані

25. Який з цих методів допоможе зменшити когнітивне навантаження на користувача під час надання зворотного зв'язку?

- 1) Додавання запитань двома мовами, обраними користувачем
- 2) Використання однозначних та зрозумілих запитань
- 3) Обмеження часу на відповідь
- 4) Використання спеціалізованих термінів

Компетенція № 4 – Створювати моделі даних

1. Що є основною метою створення моделі даних?

- 1) Визначення фізичних носіїв для даних
- 2) Побудова абстракції даних для їхньої організації та використання
- 3) Оптимізація обробки SQL-запитів
- 4) Захист бази даних від зовнішнього втручання

2. Який етап створення моделі даних вважається першим?

- 1) Нормалізація даних
- 2) Розробка ER-діаграми
- 3) Збір бізнес-вимог
- 4) Створення фізичної моделі даних

3. Що описує концептуальна модель даних?

- 1) Конкретну структуру таблиць у базі даних
- 2) Зв'язки між бізнес-об'єктами високого рівня
- 3) Реалізацію бази даних на фізичному рівні
- 4) Права доступу до даних

4. Який підхід використовується для запобігання надлишковості даних у моделі?

- 1) Денормалізація
- 2) Нормалізація
- 3) Індексація
- 4) Кешування

5. Що з наведеного є обов'язковою складовою логічної моделі даних?

- 1) Типи даних для атрибутів
- 2) Фізичне розташування серверів
- 3) Назви таблиць бази даних
- 4) План резервного копіювання

6. Яка методика допомагає визначити унікальність кожного запису у таблиці?

- 1) Створення зовнішніх ключів
- 2) Використання первинного ключа
- 3) Використання тригерів
- 4) Створення індексів

7. Що описує атрибут у моделі даних?

- 1) Тип бази даних
- 2) Властивість або характеристику сутності

- 3) Зв'язок між сутностями
- 4) Рівень доступу до даних

8. Який рівень моделювання даних включає фізичну структуру таблиць?

- 1) Концептуальний рівень
- 2) Логічний рівень
- 3) Фізичний рівень
- 4) Абстрактний рівень

9. Що таке денормалізація у моделюванні даних?

- 1) Процес розбиття таблиць для уникнення надлишковості
- 2) Процес об'єднання таблиць для підвищення продуктивності
- 3) Створення зовнішніх ключів
- 4) Додавання користувачів до бази даних

10. Яка сутність може бути слабкою?

- 1) Та, що має власний первинний ключ
- 2) Та, що не може існувати без зв'язку з іншою сутністю
- 3) Та, що має більше атрибутів, ніж інші
- 4) Та, що має зв'язок «багато до багатьох»

11. Що забезпечує зовнішній ключ у базі даних?

- 1) Індексацію таблиці
- 2) Визначення зв'язків між таблицями
- 3) Унікальність записів
- 4) Захист даних від несанкціонованого доступу

12. Що визначає кардинальність у моделі даних?

- 1) Кількість сутностей у базі даних
- 2) Кількість записів у таблиці
- 3) Типи зв'язків між сутностями
- 4) Кількість атрибутів у сутності

13. Що є головною ознакою третьої нормальної форми?

- 1) Відсутність транзитивних залежностей між атрибутами
- 2) Використання зовнішніх ключів
- 3) Обов'язковий первинний ключ
- 4) Поділ таблиць на менші

14. Що є результатом моделювання даних?

- 1) Створення SQL-запитів
- 2) Документована структура для створення бази даних
- 3) Впровадження програмного забезпечення
- 4) Інтеграція хмарних сервісів

15. Що описує модель «зірка» у сховищі даних?

- 1) Таблиці повністю незалежні одна від одної
- 2) Одна центральна таблиця пов'язана з іншими через зовнішні ключі
- 3) Дані зберігаються у вигляді дерева
- 4) Всі таблиці мають однакову кількість атрибутів

16. Що таке атрибути, які мають «складені» значення?

- 1) Атрибути, які мають більше одного компонента
- 2) Атрибути, що є унікальними для кожного запису, але не є частиною зовнішнього ключа
- 3) Атрибути, які використовуються лише в індексації
- 4) Атрибути, що визначають первинний ключ

17. Що є основною характеристикою фізичної моделі даних?

- 1) Визначення структури таблиць, індексів та типів даних
- 2) Створення абстракцій для бізнес-аналітики
- 3) Використання UML-діаграм для зв'язків
- 4) Опис ідентифікаторів користувачів

18. Що таке «суперключ» у реляційній моделі?

- 1) Унікальна комбінація атрибутів, яка ідентифікує запис
- 2) Ключ, який складається з одного атрибута
- 3) Зовнішній ключ для посилань на інші таблиці
- 4) Атрибут, який містить складені значення

19. Що є основною метою використання індексів у базах даних?

- 1) Зменшення кількості записів у таблиці
- 2) Прискорення пошуку та обробки даних
- 3) Зменшення розміру бази даних
- 4) Забезпечення транзакційної цілісності

20. Що таке складений первинний ключ?

- 1) Первинний ключ, який об'єднує кілька атрибутів
- 2) Ключ, який використовується для зовнішніх зв'язків
- 3) Ключ, який містить інклюзивні залежності
- 4) Ключ, який містить транзитивні залежності

21. Який тип діаграми найчастіше використовується для візуалізації моделі даних?

- 1) Гантова діаграма
- 2) ER-діаграма
- 3) Діаграма розсіювання
- 4) Технологічна мапа

22. Що є основною метою створення денормалізованих таблиць?

- 1) Зменшення об'єму пам'яті, що використовується базою даних
- 2) Підвищення швидкості запитів до бази даних
- 3) Уникнення транзитивних залежностей
- 4) Підвищення рівня абстракції моделі

23. Що таке транзитивна залежність у реляційній моделі?

- 1) Залежність між сутностями різних таблиць
- 2) Непряма залежність між атрибутами у таблиці
- 3) Інша назва зв'язку «багато до одного»
- 4) Інша назва зв'язку «один до багатьох»

24. Яка основна характеристика моделі даних типу «сніжинка»?

- 1) Розгалужена структура з великою кількістю таблиць і зв'язків
- 2) Єдина таблиця для всіх даних
- 3) Лінійна структура для зберігання історичних даних
- 4) Таблиці без зовнішніх ключів

25. Що описує «атрибут за замовчуванням» у моделі даних?

- 1) Тип значення, яке буде автоматично записане за відсутності введення даних
- 2) Обов'язковий атрибут для всіх записів
- 3) Атрибут, який завжди є частиною первинного ключа
- 4) Атрибут, який може бути видалений без впливу на цілісність моделі

Компетенція № 5 – Оцінювати тривалість робіт

1. Що є ключовим фактором при визначенні тривалості етапу нормалізації бази даних?

- 1) Кількість таблиць у початковій моделі
- 2) Тип обраної СУБД
- 3) Відповідність серверного обладнання
- 4) Розмір хмарного сховища, де зберігається база

2. Яка техніка оцінювання тривалості передбачає використання трьох значень: оптимістичного, песимістичного та найімовірнішого?

- 1) Метод критичного шляху
- 2) PERT (Program Evaluation and Review Technique)
- 3) Парето-аналіз
- 4) Метод Монте-Карло

3. Що з наведеного найбільше впливає на оцінку часу, потрібного для налаштування індексів у базі даних?

- 1) Кількість записів у таблиці
- 2) Тип обладнання розробника
- 3) Операційна система сервера
- 4) Локація серверної інфраструктури

4. Яка з цих методологій проєктного менеджменту найбільше допомагає в оцінці тривалості етапів розробки?

- 1) Agile
- 2) Waterfall
- 3) Kanban
- 4) PRINCE2

5. Який етап розробки бази даних зазвичай потребує найбільше часу?

- 1) Збір вимог
- 2) Проєктування логічної моделі даних
- 3) Тестування продуктивності
- 4) Резервне копіювання

6. Що є основним недоліком використання методу аналогій для оцінювання тривалості робіт?

- 1) Висока складність аналізу
- 2) Ризик недооцінки унікальних факторів проєкту
- 3) Занадто точні оцінки, які складно змінити
- 4) Повна залежність від клієнта

7. Який з наступних аспектів є ключовим при використанні методу критичного шляху для оцінки тривалості проєкту?

- 1) Ідентифікація етапів, що мають найбільший вплив на загальну тривалість проєкту
- 2) Визначення етапів, які можна виконувати паралельно, щоб зменшити тривалість проєкту
- 3) Оцінка найкоротшого часу для кожного етапу проєкту
- 4) Рівномірний розподіл завдань між усіма членами команди для оптимізації часу

8. Який метод є найбільш ефективним для оцінки часу, необхідного для міграції даних між різними СУБД?

- 1) Аналіз попередніх міграцій за допомогою історичних даних
- 2) Використання методу Монте-Карло для прогнозування часу
- 3) Визначення часу на основі кількості таблиць, що потрібно перенести
- 4) Оцінка на основі часу, затраченого на аналогічні завдання в іншій базі даних

9. Що може призвести до значного збільшення часу виконання запитів під час тестування?

- 1) Відсутність належної оптимізації запитів
- 2) Використання сторонніх інструментів моніторингу
- 3) Надмірна нормалізація даних
- 4) Невідповідність формату даних вимогам клієнта

10. Що з наведеного найчастіше уповільнює етап інтеграції бази даних?

- 1) Відсутність документації по API
- 2) Некоректна структура таблиць
- 3) Неправильне форматування індексів
- 4) Занадто велика кількість таблиць в базі

11. Що є важливим етапом після первинного оцінювання тривалості робіт?

- 1) Фіксування термінів у контракті
- 2) Порівняння із загальноприйнятими стандартами
- 3) Проведення ревізії оцінок з урахуванням ризиків
- 4) Вибір інструменту для відстеження прогресу

12. Який з факторів може бути суттєвою причиною зміщення оцінки тривалості тестування?

- 1) Нестача тестових даних

- 2) Наявність кількох середовищ розгортання
- 3) Неоптимальна конфігурація сервера
- 4) Використання некоректних тестових сценаріїв

13. Що найкраще допомагає врахувати непередбачені затримки при оцінюванні тривалості робіт?

- 1) Використання коефіцієнта запасу часу
- 2) Врахування історичних даних проєкту
- 3) Вибір максимального з песимістичних оцінок
- 4) Автоматизація всіх процесів

14. Який аспект найскладніше врахувати при оцінці тривалості масштабування бази даних?

- 1) Зростання навантаження під час масштабування
- 2) Кількість запитів під час масштабування
- 3) Збільшення кількості доступних серверів без додаткової оптимізації
- 4) Форма резервного копіювання

15. Який метод найкраще підходить для розрахунку оптимального часу виконання запитів?

- 1) Аналіз вузьких місць у плані виконання
- 2) Використання динамічних індексів
- 3) Розподіл таблиць за типами транзакцій
- 4) Виконання та аналіз запитів на тестових даних

16. Який фактор є критичним для оцінки тривалості інтеграції бази даних із зовнішніми системами?

- 1) Формат обміну даними між системами
- 2) Фізичне розташування серверів бази даних
- 3) Тип ліцензії зовнішньої системи
- 4) Типи запитів, які використовуються для інтеграції

17. Що допоможе розрахувати час, необхідний для інтеграції нових даних у вже наявну базу даних?

- 1) Оцінити кількість необхідних змін у структурі таблиць
- 2) Розрахувати час, залежно від складності алгоритмів перевірки даних
- 3) Розрахувати час на основі кількості запитів, що потрібно виконати
- 4) Розрахувати час на основі кількості рядків, які необхідно додати до таблиць

18. Яка техніка дозволяє скоротити невизначеність у розрахунках тривалості великих проєктів?

- 1) Розбиття завдань на менші підзадачі

- 2) Додаткове врахування історичних даних
- 3) Виключення часових буферів
- 4) Вибір мінімального часу серед всіх оцінок

19. Який з наступних факторів має найбільший вплив на ефективність виконання складних SQL-запитів у великій базі даних?

- 1) Тип і кількість індексів у таблицях
- 2) Розмір таблиць, що обробляються
- 3) Наявність кешування результатів запитів
- 4) Налаштування транзакційної ізоляції в СУБД

20. Який фактор може значно вплинути на точність оцінки часу при впровадженні нового функціоналу у базу даних?

- 1) Складність інтеграції нового функціоналу з наявними модулями
- 2) Кількість користувачів, що будуть взаємодіяти з базою
- 3) Розмір вихідних даних, які потрібно обробити
- 4) Час, необхідний для навчання команди адміністраторів бази даних

21. Який з факторів найбільше впливає на тривалість етапу тестування працездатності бази даних?

- 1) Складність запитів, які виконуються
- 2) Час роботи серверів
- 3) Кількість рядків у таблицях
- 4) Вибраний тип дискової пам'яті

22. Який інструмент найкраще підходить для оцінки та візуалізації тривалості робіт?

- 1) Gantt Chart
- 2) SQL Profiler
- 3) UML-діаграми
- 4) GitHub Issues

23. Яка головна перевага використання методу «знизу вгору» в оцінці тривалості робіт?

- 1) Висока деталізація плану і точність оцінки
- 2) Простота у визначенні загальної тривалості проєкту
- 3) Можливість уникнути розрахунків для невеликих завдань
- 4) Автоматичне врахування всіх ризиків

24. Який метод оцінки тривалості виконання завдання є найефективнішим для проєктів з високим рівнем невизначеності?

- 1) Метод аналогії, використовуючи дані попередніх проєктів
- 2) Метод тріангуляції, враховуючи різні джерела даних

- 3) Метод експертних оцінок із суб'єктивною оцінкою команди
- 4) Метод Дельфі, заснований на незалежних відгуках групи експертів

25. Яка стратегія є найбільш корисною для оцінки часу виконання складних робіт, що мають багато змінних і невизначеностей?

- 1) Побудова детального плану з конкретними термінами для кожної задачі
- 2) Розрахунок часу на основі максимально можливого ресурсу
- 3) Оцінка часу на основі середнього часу виконання завдання у команді
- 4) Розбиття задачі на частини і оцінка кожної окремо з урахуванням можливих змін

Компетенція № 6 – Визначати вимоги замовника робіт

1. Що є основним результатом етапу збору вимог?

- 1) ER-діаграма бази даних
- 2) Документ із формалізованими бізнес-вимогами
- 3) Звіт про потенційні обсяги даних в базі
- 4) Програмний код для створення таблиць

2. Який із підходів дозволяє найточніше формалізувати вимоги замовника?

- 1) Використання діаграм прецедентів
- 2) Пряме створення SQL-таблиць на основі розмов із замовником
- 3) Використання стандартизованої моделі
- 4) Тестування бази на даних замовника

3. Що є ключовим фактором для визначення функціональних вимог до бази даних?

- 1) Потенційний обсяг даних в базі
- 2) Кількість співробітників, які будуть працювати із базою
- 3) Очікувані бізнес-процеси замовника
- 4) Часові рамки реалізації проєкту

4. Яка техніка найкраще підходить для виявлення прихованих вимог замовника?

- 1) Аналіз конкурентних рішень
- 2) Використання детальних анкет
- 3) Створення прототипу бази даних
- 4) Виконання стрес-тестів на початкових етапах

5. Що найчастіше стає причиною пропуску критичних вимог?

- 1) Відсутність тестування системи
- 2) Неповне розуміння бізнес-процесів замовника
- 3) Використання стандартних методів моделювання
- 4) Високий рівень деталізації бази даних

6. Що є ключовим етапом перевірки отриманих вимог?

- 1) Затвердження документації із замовником
- 2) Виконання SQL-запитів для перевірки моделі
- 3) Налаштування індексів бази даних
- 4) Налаштування серверного середовища

7. Що є основною метою функціонального моделювання під час визначення вимог?

- 1) Оптимізація продуктивності бази даних

- 2) Формалізація та візуалізація процесів, які база має підтримувати
- 3) Створення фізичних таблиць і атрибутів
- 4) Підготовка інструментів для розгортання

8. Яка з методик найкраще підходить для збору вимог у великому проєкті?

- 1) SCRUM
- 2) Водоспадна модель (Waterfall)
- 3) Пряме моделювання на основі SQL
- 4) Аналіз великих даних (Big Data Analysis)

9. Що з наведеного є нефункціональною вимогою?

- 1) Час відповіді бази даних на запит
- 2) Список сутностей і зв'язків
- 3) Тип даних для кожного атрибута
- 4) Права доступу для користувачів

10. Який підхід є найбільш ефективним для визначення ролей і рівнів доступу до бази даних на етапі проєктування?

- 1) Проведення опитування серед майбутніх користувачів для визначення їхніх потреб у доступі
- 2) Використання стандартних ролей, визначених в СУБД
- 3) Аналіз поточної інфраструктури компанії замовника та її політик доступу
- 4) Тестування системи доступу після реалізації бази даних

11. Що є критично важливим для оцінки вимог до обсягу даних?

- 1) Кількість користувачів системи
- 2) Поточний рівень завантаженості сервера
- 3) Максимальний очікуваний обсяг транзакцій
- 4) Тип інтерфейсу для користувачів

12. Який із зазначених інструментів використовується для документування вимог замовника?

- 1) MS Visio
- 2) MySQL Workbench
- 3) Jira
- 4) Python

13. Яка техніка дозволяє визначити, наскільки вимоги замовника є повними?

- 1) Аналіз на відповідність SMART-критеріям
- 2) Побудова фізичної моделі бази даних
- 3) Використання тестових даних
- 4) Проведення Code Review

14. Який із підходів допомагає уникнути конфліктів між функціональними і нефункціональними вимогами?

- 1) Розробка детальної технічної специфікації перед початком роботи
- 2) Використання хмарних платформ для прототипування
- 3) Оптимізація SQL-запитів на початкових етапах
- 4) Автоматичне генерування документації за допомогою ШІ

15. Який спосіб перевірки вимог є найбільш надійним для великих проєктів?

- 1) Рецензія вимог за участю всіх стейкхолдерів
- 2) Тестування бази даних після її розробки
- 3) Використання шаблонів стандартних баз даних
- 4) Опитування користувачів готової бази даних

16. Що є критичним для визначення вимог щодо безпеки бази даних?

- 1) Наявність політики управління паролями
- 2) Фізичне розташування серверів
- 3) Кількість індексів у таблицях
- 4) Складність SQL-запитів

17. Який підхід дозволяє мінімізувати ризик пропуску важливих вимог?

- 1) Застосування методів ітеративного збору вимог із постійною перевіркою та уточненням
- 2) Інтеграція збору вимог у процес тестування з періодичними оглядами та змінами
- 3) Використання шаблонів стандартних вимог до баз даних
- 4) Використання автоматизованих інструментів для виявлення незадокументованих вимог на основі історичних даних

18. Що є основною перевагою використання прототипів для збору вимог?

- 1) Швидкість створення остаточної бази даних
- 2) Зменшення кількості документів для затвердження
- 3) Візуалізація ідеї замовника та зменшення ризику нерозуміння
- 4) Мінімізація витрат на тестування

19. Який документ є обов'язковим результатом збору вимог для бази даних?

- 1) Технічна документація із затвердженими вимогами
- 2) SQL-скрипт для створення таблиць
- 3) ER-діаграма бази даних
- 4) Діаграма класів у форматі UML

20. Який із підходів допомагає визначити нефункціональні вимоги замовника?

- 1) Вивчення наявних систем замовника
- 2) Оптимізація моделі бази даних після тестування
- 3) Використання методу FURPS
- 4) Автоматична оцінка за допомогою спеціальних SQL-запитів

21. Який тип артефактів найкраще пояснює складні функціональні вимоги?

- 1) Прототипи та сценарії використання
- 2) Таблиці з атрибутами даних
- 3) Логічні моделі даних
- 4) Тестові SQL-запити

22. Що є важливим етапом у роботі з нефункціональними вимогами?

- 1) Затвердження вимог із замовником після завершення розробки
- 2) Розробка тестової бази даних без урахування обмежень продуктивності
- 3) Визначення метрик, які дозволять оцінити їхнє виконання
- 4) Затвердження технічних параметрів бази

23. Який із факторів найсильніше впливає на обсяг збору вимог?

- 1) Масштаб проєкту і його бізнес-цілі
- 2) Доступність технічної документації
- 3) Час, відведений на тестування
- 4) Кількість залучених сторонніх постачальників

24. Що є ключовим для визначення вимог до звітності бази даних?

- 1) Розуміння типів звітів і періодичності їхнього формування
- 2) Використання аналітичного модуля бази даних
- 3) Інтеграція бази даних із сторонніми системами
- 4) Тестування інструментів візуалізації

25. Яка вимога є важливою для забезпечення масштабованості бази даних при зростанні навантаження?

- 1) Інтеграція з хмарними сервісами для автоматичного розширення ресурсів
- 2) Наявність резервних копій для захисту даних
- 3) Оптимізація індексів та нормалізація структури таблиць
- 4) Забезпечення високої швидкості запитів незалежно від обсягу даних

Компетенція № 7 – Перекладати технічні тексти**1. Що є серйозною помилкою під час перекладу технічної документації?**

- 1) Втрата точності термінів і смислових нюансів
- 2) Втрата оригінального форматування тексту
- 3) Допускання в тексті тавтологій
- 4) Переклад коментарів у коді без пояснень

2. Що є важливим при перекладі текстів, що містять SQL-запити?

- 1) Переклад коментарів і пояснень без зміни синтаксису SQL
- 2) Зміна SQL-коду для адаптації до локальних стандартів
- 3) Переклад усіх операторів SQL мовою перекладу
- 4) Переклад тексту та назв таблиць і сутностей для відповідності мові перекладу

3. Як слід працювати з термінами, що відрізняються залежно від програмного забезпечення (наприклад, SQL Server та Oracle)?

Приклад: Українською мовою слово «схема» – логічна структура, що організовує об'єкти бази даних (таблиці, індекси, процедури тощо).

Англійською мовою:

- *в SQL Server: Database*
- *в Oracle: Schema*

- 1) Застосовувати термінологію конкретного ПЗ, що описується у тексті
- 2) Використовувати універсальні терміни незалежно від контексту
- 3) Змінювати терміни на основі власного досвіду перекладача
- 4) Замінювати такі спеціалізовані терміни на загальноновживані аналоги

4. Що є ключовим фактором при перекладі документації для інтеграції баз даних?

- 1) Точний переклад інтерфейсів програмування API
- 2) Точний переклад технічних інструкцій
- 3) Ігнорування помилок оригінального тексту, якщо вони є
- 4) Оптимізація структури документа

5. Що є головною складністю при перекладі інструкцій для адміністрування баз даних?

- 1) Відсутність загальноприйнятої термінології
- 2) Збереження логічного порядку кроків і структур
- 3) Наявність багатьох синонімів для одного поняття
- 4) Використання перекладацьких автоматизованих інструментів

6. Що означає термін «локалізація» у контексті технічного перекладу?

- 1) Точний переклад без адаптації до локального контексту
- 2) Адаптація несуттєвої частини технічного тексту до культурних і мовних особливостей
- 3) Використання стандартних фраз і виразів
- 4) Заміна технічних термінів на менш точні слова для кращого сприйняття користувачами

7. Який із цих елементів технічного тексту найчастіше вимагає локалізації при перекладі?

- 1) Формати дат і чисел, одиниці вимірювання
- 2) SQL-запити
- 3) Типи змінних у коді
- 4) Список функціональних вимог

8. Яка з цих програм є популярною для перекладу технічних текстів?

- 1) Trados
- 2) MS Sway
- 3) GIMP
- 4) Blender

9. Як перекладач має діяти, якщо термін у технічному тексті не має прямого аналога в мові перекладу?

- 1) Залишити термін мовою оригіналу
- 2) Замінити термін загальним словом, що передає приблизний сенс
- 3) Пояснити термін у виносці технічною мовою перекладу
- 4) Пояснити термін у дужках художнім описом мовою перекладу

10. Який із методів найкраще допомагає валідувати переклад технічного тексту?

- 1) Рецензія тексту технічними експертами
- 2) Використання автоматичних перекладачів, щоб уникнути помилок перекладача
- 3) Додавання пояснень у тексті
- 4) Перевірка правильності перекладу назв розділів та підрозділів

11. Відповідно до положень Стандарту ISO 17100:2015 Translation services – Requirements for translation services, яке дієслово вказує на «вимоги, яких необхідно суворо дотримуватися і від яких відхилення не допускаються»?

- 1) Shall
- 2) Should

- 3) May
- 4) Can

12. Відповідно до положень Стандарту ISO 17100:2015 Translation services – Requirements for translation services, що є основною функцією автоматизованого перекладу?

- 1) Використання програмних засобів для підтримки та оптимізації роботи перекладача
- 2) Повна заміна перекладача при обробці текстів
- 3) Генерування перекладів з використанням ШІ
- 4) Автоматична заміна термінів, що не мають прямих аналогів

13. Які інструменти краще підходять для автоматизованого перекладу технічних текстів?

- 1) NMTs (Neural Machine Translation Systems) – системи нейронного машинного перекладу
- 2) TEnTs (Translation Environment Tools) – засоби перекладу або середовища перекладу
- 3) CATs (Computer-Aided Transcription) – комп'ютерні програми для автоматичної транскрипції тексту
- 4) ASRs (Automatic Speech Recognition) – системи автоматичного розпізнавання мовлення

14. Як перекладач має діяти, якщо у вихідному тексті виявлено очевидну помилку?

- 1) Перекласти помилку, як є
- 2) Залишити помилку, але в примітках вказати на це
- 3) Уточнити правильність тексту у замовника
- 4) виправити помилку без повідомлення

15. Як треба перекладати терміни, які мають кілька значень у різних контекстах?

- 1) Використати значення, яке перекладачу здається зрозумілішим
- 2) Проаналізувати контекст і вибрати одне відповідне значення
- 3) Замінити термін загальним описом мовою оригіналу
- 4) Вказати у примітках, що термін також має інші значення

16. Який з текстів вимагає найвищої точності при перекладі?

- 1) Інструкції для кінцевих користувачів
- 2) Звіти про продуктивність баз даних
- 3) Юридична документація
- 4) Маркетинговий опис функціоналу бази даних

17. Як слід обробляти технічні скорочення, що можуть бути невідомі цільовій аудиторії?

- 1) Додати їхню повну розшифровку під час першого використання у тексті
- 2) Надавати пояснення скорочень в окремому глосарії або додатку до тексту
- 3) Вказати розшифровку у примітці, не змінюючи основного тексту
- 4) Замінити скорочення на повний термін у кожному випадку його використання

18. Яка головна роль глосарію під час роботи з технічними текстами, особливо в умовах великого обсягу термінології?

- 1) Забезпечує термінологічну єдність та точність перекладу, мінімізуючи розбіжності у використанні термінів
- 2) Сприяє стандартизації термінів і дозволяє замінювати застарілі або некоректні слова на більш точні
- 3) Допомогає уникати неправильного трактування термінів завдяки їхньому однозначному визначенню
- 4) Виконує функцію єдиного довідника, що усуває потребу в додаткових консультаціях з експертами

19. Як в глосарії прийнято розташовувати терміни для забезпечення якості технічного перекладу?

- 1) В алфавітному порядку мови перекладу, щоб терміни було легко знайти
- 2) За порядком їхнього використання у тексті для логічної структури
- 3) За рівнем складності: від простих до найскладніших термінів
- 4) Групувати терміни за категоріями (наприклад, «запити», «сховища», «таблиці») для тематичної організації

20. Чому важливо розрізняти галузеву та корпоративну термінологію під час перекладу технічної документації?

- 1) Галузева термінологія є загальноприйнятою, тоді як корпоративна може бути унікальною для компанії
- 2) Галузева термінологія завжди є точнішою за корпоративну
- 3) Галузева термінологія використовується лише в академічних текстах, а корпоративна – у технічних інструкціях
- 4) Галузева термінологія не потребує узгодження з замовником, а корпоративна – потребує.

21. Що таке термінологічна інтерференція у технічному перекладі?

- 1) Використання неправильного значення терміна через вплив вихідної мови

- 2) Використання двох різних термінів для одного поняття в тексті
- 3) Переклад термінів без врахування культурних особливостей
- 4) Заміна термінів на непов'язані аналогії

22. Як омоніми можуть впливати на точність перекладу технічного?

- 1) Вони можуть призвести до неоднозначного трактування через однакову форму написання, але відмінності у значенні залежно від контексту
- 2) Вони зазвичай мають однакове значення в різних мовах, що полегшує переклад, але може ускладнити узгодження стилю
- 3) Вони можуть бути замінені іншими словами без суттєвого впливу на розуміння тексту, що створює виклики для точності перекладу
- 4) Вони ускладнюють роботу з термінологією та вимагають зміни структури речення для уникнення двозначностей

23. Яке з наведених слів може викликати складнощі під час перекладу через свою омонімію?

- 1) Log – колода (дерево)
- 2) File – картотека (паперова система)
- 3) Record – грамофонна платівка
- 4) Усі наведені варіанти

24. Який підхід до перекладу омонімів є найбільш ефективним у технічному тексті?

- 1) Визначення значення омоніма на основі його контексту та функції у тексті
- 2) Використання універсального перекладу, який підходить для всіх значень
- 3) Замінювати омоніми на складні описові конструкції
- 4) Замінювати омоніми на інші спеціальні слова – омографи

25. Що є найкращим підходом для перекладу неологізмів, які ще не мають усталеного перекладу в цільовій мові?

- 1) Транслітерувати термін та додати пояснення його значення у примітці
- 2) Використати тимчасовий переклад, узгоджений з експертом
- 3) Створити описовий переклад, що розширює текст і пояснює термін у кожному випадку його використання
- 4) Замінити термін на наявний термін з іншої, близької технічної галузі

Компетенція № 8 – Створювати резервні копії

1. Який тип резервного копіювання виконує копію лише тих даних, які змінилися після останнього резервного копіювання?

- 1) Повне копіювання (Full Backup)
- 2) Інкрементальне копіювання (Incremental Backup)
- 3) Диференційне копіювання (Differential Backup)
- 4) Логічне копіювання (Logical Backup)

2. Що є найбільш критичним під час вибору сховища для резервних копій?

- 1) Локалізація серверів
- 2) Надійність та доступність сховища
- 3) Тип файлової системи
- 4) Швидкість інтернет-з'єднання

3. Який із методів резервного копіювання бази даних мінімізує вплив на продуктивну систему?

- 1) Логічне копіювання
- 2) Копіювання у стані спокою (Cold Backup)
- 3) Гаряче копіювання (Hot Backup)
- 4) Ручне копіювання таблиць

4. Що є ключовим фактором для оцінки ефективності стратегії резервного копіювання?

- 1) Кількість створених копій
- 2) RTO (Recovery Time Objective) і RPO (Recovery Point Objective)
- 3) Тип операційної системи
- 4) Локалізація сервера та користувачів

5. Який тип резервного копіювання міститиме лише зміни, зроблені після останнього повного копіювання?

- 1) Інкрементальне копіювання
- 2) Диференційне копіювання
- 3) Фізичне копіювання
- 4) Реплікація даних

6. Який із наведених інструментів використовується для створення резервних копій баз даних MySQL?

- 1) mysqldump
- 2) pg_dump
- 3) MongoDump
- 4) SQL Server Management Studio

7. Який із наведених інструментів використовується для створення резервних копій баз даних Oracle?

- 1) pg_dump
- 2) RMAN (Recovery Manager)
- 3) Data Pump Export (expdp)
- 4) Navicat for Oracle

8. Що може спричинити помилку під час створення резервної копії бази даних?

- 1) Відсутність достатнього місця у сховищі
- 2) Відсутність індексів у таблицях
- 3) Використання застарілих SQL-запитів
- 4) Наявність транзакцій у таблицях

9. Який тип резервного копіювання передбачає копіювання лише даних, що змінилися, але не включає структуру таблиць?

- 1) Логічне копіювання
- 2) Фізичне копіювання
- 3) Диференційне копіювання
- 4) Інкрементальне копіювання

10. Яке з цих рішень допомагає автоматизувати процес створення резервних копій?

- 1) Cron Job у Linux
- 2) Ansible
- 3) FTP-клієнт
- 4) Zabbix

11. Що означає термін snapshot у контексті резервного копіювання?

- 1) Створення моментальної копії стану бази даних
- 2) Автоматичне відновлення з резервної копії
- 3) Створення фізичної копії всіх даних
- 4) Тестування даних перед реплікацією даних

12. Який підхід є найкращим для забезпечення цілісності резервної копії?

- 1) Використання контрольних сум для перевірки даних
- 2) Використання технології «знімок» snapshot для миттєвого збереження стану даних
- 3) Застосування інструментів дедуплікації для зменшення розміру копії
- 4) Додавання метаданих про резервну копію до спеціалізованого журналу змін

13. Що є критичним для безпеки резервних копій?

- 1) Шифрування резервних копій перед збереженням
- 2) Використання одного сховища для всіх копій
- 3) Регулярне переміщення копій до офлайн-сховища для додаткового захисту
- 4) Обмеження доступу до резервних копій через багаторівневу автентифікацію

14. Що означає термін «реплікація даних»?

- 1) Відновлення даних із резервної копії
- 2) Автоматичне створення копій у реальному часі між кількома серверами
- 3) Логічне копіювання бази даних
- 4) Злиття кількох баз даних в одну

15. Що є основною перевагою інкрементального резервного копіювання?

- 1) Зменшення часу, необхідного для створення копії
- 2) Повна незалежність від попередніх копій
- 3) Усі дані зберігаються в одному файлі
- 4) Виключення будь-яких повторюваних даних

16. Що може бути причиною пошкодження резервної копії?

- 1) Збої в роботі апаратного забезпечення під час створення копії
- 2) Надмірна нормалізація таблиць
- 3) Несумісність версій програмного забезпечення, яке створює та відновлює резервні копії
- 4) Пошкодження файлів через неправильне завершення сесії логічного копіювання

17. Що означає термін «ретенційна політика» у контексті резервного копіювання?

- 1) Політика обмеження доступу до бази даних
- 2) Визначення часу зберігання резервних копій
- 3) Вибір типу резервного копіювання
- 4) Налаштування частоти створення копій

18. Який зі способів є найбільш безпечним для передавання резервних копій?

- 1) Використання протоколу SFTP
- 2) Використання протоколу HTTP
- 3) Використання зовнішнього жорсткого диска з симетричним шифруванням даних
- 4) Передавання через захищені поштові сервіси

19. Що є важливим для регулярного тестування резервних копій?

- 1) Перевірка можливості повного відновлення даних
- 2) Оновлення всієї інфраструктури резервного копіювання
- 3) Видалення старих тестових копій
- 4) Створення нових таблиць для тестування

20. Яка перевага гарячого резервного копіювання?

- 1) Можливість копіювання без перерви у роботі системи
- 2) Зменшення обсягу копії
- 3) Створення резервних копій на фізичному рівні
- 4) Видалення даних, що дублюються

21. Що може допомогти зменшити розмір резервної копії?

- 1) Компресія даних під час створення копії
- 2) Використання фізичного резервного копіювання
- 3) Зменшення кількості таблиць через об'єднання невеликих таблиць
- 4) Виключення індексів із копії

22. Що є обов'язковим для створення автоматизованого плану резервного копіювання?

- 1) Налаштування скриптів для планувальника завдань
- 2) Оновлення операційної системи сервера
- 3) Інтеграція інструментів моніторингу для контролю статусу копіювання
- 4) Попередня реалізація диференціального або інкрементного копіювання для зменшення навантаження

23. Що є головним недоліком інкрементального резервного копіювання?

- 1) Довший час відновлення даних
- 2) Збільшення розміру резервних копій
- 3) Відсутність підтримки автоматизації
- 4) Високе навантаження на систему під час копіювання

24. Які фактори можуть вплинути на успішність відновлення даних із резервної копії, особливо у критичних середовищах?

- 1) Відсутність регулярного тестування резервних копій та перевірки їхньої цілісності
- 2) Неповна відповідність структури бази даних, у якій виконувалося резервне копіювання, поточній версії системи під час відновлення
- 3) Наявність конфліктів між поточними даними системи та відновлюваною копією через некоректну конфігурацію резервного копіювання

4) Використання повного резервного копіювання без додаткових інкрементальних або диференційних копій у багаторівневих системах

25. Який із факторів найбільше впливає на вибір стратегії резервного копіювання?

- 1) Обсяг і критичність даних, які необхідно резервувати
- 2) Вибір операційної системи для серверів
- 3) Тип SQL-запитів, які використовуються у базі
- 4) Кількість користувачів бази даних

Компетенція № 9 – Звітувати про результати аналізу

1. Як слід обирати графічний формат для візуалізації даних у звіті?

- 1) Вибирати формат, який найточніше відображає числові дані та тренди
- 2) Вибирати найзрозуміліший для аудиторії формат, враховуючи контекст
- 3) Використовувати стандартні формати графіків, прийняті у галузі
- 4) Підлаштовувати візуалізацію під технічний рівень розуміння аудиторії

2. Чому важливо зазначати джерела даних у звіті?

- 1) Щоб аудиторія могла перевірити достовірність представлених висновків
- 2) Щоб уникнути маніпуляцій даними у звіті
- 3) Для підтвердження відповідності звіту корпоративним стандартам
- 4) Для забезпечення надійності та прозорості використаних джерел

3. Що є найбільш важливим у секції висновків у звіті?

- 1) Резюме основних результатів із рекомендаціями для прийняття рішень
- 2) Деталізація проведеного аналізу та застосованих інструментів
- 3) Перелік джерел, дані з яких використовувалися під час підготовки звіту
- 4) Обґрунтування методології аналізу для підтвердження коректності висновків

4. Що вказує на погану якість звіту про аналіз?

- 1) Наявність висновків, що не підкріплені реальними даними
- 2) Використання надто складних для сприйняття графіків
- 3) Відсутність пояснень щодо джерел і методів обробки даних
- 4) Включення непотрібних технічних деталей у висновки

5. Чому важливо тестувати звіт на представниках цільової аудиторії?

- 1) Щоб перевірити, чи є висновки зрозумілими та практичними
- 2) Щоб уникнути непотрібних деталей у тексті
- 3) Щоб зменшити обсяг візуалізації у звіті
- 4) Щоб підготувати звіт до публікації у відкритих джерелах

6. Що є основним викликом під час звітування для нетехнічної аудиторії?

- 1) Спрощення складних термінів і концепцій для зрозумілості

- 2) Збільшення обсягу тексту для кращого пояснення
- 3) Додавання всіх технічних деталей для зрозумілості
- 4) Використання лише графічних елементів замість тексту

7. Яка перевага використання інтерактивної візуалізації у звіті?

- 1) Можливість динамічного оновлення даних для різних сценаріїв
- 2) Зменшення текстового навантаження звіту
- 3) Спрощення обробки даних перед презентацією
- 4) Виключення необхідності в додаткових поясненнях

8. Який підхід найкращий для відображення кореляцій у звіті?

- 1) Графічне представлення із поясненням можливої причинності
- 2) Використання таблиць із деталізацією числових значень
- 3) Текстовий опис із включенням статистичних показників
- 4) Виділення кореляцій як ключових висновків звіту

9. Що означає «прозорість звіту» у контексті аналітики даних?

- 1) Надання чіткої інформації про джерела, методи та обмеження аналізу
- 2) Використання максимально простої структури звіту
- 3) Виключення складних даних, що можуть викликати запитання
- 4) Включення лише релевантних даних без додаткових пояснень

10. Що є основним ризиком використання даних із кількох джерел?

- 1) Виникнення розбіжностей між джерелами, які потребують пояснення
- 2) Ускладнення структури звіту через велику кількість інформації
- 3) Занадто просте сприйняття даних аудиторією
- 4) Необхідність перетворення всіх даних у єдиний формат

11. Що є основною перевагою звіту у реальному часі?

- 1) Можливість миттєвого реагування на зміну даних
- 2) Зменшення обсягу підготовчої роботи
- 3) Автоматизація всієї аналітики даних
- 4) Виключення похибок у даних

12. Як забезпечити узгодженість термінології у звіті?

- 1) Створити глосарій ключових термінів і використовувати його послідовно
- 2) Використовувати різні терміни залежно від аудиторії
- 3) Пояснювати терміни у вступі звіту
- 4) Замінювати складні терміни на більш зрозумілі синоніми

13. Як слід діяти, якщо у звіті виявлено конфлікт даних?

- 1) Вказати на конфлікт із поясненням його можливих причин
- 2) Виключити конфліктні дані із загального аналізу
- 3) Додати метрики, що уточнюють, для спрощення ситуації
- 4) Замінити конфліктні дані на середні значення

14. Як правильно зазначити рівень похибки в аналізі?

- 1) Окремо виділити похибки в секції методології або обмежень
- 2) Вказати похибки у висновках із деталізацією їхнього впливу на результати
- 3) Зазначити похибки в окремих поясненнях у додатках до звіту
- 4) Додати окрему секцію, присвячену лише обчисленням похибок

15. Як слід діяти, якщо висновки звіту суперечать очікуванням замовника?

- 1) Надати додаткові пояснення та рекомендації щодо ситуації
- 2) Узгодити суперечливі висновки із замовником і адаптувати їхню форму представлення у звіті
- 3) Переглянути параметри аналізу та уточнити методологію для корекції результатів
- 4) Замінити дані на інші, які краще відповідають очікуванням

16. Яка основна мета використання KPI у звіті?

- 1) Відстеження прогресу у досягненні бізнес-цілей
- 2) Визначення ключових точок для моніторингу вартості реалізації бізнес-процесів
- 3) Надання широкого огляду всіх метрик
- 4) Спрощення візуалізації для аудиторії

17. Як уникнути інформаційного перевантаження у звіті?

- 1) Фокусуватися лише на ключових даних та висновках
- 2) Включати всі дані, але додавати їх у додатки
- 3) Пояснювати складні метрики за допомогою детального тексту у висновках
- 4) Скорочувати візуалізації, залишаючи лише найважливіші графіки для акцентів

18. Чому важливо включати історичні дані у звіт?

- 1) Для демонстрації трендів та змін з часом
- 2) Для ілюстрації змін у довгостроковій перспективі і визначення ключових закономірностей
- 3) Для підвищення складності аналізу
- 4) Для підтвердження точності сучасних даних

19. Як інтегрувати різні джерела даних у звіт?

- 1) Узгоджувати їхній формат і пояснювати будь-які розбіжності
- 2) Використовувати лише ті джерела, які підтверджують очікування замовників
- 3) Включати всі джерела та надавати узагальнений аналіз їхньої відповідності один одному
- 4) Представляти кожне джерело із зазначенням його специфіки та ролі у загальному аналізі

20. Як звіт має відображати аномалії у даних?

- 1) Виділяти аномалії окремим розділом із поясненням їхніх причин
- 2) Аналізувати аномальні дані окремо та пояснювати їхній вплив на загальний результат
- 3) Замінювати аномалії середніми значеннями
- 4) Візуалізувати дані з урахуванням аномалій, щоб показати їхню можливу значущість

21. Як звіт має відображати обмеження аналізу?

- 1) Вказувати обмеження в окремій секції висновків, пояснюючи їхній вплив
- 2) Узагальнювати обмеження в окремому розділі звіту без детального пояснення їхнього впливу
- 3) Додавати всі обмеження у додатки до звіту
- 4) Формулювати коментарі щодо обмежень у загальних рисах для підвищення зручності читання

22. Як вибрати правильний рівень деталізації для звіту?

- 1) Враховувати потреби аудиторії та мету звіту
- 2) Включати всі релевантні дані та додатково пояснювати їхні значення для контексту
- 3) Орієнтуватися на внутрішні стандарти звітування
- 4) Зосередитися на ключових метриках із акцентом на їхньому практичному застосуванні

23. Як звіт має відображати взаємозв'язок між кількома змінними?

- 1) Показувати взаємозв'язок через графіки та підкріплювати текстовими висновками
- 2) Пояснювати взаємозв'язки текстово, додаючи обмежений аналіз графічних даних
- 3) Використовувати статистичні показники, не деталізуючи їх графічно
- 4) Уникати акценту на взаємозв'язках, якщо немає чіткої прямої залежності між змінними

24. Як діяти, якщо під час аналізу було виявлено невідповідності в даних?

- 1) Вказати на невідповідності та пояснити їхній можливий вплив на висновки
- 2) Видалити дані, що не відповідають очікуваним результатам, із зазначенням причин їхнього виключення
- 3) Замінити відсутні значення наближеними значеннями, із коротким поясненням у висновках
- 4) Використовувати лише дані, що найкраще відображають основну мету аналізу, пояснюючи свій вибір

25. Що є найбільш доречним способом інтеграції прогнозів у звіт?

- 1) Представлення результатів прогнозу разом із методологією його побудови
- 2) Включення прогнозів із загальним описом підходу до аналізу, без деталізації технічних аспектів
- 3) Представлення результатів статистичних моделей із короткими текстовими поясненнями замість візуалізації
- 4) Зосередження на ключових сценаріях прогнозу, наголошуючи увагу на практичних висновках

Компетенція № 10 – Тестувати ІКТ-запити

1. Що є першочерговим під час тестування SQL-запиту на продуктивність?

- 1) Визначення часу виконання запиту
- 2) Оцінка обсягу даних, що обробляються
- 3) Аналіз наявних індексів
- 4) Аналіз плану виконання запиту

2. Яка мета використання інструмента EXPLAIN у MySQL?

- 1) Визначення часу виконання запиту
- 2) Демонстрація послідовності операцій у запиті
- 3) Оптимізація індексів таблиці
- 4) Тестування параметрів з'єднання бази даних з сервером

3. Що може викликати уповільнення запиту при JOIN-операціях?

- 1) Невідповідність типів даних у стовпцях для JOIN
- 2) Наявність NULL-значень у ключових полях
- 3) Відсутність вторинних індексів у таблицях
- 4) Надлишкова нормалізація таблиць

4. Що найкраще підходить для перевірки коректності запитів?

- 1) Написання тестових кейсів з очікуваними результатами
- 2) Тестування на невеликій вибірці даних
- 3) Використання профілювання запитів для аналізу продуктивності
- 4) Автоматичне генерування SQL-запитів для тестів

5. Чому важливо тестувати запити в умовах реального навантаження?

- 1) Щоб перевірити продуктивність системи в пікові години
- 2) Щоб уникнути можливих помилок у тестовому середовищі
- 3) Щоб виявити конфлікти між різними типами запитів
- 4) Щоб підготувати документацію для користувачів

6. Як оцінити ефективність кешування результатів SQL-запитів?

- 1) Кешування ефективне, якщо зменшується кількість виконаних запитів до бази
- 2) Кешування ефективне, якщо скорочується час виконання запитів під час повторного виклику
- 3) Провести аналіз обсягу даних, збережених у кеші
- 4) Використати функції профілювання бази даних

7. Що важливо перевірити під час тестування транзакцій?

- 1) Час виконання кожної окремої операції в транзакції
- 2) Відкат транзакцій у разі помилок
- 3) Наявність усіх необхідних індексів для транзакцій
- 4) Сумісність транзакцій з іншими операціями в базі

8. Що може призвести до надмірного використання ресурсів під час виконання SQL-запиту?

- 1) Використання підзапитів замість JOIN
- 2) Наявність дублікатів у даних
- 3) Відсутність WHERE-умови для фільтрації результатів
- 4) Вибір стовпців із великим обсягом текстових даних

9. Який тест допоможе перевірити стійкість запитів до помилок даних?

- 1) Навантажувальне тестування
- 2) Тестування за допомогою некоректних вхідних даних
- 3) Оптимізаційне тестування
- 4) Тестування часу відгуку запиту

10. Чому важливо тестувати запити на реальних даних?

- 1) Щоб перевірити час виконання запиту у великій базі
- 2) Щоб виявити потенційні конфлікти в схемі бази
- 3) Щоб врахувати особливості реального розподілу даних
- 4) Щоб забезпечити сумісність запитів з різними версіями SQL

11. Що є найкращим способом виявлення помилок у запитах?

- 1) Використання автоматизованих тестів для SQL-запитів
- 2) Використання обмеженого обсягу тестових даних
- 3) Виконання написаних вручну запитів у тестовому середовищі
- 4) Аналіз результатів виконання запиту у плані виконання транзакції

12. Як тестувати складні запити з великою кількістю підзапитів?

- 1) Розділяти запит на менші частини та тестувати їх окремо
- 2) Зменшити обсяг даних для тестування
- 3) Використовувати інструменти для візуалізації структури запиту
- 4) Оптимізувати всі частини перед тестуванням

13. Що означає термін «план виконання запиту»?

- 1) Послідовність інструкцій, що визначають оптимізацію запиту
- 2) Детальний звіт про результати виконання запиту
- 3) Опис структури бази даних для виконання запиту
- 4) Графік навантаження, створений у результаті виконання запиту

14. Що є основною метою тестування SQL-запитів у багатокористувацькому середовищі?

- 1) Перевірка швидкості виконання при одночасному доступі до даних кількох користувачів
- 2) Забезпечення цілісності даних під час оновлення записів
- 3) Мінімізація часу виконання SELECT-запитів
- 4) Успішне виконання тестів із симульованим доступом користувачів

15. Що може свідчити про необхідність оптимізації SQL-запиту?

- 1) Високий час виконання запиту
- 2) Зменшення продуктивності системи під час виконання запиту
- 3) Наявність у результатах некоректних або зайвих даних
- 4) Використання великої кількості підзапитів замість JOIN-операцій

16. Який інструмент найбільш ефективний для тестування продуктивності запитів у базі даних PostgreSQL?

1. EXPLAIN
2. JMeter
3. pgBench
4. SQL Profiler

17. Що може призвести до блокування в багатокористувацькому середовищі?

- 1) Виконання SELECT-запитів без фільтрації
- 2) Використання UPDATE-запитів без умов WHERE
- 3) Виконання складних JOIN-операцій відразу багатьма користувачами
- 4) Виконання транзакцій без завершення (COMMIT або ROLLBACK)

18. Як визначити, що SQL-запит виконується неефективно?

- 1) План виконання запиту показує повне сканування таблиці
- 2) Результати містять помилки в даних
- 3) Час виконання перевищує 1 секунду
- 4) Індокси на ключових стовпцях не були створені

19. Що допомагає зменшити час виконання складних запитів?

- 1) Використання матеріалізованих уявлень
- 2) Створення індексів на стовпцях, які часто використовуються
- 3) Розділення таблиць на менші сегменти
- 4) Уникання використання функцій у WHERE-умовах

20. Як найкраще перевірити коректність роботи умов WHERE у запиті?

- 1) Виконати запит із включенням усіх можливих умов

- 2) Тестувати запит із малим набором тестових даних
- 3) Використовувати запити без фільтрації для порівняння результатів
- 4) Аналізувати час виконання запиту на великій вибірці даних

21. Що допомагає уникнути надмірного використання пам'яті під час виконання SQL-запитів?

- 1) Використання LIMIT у SELECT-запитах
- 2) Зменшення кількості стовпців у результатах
- 3) Уникання ORDER BY у великих запитах
- 4) Розбиття великих таблиць на партиції

22. Яка основна мета використання HINTS у SQL-запитах?

- 1) Примусити систему виконати запит певним чином
- 2) Визначити оптимальний порядок виконання операцій
- 3) Збільшити швидкість виконання запиту в обхід індексів
- 4) Забезпечити сумісність запиту з різними СУБД

23. Що допомагає перевірити стійкість SQL-запиту до зміни даних?

- 1) Запуск запиту на базі даних із великими обсягами записів
- 2) Виконання запиту в умовах високого навантаження
- 3) Тестування із симуляцією реальних сценаріїв змін у даних
- 4) Аналіз результатів запиту після оптимізації

24. Як протестувати ефективність паралельного виконання SQL-запитів?

- 1) Використовувати SQL HINTS для примусового паралелізму
- 2) Запускати запити в окремих потоках вручну
- 3) Аналізувати результати виконання з різною кількістю одночасних з'єднань
- 4) Оптимізувати індекси запитів перед тестуванням

25. Як перевірити коректність обробки дублікатів у запитах?

- 1) Використовувати DISTINCT у SELECT-запитах
- 2) Виконувати запити на тестовій базі без дублікатів
- 3) Застосовувати JOIN-операції для видалення дублікатів
- 4) Аналізувати результати із включенням усіх можливих комбінацій даних

Компетенція № 11 – Використовувати інтерфейс конкретного додатку

1. Який інструмент SQL Server Management Studio (SSMS) дозволяє візуалізувати залежності між об'єктами бази даних?

- 1) Object Explorer
- 2) Query Designer
- 3) Database Diagram
- 4) Table Designer

2. Що дозволяє виконати інструмент «Profiler» у SSMS?

- 1) Оптимізувати запит на основі рекомендацій
- 2) Відстежувати виконання SQL-запитів у реальному часі
- 3) Виконувати моніторинг фізичного стану сервера
- 4) Проаналізувати час виконання транзакцій

3. Яка функція в MySQL Workbench використовується для візуального проєктування бази даних?

- 1) Workbench Visual Designer
- 2) Database Modeler
- 3) EER Diagram
- 4) Schema Builder

4. Що потрібно для підключення до бази даних через PostgreSQL pgAdmin?

- 1) Ввести назву бази даних і тип автентифікації
- 2) Вказати URL сервера, порт, назву бази та облікові дані
- 3) Зазначити IP-адресу сервера, порт та назву бази
- 4) Вказати назву бази та SSL-сертифікати

5. Як у MySQL Workbench перевірити статус серверного підключення?

- 1) Виконати команду SHOW STATUS
- 2) Відкрити вкладку Server Status
- 3) Використати SQL-запит SELECT @@server_status
- 4) Запустити Performance Schema

6. Що дозволяє зробити інструмент Table Designer у SSMS?

- 1) Додавати, змінювати або видаляти індекси таблиці
- 2) Виконувати операції з транзакціями
- 3) Встановлювати зв'язки між таблицями
- 4) Змінювати значення в таблиці в режимі реального часу

7. Як можна перевірити журнал транзакцій в PostgreSQL pgAdmin?

- 1) На вкладці Query Tool

- 2) У розділі Log Files у меню Maintenance
- 3) У розділі Statistics бази даних
- 4) На вкладці SQL Profiler

8. Як можна створити нову схему в базі даних в PostgreSQL pgAdmin?

- 1) Виконати SQL-запит CREATE SCHEMA
- 2) Використати вкладку Schemas у головному вікні
- 3) Створити через контекстне меню бази даних > New Schema
- 4) Використати інструмент Schema Designer

9. Яка опція в SSMS дозволяє планувати автоматичні завдання, наприклад, резервне копіювання?

- 1) Task Scheduler
- 2) SQL Agent Jobs
- 3) Maintenance Plans
- 4) Automation Console

10. Як в інтерфейсі Oracle SQL Developer можна переглянути дані в таблиці?

- 1) Виконати SQL-запит SELECT * FROM table_name
- 2) Використати вкладку Data у вікні таблиці
- 3) Вибрати опцію View Data у контекстному меню таблиці
- 4) Перейти до вкладки Contents у вікні об'єкта

11. Яка функція в MySQL Workbench дозволяє відстежувати використання ресурсів сервера?

- 1) Resource Monitor
- 2) Server Performance Dashboard
- 3) Performance Schema Viewer
- 4) Server Resource Tracker

12. Як у PostgreSQL pgAdmin виконати масове завантаження даних у таблицю?

- 1) Використати утиліту pg_dump
- 2) Виконати команду COPY у Query Tool
- 3) Скористатися опцією Import Data у контекстному меню таблиці
- 4) Використати функцію Bulk Data Loader

13. Що дозволяє зробити вкладка «Triggers» у SSMS?

- 1) Створювати, редагувати та видаляти тригери
- 2) Переглядати залежності між тригерами та таблицями
- 3) Вмикати або вимикати тригери
- 4) Генерувати SQL-код для створення тригерів

14. Як у MySQL Workbench перевірити журнал подій сервера?

- 1) Використати Server Logs у меню Management
- 2) Виконати команду SHOW EVENTS у SQL Editor
- 3) Відкрити вкладку Logs у Server Administration
- 4) Використати утиліту Event Viewer

15. Як у PostgreSQL pgAdmin створити зв'язок між таблицями?

- 1) Використати опцію Create Foreign Key у контекстному меню таблиці
- 2) Виконати SQL-запит ALTER TABLE ADD FOREIGN KEY
- 3) Використати Relationship Tool у панелі інструментів
- 4) Створити новий зв'язок через вкладку Constraints у таблиці

16. Яка функція у MySQL Workbench дозволяє порівняти схеми баз даних?

- 1) Schema Compare
- 2) Database Diff
- 3) Workbench Schema Sync
- 4) Database Migration Wizard

17. Як у PostgreSQL pgAdmin виконати аналіз продуктивності SQL-запиту?

- 1) Використати функцію Query Plan Viewer
- 2) Запустити Performance Profiler
- 3) Перейти у вкладку Query Performance у головному вікні
- 4) Виконати запит через Explain

18. Як у SSMS створити індекс для таблиці?

- 1) Використати вкладку Indexes у властивостях таблиці
- 2) Виконати команду CREATE INDEX у Query Editor
- 3) Обрати New Index у контекстному меню таблиці
- 4) Використати опцію Index Builder у панелі інструментів

19. Як у MySQL Workbench створити резервну копію бази даних?

- 1) Використати функцію Data Export
- 2) Виконати SQL-запит BACKUP DATABASE
- 3) Використати Server Backup Manager
- 4) Обрати Backup Now у вкладці Server Status

20. Що потрібно зробити у PostgreSQL pgAdmin для відновлення бази даних із резервної копії?

- 1) Виконати команду RESTORE DATABASE у Query Tool
- 2) Використати опцію Restore у контекстному меню бази

- 3) Обрати вкладку Recovery у Server Dashboard
- 4) Використати Backup and Restore Manager

21. Яка функція у PostgreSQL pgAdmin дозволяє перевірити розмір таблиці?

- 1) Виконати SQL-запит SELECT pg_table_size()
- 2) Використати вкладку Statistics для таблиці
- 3) Відкрити Server Dashboard і знайти розмір таблиці
- 4) Обрати опцію Table Size Checker у контекстному меню таблиці

22. Як у MySQL Workbench створити обмеження FOREIGN KEY для таблиці?

- 1) Використати вкладку Foreign Keys у Table Editor
- 2) Виконати SQL-запит ADD CONSTRAINT FOREIGN KEY
- 3) Створити через опцію Relationship Builder у меню Tools
- 4) Обрати New Relationship у візуальному редакторі EER Diagram

23. Як у PostgreSQL pgAdmin переглянути історію виконаних запитів?

- 1) Використати вкладку Query History у Query Tool
- 2) Перейти до розділу Logs у головному меню
- 3) Використати Execution History у Server Dashboard
- 4) Обрати SQL Logs у контекстному меню бази даних

24. Яка функція у MySQL Workbench дозволяє переглянути всі поточні з'єднання до сервера?

- 1) Server Status
- 2) Connections Viewer
- 3) Performance Schema
- 4) Client Connections

25. Як у SQL Developer виконати відкат транзакції?

- 1) Виконати SQL-запит ROLLBACK
- 2) Обрати Undo Transaction у меню Tools
- 3) Використати опцію Rollback Transaction у вкладці Connection
- 4) Виконати SQL-запит TRANSACTION ROLLBACK

Компетенція № 12 – Використовувати бази даних

1. Що забезпечує ACID-принцип у базах даних?

- 1) Безпеку доступу до даних
- 2) Цілісність транзакцій
- 3) Зменшення часу виконання запитів
- 4) Створення резервних копій

2. Що є ключовою особливістю OLAP-систем?

- 1) Підтримка транзакцій
- 2) Аналіз великих обсягів даних
- 3) Підтримка складних JOIN-операцій
- 4) Використання NoSQL-баз

3. Яка команда SQL використовується для видалення всіх даних з таблиці без збереження її структури?

- 1) DELETE
- 2) DROP
- 3) TRUNCATE
- 4) REMOVE

4. Що означає ключове слово DISTINCT у SQL-запитах?

- 1) Виконує сортування результатів
- 2) Видаляє всі дублікати з результатів
- 3) Обмежує кількість рядків у результатах
- 4) Вибирає лише рядки з NULL-значеннями

5. Який запит використовується для обмеження кількості результатів у SQL?

- 1) LIMIT
- 2) RESTRICT
- 3) FETCH
- 4) REDUCE

6. Який із перелічених типів баз даних є NoSQL?

- 1) PostgreSQL
- 2) MongoDB
- 3) Oracle DB
- 4) MySQL

7. Що забезпечує індекс типу TTL (Time To Live) у MongoDB?

- 1) Сортування документів за часом створення
- 2) Автоматичне видалення документів після визначеного часу
- 3) Фільтрацію документів у реальному часі
- 4) Створення журналу транзакцій для кожного документа

8. Що виконує функція \$lookup в агрегації у MongoDB?

- 1) Фільтрує документи за критеріями
- 2) З'єднує колекції на основі вказаного ключа
- 3) Додає нове поле до документів
- 4) Підраховує кількість документів у колекції

9. Що виконує функція \$graphLookup у MongoDB?

- 1) З'єднує дані між колекціями у вигляді графа
- 2) Додає рекурсивні зв'язки між документами
- 3) Пошук документів на основі графової структури з рекурсією
- 4) Генерує дерево залежностей для колекції

10. Для чого використовується опція collation у MongoDB?

- 1) Задає порядок сортування для колекції
- 2) Налаштовує правила обробки тексту для пошуку та сортування
- 3) Оптимізує індекси для текстових запитів
- 4) Використовується для налаштування реплікації даних

11. Що робить функція shardCollection у MongoDB?

- 1) Збирає всі шардовані дані в один вузол
- 2) Розподіляє дані між шардованими колекціями
- 3) Ініціює процес реплікації між шардованими серверами
- 4) Встановлює конфігурацію шардованої колекції

12. Що забезпечує функція changeStream у MongoDB?

- 1) Запис змін у колекціях у спеціальний журнал
- 2) Створення резервних копій у режимі реального часу
- 3) Поточкову відстежуваність змін у документах колекції
- 4) Ініціалізацію реплікації між вузлами кластера

13. Що робить функція array_agg() у PostgreSQL?

- 1) Обчислює середнє значення числових стовпців
- 2) Об'єднує рядки в одне текстове поле
- 3) Перетворює значення стовпця в масив
- 4) Видаляє дублікати значень у результатах

14. Що робить функція tsvector у PostgreSQL?

- 1) Перетворює текст у структурований формат для пошуку
- 2) Стискає текстові поля для економії місця
- 3) Визначає унікальні рядки в таблиці
- 4) Аналізує продуктивність текстових запитів

15. Яка функція використовується для пошуку за патернами у PostgreSQL?

- 1) ILIKE
- 2) REGEXP_LIKE
- 3) SIMILAR TO
- 4) MATCH

16. Що робить функція jsonb_each_text() у PostgreSQL?

- 1) Розбирає JSON-документ на ключі та значення у вигляді тексту
- 2) Конвертує JSON у текстовий рядок
- 3) Фільтрує JSON-об'єкти за вказаними ключами
- 4) Створює новий JSON-документ із текстових полів

17. Що забезпечує функція hstore у PostgreSQL?

- 1) Зберігання даних у форматі ключ-значення в одному стовпці
- 2) Зберігання усіх первинних ключів в окремій таблиці
- 3) Зберігання активних зовнішніх ключів в окремій таблиці
- 4) Зберігання даних, які відповідають зовнішнім ключам, в окремому стовпці активної таблиці

18. Яка функція PostgreSQL використовується для пошуку схожих рядків?

- 1) levenshtein_distance()
- 2) similarity()
- 3) text_similarity_score()
- 4) match_ratio()

19. Яка команда дозволяє створити тригер у MySQL?

- 1) ADD TRIGGER
- 2) CREATE TRIGGER
- 3) INSERT TRIGGER
- 4) NEW TRIGGER

20. Яка команда в MySQL використовується для створення тимчасового індексу?

- 1) CREATE INDEX TEMPORARY
- 2) ALTER TABLE ADD INDEX TEMP
- 3) CREATE TEMPORARY INDEX
- 4) CREATE INDEX

21. Для чого використовується функція GROUP_CONCAT() у MySQL?

- 1) Конкатенує рядки, що мають однаковий групуючий ключ
- 2) Конкатенує всі рядки в таблиці в одну строку

- 3) Обчислює кількість рядків у кожній групі
- 4) Видаляє дублікати значень у групах

22. Що робить функція ST_Contains() у MySQL?

- 1) Визначає, чи один геометричний об'єкт повністю міститься в іншому
- 2) Обчислює площу багатокутника
- 3) Перетворює геометричний об'єкт у текстовий формат
- 4) Визначає перетин двох геометричних об'єктів

23. Яка функція використовується для перевірки цілісності паролів у MySQL?

- 1) VALIDATE_PASSWORD_STRENGTH()
- 2) PASSWORD_POLICY_CHECK()
- 3) VALIDATE_PASSWORD()
- 4) CHECK_PASSWORD_HASH()

24. Що виконує команда ANALYZE TABLE у MySQL?

- 1) Визначає найефективніші індекси для таблиці
- 2) Збирає статистику для оптимізації запитів
- 3) Аналізує структуру таблиці та виправляє помилки
- 4) Виконує тестове завантаження таблиці для аналізу продуктивності

25. Для чого використовується функція INET_ATON() у MySQL?

- 1) Перетворює IP-адресу в текстовий формат
- 2) Перетворює IP-адресу з текстового формату у числовий формат
- 3) Виконує пошук IP-адрес у діапазоні
- 4) Аналізує використання IP-адрес у мережі

Компетенція № 13 – Писати документацію до баз даних

1. Який розділ є обов'язковим у технічній документації до бази даних?

- 1) Глосарій термінів
- 2) Розділ із прикладами SQL-запитів
- 3) Опис даних і взаємозв'язків між таблицями
- 4) Інструкція з налаштування серверів

2. Як найкраще документувати структуру бази даних?

- 1) Використовувати текстові описи для кожної таблиці
- 2) Створити ER-діаграму для візуального представлення структури
- 3) Перерахувати всі таблиці в алфавітному порядку з короткими описами
- 4) Використовувати Use Cases для опису кожної таблиці

3. Що є ключовим під час документування реляцій у базі даних?

- 1) Визначення типів реляцій (один-до-одного, один-до-багатьох тощо)
- 2) Опис обмежень для кожного стовпця таблиці
- 3) Включення SQL-коду для створення зв'язків
- 4) Документування правил оновлення та видалення зв'язаних даних

4. Що є найбільш важливим під час документування змін у структурі бази даних?

- 1) Зазначення дати й автора змін
- 2) Створення нового розділу документації для кожної зміни
- 3) Додавання SQL-коду для відновлення попередньої версії структури
- 4) Документування лише змін у ключових таблицях

5. Що є обов'язковим у розділі документації, присвяченому безпеці бази даних?

- 1) Інструкції щодо шифрування даних у таблицях
- 2) Перелік користувачів із правами доступу
- 3) Інструкції для створення резервних копій
- 4) Опис механізмів автентифікації та авторизації

6. Які інструменти допомагають автоматизувати створення документації до баз даних?

- 1) dbForge Studio for SQL Server
- 2) Doxygen
- 3) Dataedo
- 4) Sphinx

7. Що має бути зазначено у документації до логів бази даних?

- 1) Місце зберігання логів і формат файлів
- 2) Типи запитів, які записуються в логи
- 3) Імена користувачів, які мають доступ до логів
- 4) Графік очищення логів

8. Як краще організувати розділ із прикладами SQL-запитів?

- 1) Сортувати приклади за алфавітом
- 2) Групувати за категоріями запитів (вибірка, оновлення, видалення)
- 3) Додавати приклади лише для складних запитів з підзапитами
- 4) Перераховувати SQL-запити у порядку, який відповідає списку користувачів та адміністраторів бази

9. Що важливо зазначити у розділі про обмеження бази даних?

- 1) Типи індексів, що використовуються
- 2) Максимальні обсяги збереження даних і кількість користувачів
- 3) Політики резервного копіювання
- 4) Методи обробки дублікатів у таблицях

10. Що є важливим при написанні документації до індексів?

- 1) Перелік індексів із зазначенням відповідних таблиць і стовпців
- 2) Назви таблиць та стовпців, які мають індекси
- 3) Алгоритми обчислення хеш-функцій
- 4) Опис сценаріїв, у яких індекси не застосовуються

11. Що є основною частиною документації до транзакцій?

- 1) Приклади складних транзакцій
- 2) Визначення рівнів ізоляції транзакцій
- 3) Алгоритми роботи з транзакціями в піковий час
- 4) Вибір типу бази даних для підтримки транзакцій

12. Що включає документація до процедур і тригерів у базі даних?

- 1) SQL-код процедур і тригерів
- 2) Опис сценаріїв, у яких використовуються процедури і тригери
- 3) Інструкцію зі створення та видалення процедур
- 4) Пояснення, як тригери взаємодіють із зовнішніми додатками

13. Які інструменти можуть бути використані для автоматичного створення ER-діаграм у документації?

- 1) DataGrip
- 2) dbForge Studio
- 3) Lucidchart
- 4) Усі перераховані вище

14. Що є ключовим під час документування резервного копіювання бази даних?

- 1) Зазначення розміру резервної копії
- 2) Перелік інструментів, які використовуються для резервування
- 3) Графік створення копій та інструкція з відновлення
- 4) SQL-запити, які виконуються під час резервного копіювання

15. Що є основною перевагою використання Markdown для документації?

- 1) Можливість інтеграції з системами контролю версій
- 2) Підтримка візуалізації SQL-запитів
- 3) Автоматичне створення ER-діаграм
- 4) Генерація SQL-запитів безпосередньо з тексту

16. Що має бути описано в розділі документації, присвяченому оптимізації запитів?

- 1) Найкращі практики використання індексів
- 2) Алгоритми для пошуку оптимальних маршрутів виконання запитів
- 3) Типи запитів, які слід уникати
- 4) ER-діаграми запитів

17. Що має бути зазначено у документації до текстового пошуку в базі даних?

- 1) Алгоритм роботи індексів для текстового пошуку
- 2) Приклади використання FULLTEXT-індексів
- 3) Визначення підтримуваних мов для пошуку
- 4) Усе перераховане вище

18. Що забезпечує опис журналів транзакцій у документації?

- 1) Аналіз часу виконання запитів
- 2) Зазначення розміру файлів журналу
- 3) Розуміння послідовності змін у базі
- 4) Інструкцію з видалення журналів після архівування

19. Як найкраще документувати обмеження FOREIGN KEY у базі даних?

- 1) Додати діаграму, що показує всі залежності між таблицями
- 2) Перерахувати всі ключі у розділі «Обмеження»
- 3) Надати приклади запитів, що демонструють використання ключів
- 4) Вказати таблиці, до яких застосовуються ключі

20. Що є критичним під час документування ролей і прав доступу?

- 1) Опис сценаріїв використання кожної ролі

- 2) Перелік дозволів, наданих кожній ролі
- 3) Таблиця, яка порівнює права доступу між ролями
- 4) Політика безпеки щодо видалення ролей

21. Що має бути зазначено в розділі про механізми реплікації?

- 1) Тип реплікації, що використовується (синхронна або асинхронна)
- 2) Частота реплікаційних операцій
- 3) Опис серверів-джерел і серверів-отримувачів
- 4) Усе перераховане вище

22. Що є ключовим аспектом у розділі документації про помилки та їхню обробку?

- 1) Рекомендації щодо усунення помилок
- 2) Опис впливу помилок на продуктивність бази даних
- 3) Перелік типових помилок і кодів помилок
- 4) Інструкції щодо моніторингу помилок у реальному часі

23. Що є ключовим аспектом розділу документації про конфігурацію бази даних?

- 1) Опис параметрів конфігурації, що впливають на продуктивність
- 2) Інструкції з налаштування параметрів безпеки
- 3) Приклади конфігураційних файлів
- 4) Рекомендації щодо оновлення конфігурації для нових версій

24. Що є основною метою розділу документації про резервування даних?

- 1) Інструкції з тестування резервних копій
- 2) Перелік інструментів, що використовуються для резервування
- 3) Опис частоти резервування і способів збереження копій
- 4) Рекомендації щодо автоматизації процесу резервування

25. Що є ключовим аспектом під час документування підтримки різних версій бази даних?

- 1) Опис проблем сумісності між версіями
- 2) Перелік змін у схемі та функціях між версіями
- 3) Інструкції щодо оновлення бази до останньої версії
- 4) Приклади можливих сценаріїв міграції між версіями

2.3. Відповіді на тести

Компетенція № 1 – Застосовувати політику інформаційної безпеки

1. 2	6. 1	11. 2	16. 2	21. 2
2. 2	7. 2	12. 2	17. 2	22. 1
3. 2	8. 1	13. 2	18. 2	23. 2
4. 1	9. 2	14. 2	19. 1	24. 1
5. 1	10. 2	15. 1	20. 1	25. 1

Компетенція № 2 – Збалансовувати ресурси бази даних

1. 2	6. 2	11. 2	16. 1	21. 2
2. 3	7. 2	12. 2	17. 1	22. 1
3. 2	8. 1	13. 1	18. 1	23. 1
4. 3	9. 1	14. 1	19. 1	24. 1
5. 2	10. 3	15. 3	20. 1	25. 1

Компетенція № 3 – Збирати відгуки клієнтів про додатки

1. 2	6. 1	11. 1	16. 4	21. 2
2. 2	7. 1	12. 2	17. 2	22. 2
3. 3	8. 2	13. 2	18. 2	23. 2
4. 1	9. 2	14. 2	19. 2	24. 4
5. 4	10. 2	15. 2	20. 2	25. 2

Компетенція № 4 – Створювати моделі даних

1. 2	6. 2	11. 2	16. 1	21. 2
2. 3	7. 2	12. 3	17. 1	22. 2
3. 2	8. 3	13. 1	18. 1	23. 2
4. 2	9. 2	14. 2	19. 2	24. 1
5. 1	10. 2	15. 2	20. 1	25. 1

Компетенція № 5 – Оцінювати тривалість робіт

1. 1	6. 2	11. 3	16. 1	21. 1
2. 2	7. 1	12. 1	17. 2	22. 1
3. 1	8. 1	13. 1	18. 1	23. 1
4. 2	9. 1	14. 1	19. 1	24. 4
5. 2	10. 1	15. 1	20. 1	25. 4

Компетенція № 6 – Визначати вимоги замовника

1. 2	6. 1	11. 3	16.1	21. 1
2. 1	7. 2	12. 3	17. 1	22. 3
3. 3	8. 1	13. 1	18. 3	23. 1
4. 3	9. 1	14. 1	19. 1	24. 1
5. 2	10. 3	15. 1	20. 3	25. 1

Компетенція № 7 – Перекладати технічні тексти

1. 1	6. 2	11. 1	16. 3	21. 1
2. 1	7. 1	12. 1	17. 1	22. 1
3. 1	8. 1	13. 2	18. 1	23. 4
4. 1	9. 3	14. 3	19. 1	24. 1
5. 2	10. 1	15. 2	20. 1	25. 1

Компетенція № 8 – Створювати резервні копії

1. 2	6. 1	11. 1	16. 1	21. 1
2. 2	7. 2	12. 1	17. 2	22. 1
3. 3	8. 1	13. 1	18. 1	23. 1
4. 2	9. 4	14. 2	19. 1	24. 1
5. 2	10. 1	15. 1	20. 1	25. 1

Компетенція № 9 – Звітувати про результати аналізу

1. 2	6. 1	11. 1	16. 1	21. 1
2. 4	7. 1	12. 1	17. 1	22. 1
3. 1	8. 1	13. 1	18. 1	23. 1
4. 1	9. 1	14. 1	19. 1	24. 1
5. 1	10. 1	15. 1	20. 1	25. 1

Компетенція № 10 – Тестувати ІКТ-запити

1. 4	6. 2	11. 1	16. 3	21. 1
2. 2	7. 2	12. 1	17. 4	22. 1
3. 1	8. 3	13. 1	18. 1	23. 3
4. 1	9. 2	14. 1	19. 1	24. 3
5. 1	10. 3	15. 2	20. 2	25. 1

Компетенція № 11 – Використовувати інтерфейс конкретного додатку

1. 3	6. 1	11. 2	16. 2	21. 2
2. 2	7. 2	12. 2	17. 4	22. 1
3. 3	8. 3	13. 1	18. 3	23. 1
4. 2	9. 3	14. 1	19. 1	24. 1
5. 2	10. 2	15. 1	20. 2	25. 1

Компетенція № 12 – Використовувати бази даних

1. 2	6. 2	11. 4	16. 1	21. 1
2. 2	7. 2	12. 3	17. 1	22. 1
3. 2	8. 2	13. 3	18. 2	23. 3
4. 2	9. 3	14. 1	19. 2	24. 2
5. 1	10. 2	15. 3	20. 4	25. 2

Компетенція № 13 – Писати документацію до баз даних

1. 3	6. 3	11. 2	16. 1	21. 4
2. 2	7. 1	12. 2	17. 4	22. 3
3. 1	8. 2	13. 4	18. 3	23. 1
4. 1	9. 2	14. 3	19. 1	24. 3
5. 4	10. 1	15. 1	20. 3	25. 2

ВИСНОВКИ ЗА РОЗДІЛОМ II

Розділ II навчального посібника присвячений аналізу професії «Розробник баз даних», її вимогам та ключовим компетенціям, заснованим на класифікації ESCO.

Професія «Розробник баз даних» розглядається як невіддільна частина цифрової інфраструктури. Вона забезпечує основи для зберігання, обробки та аналізу даних, необхідних для ухвалення обґрунтованих рішень у бізнесі, державному управлінні та науці. Вимоги до представників цієї професії поєднують високий рівень технічної компетенції, стратегічного мислення та аналітичних здібностей.

Основними аспектами, розкритими в розділі, є завдання, що постають перед розробниками баз даних, та деталі компетенцій, необхідних для ефективного виконання завдань у цій професії. Зокрема, розглядається інтеграція баз даних у складні інформаційні середовища, підтримка їхньої функціональності, а також адаптація до змін у технологічному ландшафті. Значна увага приділяється здатності фахівців впроваджувати сучасні політики інформаційної безпеки, розробляти рекомендації щодо вибору та

використання відповідних інструментів управління даними, а також забезпечувати відповідність законодавчим нормам і регламентам.

Окрему частину розділу займає система тестування компетенцій, яка відіграє важливу роль у визначенні рівня підготовки фахівців. У розділі запропоновано 325 тестових завдань, кожне з яких має лише одну правильну відповідь. Завдання розділено на 13 тем, що відповідають основним компетенціям розробника баз даних. Ці теми охоплюють різні аспекти професійної діяльності, включно з інформаційною безпекою, моделюванням даних, оптимізацією запитів, створенням резервних копій, моніторингом продуктивності та багато інших. Такий формат тестування дозволяє перевірити як теоретичні знання, так і практичні навички, необхідні для ефективної роботи з базами даних.

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ДЛЯ ПІДГОТОВКИ

Друковані видання

1. Hansen K.B. Practical Oracle SQL: Mastering the Full Power of Oracle Database. New York : Apress Media, 2024. 460 p.
2. Powell G. Database Modeling Step by Step. London : Taylor & Francis. 2023. 248 p.
3. Nuijten A., Barel P. Modern Oracle Database Programming: Level Up Your Skill Set to Oracle's Latest and Most Powerful Features in SQL, PL/SQL, and JSON. New York : Apress Media, 2023. 576 p.
4. Abramson I., Malcher M. Oracle Database 12c Install, Configure & Maintain Like a Professional. New York : McGraw-Hill, 2023. 472 p.
5. Meier A., Kaufmann M. SQL & NoSQL Databases: Models, Languages, Consistency Options and Architectures for Big Data Management. New York : Springer, 2022. 229 p.
6. Foster E.C. Database Systems: A Pragmatic Approach, 3rd edition. London : Taylor & Francis, 2022. 596 p.
7. Zygiaris S. Database Management Systems: A Business-Oriented Approach Using ORACLE, MySQL and MS Access. Leeds : Emerald Publishing, 2023. 328 p.
8. Oppel A.J. SQL: A Beginner's Guide, Fourth Edition. New York : McGraw-Hill, 2022. 560 p.
9. Patni J.C. A Comprehensive Study of SQL: Practice and Implementation. London : Taylor & Francis, 2022. 218 p.
10. Garmany J., Walker J. Logical Database Design Principles. London : Taylor & Francis, 2021. 200 p.
11. Eckstein J., Schultz B.R. Introductory Relational Database Design for Business, with Microsoft Access. New York : Wiley. John Wiley & Sons, 2023. 328 p.
12. Taylor A.G. SQL All-in-One For Dummies, 7s book. New York : Wiley. John Wiley & Sons, 2022. 816 p.
13. Weiss K. A. Job Ready SQL. New York : Wiley. John Wiley & Sons, 2023. 416 p.
14. Gillenson M.L. Fundamentals of Database Management Systems. New York : Wiley. John Wiley & Sons, 2023. 416 p.
15. Date C.J. Database Design and Relational Theory: Normals Forms and All That Jazz. New York : Springer, 2022. 318 p.
16. Mannino M. Database Design, Query, Formulation, and Administration, 8th Edition. New York : McGraw-Hill, 2022. 427 p.

Інформаційні ресурси

Ці ресурси надають офіційну документацію, навчальні матеріали та інструменти, які можуть бути корисними для розробників баз даних та тих, хто прагне поглибити свої знання у цій сфері.

1. Офіційний сайт стандарту ESCO: <https://esco.ec.europa.eu/en>

Офіційний сайт стандарту ISO для мови структурованих запитів SQL: <https://www.iso.org/standard/76583.html>

2. Офіційний сайт регламенту захисту даних General Data Protection Regulation: <https://gdpr-info.eu/>

3. Офіційний сайт компанії Oracle: <https://www.oracle.com/>

4. Офіційний сайт платформи для управління базами даних Microsoft SQL Server: <https://www.microsoft.com/en-us/sql-server>

5. Офіційний сайт системи управління реляційними базами даних MySQL: <https://www.mysql.com/>

6. Офіційний сайт об'єктно-реляційної системи управління реляційними базами даних з відкритим кодом PostgreSQL: <https://www.postgresql.org/>

7. Офіційний сайт документо-орієнтована NoSQL база даних MongoDB: <https://www.mongodb.com/>

8. Офіційний сайт системи управління базами даних IBM Db2 від компанії IBM: <https://www.ibm.com/products/db2>

9. Офіційний сайт вбудованої реляційної бази даних з відкритим кодом SQLite: <https://www.sqlite.org/>

10. Офіційний сайт реляційної системи управління базами даних MariaDB: <https://mariadb.org/>

11. Офіційний сайт платформи для управління базами даних та аналітики HANA від компанії SAP: <https://www.sap.com/products/hana.html>

12. Цикл курсів з проектування, створення та адміністрування баз даних на платформі Coursera від University of Colorado: <https://www.coursera.org/instructor/~5265596>

13. Цикл курсів з проектування, створення та адміністрування баз даних на платформі LinkedIn: https://www.linkedin.com/learning/topics/database-management?trk=learning-browse_library-tabs-category-url&upsellOrderOrigin=default_guest_learning

14. Цикл курсів з проектування, створення та адміністрування баз даних на платформі Alison: Empower Yourself: <https://alison.com/tag/databases>

15. Цикл курсів з проектування, створення та адміністрування баз даних від Stanford University: <https://online.stanford.edu/courses/soe-ydatabases0005-databases-relational-databases-and-sql>

16. Цикл курсів з проектування, створення та адміністрування баз даних на платформі edX: <https://www.edx.org/learn/databases>

17. Безплатні курси з проектування, створення та адміністрування баз даних від компанії EPAM: <https://campus.epam.com/en/training>

18. Безплатні курси з проектування, створення та адміністрування баз даних від компанії SoftServe: <https://career.softserveinc.com/en-us/learning-and-certification/formats-fundamental>

Навчальне видання

Чуб Ольга Ігорівна
Хруслов Максим Михайлович

РОЗРОБНИК БАЗ ДАНИХ

Теоретично-тестовий практикум

Коректор *О. В. Анцибора*
Комп'ютерне верстання *В. В. Савінкова*
Макет обкладинки

Формат 60х84/16. Ум. друк. арк. 5,63. Наклад 100 пр. Зам. № 332/25.

Видавець і виготовлювач
Харківський національний університет імені В. Н. Каразіна,
61022, м. Харків, майдан Свободи, 4.
Свідоцтво суб'єкта видавничої справи ДК № 3367 від 13.01.2009
Видавництво ХНУ імені В. Н. Каразіна