

ХЕРСОНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
КАФЕДРА ІНФОРМАТИКИ І КОМП'ЮТЕРНИХ НАУК
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ДИЗАЙНУ

методичні рекомендації
та індивідуальні завдання для самостійної роботи студентів

Розгортання інформаційних систем
*(Архітектурні методики та архітектура програмних додатків у
інформаційних системах)*

для здобувачів	Третього курсу
підготовки	першого (бакалаврського) рівня
галузі знань	F (12) Інформаційні технології
спеціальності	F3 (122) Комп'ютерні науки
освітньо-професійної програми	Комп'ютерні науки
факультету	Інформаційних технологій та дизайну

Методичні рекомендації та індивідуальні завдання для самостійної роботи студентів з дисципліни « **Розгортання інформаційних систем**» (Архітектурні методики та архітектура програмних додатків у інформаційних системах) для здобувачів вищої освіти першого (бакалаврського) рівня спеціальності F3 (122) Комп'ютерні науки

РОЗРОБНИКИ: Наталя Корніловська к.т.н., доц.

Рецензент: Світлана Вишемирська к.т.н., доц.

Затверджено на засіданні кафедри І та КН

Протокол No 6 від 10.03.2026

В.о. зав. каф. І та КН к.м.н., доц. _____ Світлана Моїсеєнко

Відповідальний за випуск Наталя Корніловська к.т.н., доц.

Узгоджено з навчально-методичним відділом

Реєстраційний номер No _____

ВСТУП

Навчальна дисципліна «Розгортання інформаційних систем» спрямована на формування у студентів практичних компетентностей у сфері розгортання, налаштування та підтримки сучасних веб-застосунків і хмарних сервісів. Сучасний ринок праці у галузі інформаційних технологій висуває підвищені вимоги до фахівців, здатних не лише розробляти програмне забезпечення, але й забезпечувати його стабільну роботу у виробничому середовищі.

Процес розгортання (deployment) являє собою комплекс дій із перенесення застосунку з середовища розробника на сервери, де він буде доступний кінцевим користувачам. Опанування цього процесу є обов'язковою умовою для підготовки повноцінного IT-фахівця.

Дисципліна охоплює широке коло тем: від базових концепцій хмарних обчислень і моделей надання послуг (SaaS, PaaS, IaaS) до практичного використання сучасних платформ розгортання - Google Sites, GitHub Pages, Netlify та Amazon Web Services. Студенти опановують технології створення статичних веб-сайтів (HTML5, CSS3, JavaScript), системи контролю версій Git, основи кібербезпеки, роботу з REST API та хмарними сховищами.

Ця методичка є логічним доповненням до комплексу лабораторних робіт з дисципліни «Розгортання інформаційних систем». Вона не замінює лабораторні роботи, а розширює їх - пропонує додаткові кейси, теоретичні питання та тести, щоб студент міг самостійно перевірити та закріпити рівень засвоєння матеріалу.

Структура методички відповідає структурі курсу:

Розділ	Тема	Лабораторна робота
1	Хмарні технології та перший сайт у хмарі	Лаб. No 1-2
2	Публікація на GitHub Pages та Netlify	Лаб. No 3-4
3	Amazon Web Services: базова конфігурація та S3	Лаб. No 5-7, 10
4	REST API та кібербезпека	Лаб. No 8-9
5	Безсерверні обчислення та AWS CLI	Лаб. No 11-12

Кожен розділ містить:

- теоретичні питання для самоперевірки (перевірте себе перед здачею лабораторної);
- додатковий практичний кейс - аналогічний до лабораторного, але на новому матеріалі;
- індивідуальні варіанти завдань для самостійного виконання;
- тест з вибором відповіді.

РОЗДІЛ 1. ХМАРНІ ТЕХНОЛОГІЇ ТА ПЕРШИЙ САЙТ У ХМАРІ

Відповідає лабораторним роботам No 1-2

1.1. Теоретичні питання для самоперевірки

Дайте розгорнуту відповідь (3-5 речень) на кожне питання самостійно, без заглядання у конспект. Потім звірте з лекційним матеріалом.

- Що таке хмарні обчислення? Наведіть аналогію з реального життя, що пояснює цю концепцію.
- Поясніть відмінності між моделями IaaS, PaaS та SaaS. Наведіть конкретний приклад кожної моделі.
- Що таке SaaS-платформа? Чому Google Sites є прикладом SaaS?
- Поясніть «піраміду відповідальності» у SaaS: що залишається за клієнтом, а що бере на себе провайдер?
- Що таке HTML? Чим він відрізняється від мови програмування (наприклад, Python)?
- Поясніть роль CSS у веб-розробці. Що означає слово «каскадні» у назві CSS?
- Що таке адаптивний (responsive) дизайн і чому він є обов'язковою вимогою у 2026 році?
- Поясніть, що відбувається у браузері покроково: від відкриття сайту до відображення сторінки на екрані.

1.2. Додатковий практичний кейс: Онлайн-школа «Цифровий клас»

Контекст кейсу

Травень 2025 р. Невелика онлайн-школа «Цифровий клас» вирішила перенести свій навчальний портал з власного ноутбука викладача до хмарного середовища. До цього сайт існував лише локально і був недоступний для учнів поза домашньою мережею. Команда розглядає три варіанти розгортання: власний фізичний сервер, хостинг-провайдер (On-

Premise-like підхід) та Google Sites (SaaS). Бюджет обмежений, терміни стислі - необхідно забезпечити доступність сайту для 200 учнів вже за 48 годин.

Завдання до кейсу (обов'язкові)

Завдання 1.1. Порівняльний аналіз (30 балів)

Заповніть таблицю порівняння трьох варіантів розгортання для «Цифрового класу»:

Критерій	Власний сервер	Хостинг-провайдер	Google Sites (SaaS)
Вартість початкового розгортання			
Час до результату			
Технічні навички, необхідні команді			
Доступність 24/7 (відповідальність)			
Масштабування при зростанні кількості учнів			

Зробіть обґрунтований висновок: який варіант оптимальний для «Цифрового класу» за умовами кейсу?

Завдання 1.2. Проектування структури сайту (20 балів)

Спроектуйте структуру сайту «Цифрового класу» для Google Sites:

- Визначте назви 4-6 розділів головної сторінки (наприклад: Про школу, Розклад, Викладачі тощо).
- Запропонуйте 3 додаткові сторінки (окрім головної) та опишіть їх зміст (2-3 речення кожна).
- Обґрунтуйте вибір теми оформлення: яке враження повинен справляти навчальний сайт?

Завдання 1.3. HTML/CSS аналіз (20 балів)

Студент отримав від замовника такий фрагмент коду сторінки:

```
<nav class="navbar">    <div class="logo">Цифровий клас</div>    <ul class="nav-links">        <li><a href="#courses">Курси</a></li>    </ul></nav>
```

- Визначте, який семантичний тег використаний та яке його призначення.
- Поясніть, що таке клас (class) у HTML та як він пов'язаний з CSS.
- Запропонуйте, який CSS Flexbox-код додати до .navbar, щоб логотип був ліворуч, а меню - праворуч.

Індивідуальні варіанти (самостійна робота)

Оберіть варіант за списком групи. Виконайте Завдання 1.1-1.3 для свого варіанту організації.

Вар.	Організація / проект	Кількість потенційних відвідувачів	Бюджет та термін
1	Студентський гурток робототехніки	150 осіб	Безкоштовно, 24 год
2	Домашня пекарня (мала підприємниця)	500 осіб	до 300 грн/міс, тиждень
3	Благодійний фонд збору коштів	2 000 осіб	Безкоштовно, 48 год
4	Місцева бібліотека	800 осіб	до 500 грн/міс, 2 тижні
5	ІТ-стартап (команда 3 особи)	1 000 осіб	до 1 000 грн/міс, 3 дні
6	Школа танців	400 осіб	Безкоштовно, 72 год
7	Особистий сайт-портфоліо студента	Рекрутери та HR	Безкоштовно, 2 дні
8	Районна газета (онлайн-версія)	5 000 осіб	до 2 000 грн/міс, місяць

1.3. Тест для самоперевірки (Розділ 1)

Оберіть одну правильну відповідь. Ключі наведено у виділеному рядку під кожним питанням.

1. Яка модель хмарних послуг передбачає, що клієнт відповідає ЛИШЕ за контент, а провайдер - за все інше?

А. IaaS Б. PaaS В. SaaS Г. On-Premise

Правильна відповідь: В - у моделі SaaS (Google Sites, Gmail) клієнт відповідає лише за вміст, провайдер керує усім технічним стеком.

2. Що означає тег <meta charset="UTF-8"> у HTML-документі?

А. Задає розмір шрифту для всієї сторінки Б. Встановлює кодування символів, що дозволяє коректно відображати кирилицю В. Підключає зовнішній CSS-файл Г. Вказує мову сторінки для пошукових систем

Правильна відповідь: Б - без UTF-8 українські літери відображатимуться як «кракозябри».

3. Яка CSS-властивість дозволяє розташувати елементи навігаційного меню у горизонтальний рядок?

А. position: fixed Б. display: flex В. margin: auto Г. list-style: none

Правильна відповідь: Б - display: flex перетворює контейнер на flex-контейнер, елементи якого за замовчуванням розташовуються у рядок.

4. Чому файл головної сторінки сайту обов'язково повинен називатися index.html?

А. Це вимога браузера Chrome Б. Сервер шукає саме цей файл як головну сторінку - це міжнародний стандарт В. Інша назва призведе до помилок у CSS Г. Це обмеження Google Sites

Правильна відповідь: Б - усі веб-сервери за замовчуванням шукають index.html при зверненні до кореневого URL.

5. Студент написав `@media (max-width: 768px) { .burger { display: flex; } }`. Що це означає?

А. Кнопка-"гамбургер" відображається на всіх екранах Б. Кнопка-"гамбургер" з'являється лише на екранах шириною 768px і менше В. Кнопка-"гамбургер" ніколи не відображається Г. Медіа-запит вимикає Flexbox для мобільних пристроїв

Правильна відповідь: Б - медіа-запит `@media (max-width: 768px)` застосовується лише тоді, коли ширина екрана не перевищує 768 пікселів.

РОЗДІЛ 2. ПУБЛІКАЦІЯ НА GITHUB PAGES ТА NETLIFY

Відповідає лабораторним роботам No 3-4

2.1. Теоретичні питання для самоперевірки

- Що таке Git? Поясніть різницю між Git та GitHub простими словами.
- Що таке commit у Git? Чому важливо писати осмислені описи до кожного коміту?
- Яка послідовність трьох команд Git використовується для оновлення сайту після змін?
- Чому репозиторій для GitHub Pages обов'язково повинен називатися `username.github.io`?
- Що таке Netlify? Чим він відрізняється від GitHub Pages?
- Що означає принцип «git push і забув» у контексті Netlify?
- Що таке Netlify Forms? Яку проблему вони вирішують для статичного сайту?
- Що таке Deploy Preview у Netlify і навіщо він потрібен?

2.2. Додатковий практичний кейс: Репозиторій команди Robotech

Контекст кейсу

Червень 2025 р. Команда студентів «Robotech» (4 особи) розробляє сайт-документацію для свого курсового проекту. Вони вирішили використати GitHub Pages для

безкоштовного хостингу та Git для командної роботи над кодом. Перший тиждень роботи показав такі дані про git-активність команди:

- Разом зроблено 28 комітів за 5 днів роботи.
- Один учасник допустив помилку і завантажив файл style.css як Style.CSS (великими літерами) - сайт відображається без стилів на сервері.
- Після підключення до Netlify автоматичне оновлення не спрацює після git push.
- Команда хоче додати форму «Зворотний зв'язок» без використання серверного коду.

Завдання до кейсу

Завдання 2.1. Аналіз git-активності (25 балів)

- Розрахуйте середню кількість комітів на день та на одного учасника команди за перший тиждень.
- Які 5 команд Git мають знати всі члени команди як обов'язковий мінімум?

Поясніть кожному.

- Запропонуйте 3 правила написання описів комітів для командної роботи з прикладами (хороший та поганий опис).

Завдання 2.2. Діагностика та виправлення помилок (35 балів)

Для кожної проблеми кейсу:

- Поясніть причину виникнення (чому так трапилось).
- Запропонуйте конкретне рішення з покроковими інструкціями.
- Сформулюйте профілактичне правило для запобігання повторенню.

Завдання 2.3. Налаштування Netlify Forms (40 балів)

Напишіть HTML-код форми зворотного зв'язку для сайту «Robotech», що коректно працює з Netlify Forms. Форма повинна містити: поля «Ім'я», «Email», «Тема повідомлення» (select з 3 варіантами), «Текст повідомлення» та кнопку відправки. Поясніть роль атрибутів data-netlify та name у тегу form.

Індивідуальні варіанти (самостійна робота)

Оберіть варіант. Проаналізуйте ситуацію та дайте відповіді на запитання Завдань 2.1-2.2.

Вар.	Сценарій / проект	Платформа	Проблема для діагностики
1	Портфоліо студента-дизайнера	GitHub Pages	Зображення не відображаються (шлях images/ vs Images/)

2	Документація open-source проекту	GitHub Pages	Сайт відкривається, але без CSS після публікації
3	Лендінг для студентської конференції	Netlify + GitHub	Автооновлення не спрацьовує після push
4	Сайт туристичного клубу	Netlify	Форма не з'являється у Netlify Dashboard
5	Блог викладача університету	GitHub Pages	Помилка 404 після активації Pages
6	Онлайн-меню кафе	Netlify	Після відновлення через Rollback сайт не оновлюється
7	Сайт студентської ради	GitHub Pages	git push не відправляє зміни (помилка 403)
8	Портфоліо команди веб-розробників	Netlify + GitHub	Deploy Preview не генерується для Pull Request

2.3. Тест для самоперевірки (Розділ 2)

Оберіть одну правильну відповідь.

1. Що означає команда `git add .` (з крапкою)?

А. Додати лише файл `index.html` Б. Підключитись до GitHub-репозиторію В. Додати всі змінені файли поточної папки та підпапок до наступного коміту Г. Створити новий репозиторій

Правильна відповідь: В - крапка означає «поточна папка і все всередині». Відповідає `git add index.html + git add style.css + усі інші файли`.

2. Netlify автоматично оновив сайт після `git push`. Скільки часу зазвичай займає оновлення?

А. 30-90 секунд Б. 10-15 хвилин В. 1-2 години Г. Сайт оновлюється лише вручну

Правильна відповідь: А - Netlify CI/CD pipeline займає 30-90 секунд від push до публікації нової версії.

3. Що необхідно обов'язково додати до тегу `<form>`, щоб Netlify Forms обробляв дані?

А. Атрибут `action` з URL Netlify Б. Атрибути `data-netlify="true"` та `name` у тегу `form` і приховане поле `form-name` В. Тег `<script>` з Netlify SDK Г. Атрибут `method="netlify"`

Правильна відповідь: Б - без `data-netlify` та `hidden input form-name` Netlify не розпізнає форму.

4. Студент зробив помилку і опублікував сайт з критичним багом. Яка функція Netlify дозволяє найшвидше відновити попередню версію?

A. git revert Б. Deploy Preview В. Rollback (Publish deploy) у розділі Deploys Г. netlify reset

Правильна відповідь: В - Rollback у Netlify Dashboard займає 30 секунд і не потребує роботи з Git.

5. Яка перевага Netlify над GitHub Pages за ліміту трафіку?

A. Netlify надає 5 ГБ/місяць, GitHub Pages - 10 ГБ/місяць Б. Обидва надають однаково - по 10 ГБ/місяць В. Netlify надає 100 ГБ/місяць, GitHub Pages - 10 ГБ/місяць Г. Netlify не має обмежень трафіку

Правильна відповідь: В - Netlify безкоштовний план включає 100 ГБ/місяць трафіку, тоді як GitHub Pages обмежує 10 ГБ.

РОЗДІЛ 3. AMAZON WEB SERVICES: БАЗОВА КОНФІГУРАЦІЯ ТА S3

Відповідає лабораторним роботам No 5-7, 10

3.1. Теоретичні питання для самоперевірки

- Що таке AWS Free Tier? Які ресурси надаються безкоштовно протягом 12 місяців?
- Поясніть, навіщо при реєстрації в AWS потрібна кредитна картка, якщо послуга безкоштовна.
- Що таке IAM? Поясніть принцип мінімальних привілеїв на конкретному прикладі.
- Що таке Amazon S3-бакет? Чим він відрізняється від звичайної папки на комп'ютері?
- Що таке Bucket Policy і для чого вона використовується при статичному хостингу?
- Поясніть різницю між класами зберігання S3: Standard, Standard-IA та Glacier.
- Що таке версіонування (Versioning) в S3? Наведіть приклад, коли воно може «врятувати» дані.
- Що таке CDN? Поясніть, як географічно розподілена мережа серверів прискорює завантаження сайту.

3.2. Додатковий практичний кейс: Архівація медіаконтенту стартапу PhotoVault

Контекст кейсу

Серпень 2025 р. Стартап PhotoVault зберігає фотографії користувачів у Amazon S3. Поточна ситуація: загальний обсяг зберігання - 2 ТБ. Розподіл: 200 ГБ - нові завантаження (останні 30 днів), 600 ГБ - фото від 1 до 12 місяців, 1 200 ГБ - архівні фото (старіші за 1 рік). Тарифи (орієнтовні): S3 Standard - 0,023 USD/ГБ/міс, S3 Standard-IA - 0,0125 USD/ГБ/міс, S3 Glacier Instant - 0,004 USD/ГБ/міс.

Завдання до кейсу

Завдання 3.1. Розрахунок вартості зберігання (30 балів)

Розрахуйте поточну місячну вартість зберігання, якщо PhotoVault зберігає ВСІ дані у S3 Standard. Порівняйте з оптимальним розподілом: новий контент - у Standard, контент від 1 до 12 місяців - у Standard-IA, архів - у Glacier Instant. Визначте щомісячну економію у доларах США.

Завдання 3.2. Проектування Bucket Policy (30 балів)

PhotoVault має два типи контенту: публічні (доступні всім) та приватні (лише авторизованим). Опишіть структуру бакетів та зміст Bucket Policy для кожного типу. Чому не можна зберігати публічні та приватні фото в одному бакеті з однією політикою?

Завдання 3.3. Версіонування та відновлення (40 балів)

Один із користувачів PhotoVault випадково перезаписав цінне фото через завантаження нового файлу з такою самою назвою. Поясніть: як версіонування S3 захищає від такої ситуації; покрокову послідовність дій для відновлення попередньої версії файлу через AWS Console; яку Lifecycle Rule варто налаштувати, щоб зберігати не більше 5 версій кожного об'єкта.

Індивідуальні варіанти (самостійна робота)

Оберіть варіант. Виконайте Завдання 3.1-3.3, підставляючи власні дані.

Вар.	Тип сервісу	Standard (ГБ)	Standard-IA (ГБ)	Glacier (ГБ)
1	Відеоплатформа (навчальні курси)	100	500	2 000
2	Медична інформаційна система	50	200	800
3	Е-commerce (фото товарів)	300	700	1 500
4	Новинний портал (архів статей)	80	400	3 000

5	Сервіс резервного копіювання	20	150	5 000
6	Стрімінговий музичний сервіс	400	1 000	2 500
7	Платформа для гемінгу (скріншоти)	600	1 200	800
8	Університетська бібліотека (скани)	30	300	4 000

3.3. Тест для самоперевірки (Розділ 3)

1. Яка дія є ПЕРШОЮ, яку рекомендується виконати після реєстрації AWS-акаунту для забезпечення безпеки?

А. Запустити EC2-інстанс Б. Налаштувати Multi-Factor Authentication (MFA) для кореневого акаунту В. Створити S3-бакет Г. Завантажити AWS CLI

Правильна відповідь: Б - налаштування MFA для root user є базовим заходом безпеки, оскільки кореневий акаунт має необмежений доступ до всіх ресурсів.

2. Студент завантажив файл my_photo.jpg у S3-бакет і отримав публічний URL. Але при відкритті URL з'явилася помилка Access Denied. Найімовірніша причина:

А. Файл завантажено у неправильний регіон Б. Не налаштовано Bucket Policy для публічного доступу або увімкнено Block Public Access В. Розмір файлу перевищує ліміт S3 Г. URL неправильно сформовано браузером

Правильна відповідь: Б - без Bucket Policy з Action: s3:GetObject та Principal: * або при увімкненому Block Public Access файли S3 не доступні публічно.

3. Для яких даних найдоцільніше використовувати клас S3 Glacier Instant?

А. Файли сайту, до яких звертаються тисячі користувачів щодня Б. Бекап-копії баз даних, які можуть знадобитись у надзвичайних ситуаціях В. Зображення, що завантажуються на сторінки інтернет-магазину Г. Логи сервера за поточний день

Правильна відповідь: Б - Glacier Instant призначений для архівних даних з рідким доступом, але миттєвим отриманням при потребі.

4. Версіонування S3 увімкнено для бакету. Студент завантажив новий файл report.pdf поверх старого. Що відбудеться з попередньою версією?

А. Вона буде автоматично видалена Б. Вона буде переміщена до S3 Glacier В. Вона збережеться з новим Version ID і буде доступна для відновлення Г. Завантаження нової версії буде заблоковано

Правильна відповідь: В - версіонування S3 зберігає всі версії об'єкта з унікальними Version ID, жодна версія не видаляється автоматично.

5. Навіщо при налаштуванні AWS Budget обирається шаблон «Zero spend budget»?

А. Щоб автоматично видаляти всі ресурси при досягненні ліміту Б. Щоб отримати сповіщення одразу, як тільки AWS почне нараховувати хоч будь-яку суму (навіть 0,01 USD) В. Щоб отримати додатковий безкоштовний ліміт ресурсів Г. Щоб заморозити акаунт при перевищенні ліміту

Правильна відповідь: Б - Zero spend budget сигналізує про перший нарахований цент, що захищає від несподіваних витрат при випадковому виході за межі Free Tier.

РОЗДІЛ 4. REST API ТА КІБЕРБЕЗПЕКА

Відповідає лабораторним роботам No 8-9

4.1. Теоретичні питання для самоперевірки

- Що таке API? Поясніть аналогію з меню ресторану.
- Чим REST API відрізняється від звичайного браузерного запиту?
- Які чотири основні HTTP-методи (GET, POST, PUT/PATCH, DELETE) існують

і для чого використовується кожен?

- Що таке HTTP-статус-коди? Поясніть значення 200, 201, 404 та 500.
- Що таке JSON? Наведіть приклад об'єкта JSON з трьома полями різних типів.
- Що таке фішинг? Назвіть п'ять ознак фішингового листа або сайту.
- Що таке двофакторна аутентифікація (2FA)? Чому SMS-коди є менш надійними,

ніж застосунок-аутентифікатор?

- Чому забороняється зберігати API-ключі та паролі безпосередньо у коді JavaScript, що публікується на GitHub?

4.2. Додатковий практичний кейс: API-інтеграція для сайту «WeatherBoard»

Контекст кейсу

Жовтень 2025 р. Студентська команда розробляє сайт «WeatherBoard» - дошку оголошень для туристів з відображенням погоди у містах маршруту. Вони вирішили використати публічне API OpenWeatherMap. У процесі тестування через Postman отримано такі результати запитів:

URL запиту	Метод	Статус	Результат
api.openweathermap.org/data/2.5/weather?q=Kyiv	GET	401	Unauthorized
api.openweathermap.org/data/2.5/weather?q=Lviv&appid=APIKEY	GET	200	JSON з даними погоди
jsonplaceholder.typicode.com/posts/9999	GET	404	Порожній об'єкт {}

jsonplaceholder.typicode.com/posts	POST	201	Створений об'єкт з id
------------------------------------	------	-----	-----------------------

Крім того, один із студентів команди випадково завантажив файл config.js з API-ключем у публічний репозиторій на GitHub.

Завдання до кейсу

Завдання 4.1. Аналіз HTTP-запитів (35 балів)

- Поясніть причину кожного статус-коду з таблиці та що потрібно зробити для виправлення запиту зі статусом 401.
- Яка різниця між статусами 200 та 201? У якому з наведених запитів який статус є очікуваним і чому?
- Що означає повернення порожнього об'єкта {} у відповідь на GET-запит? Яку логіку повинен мати фронтенд при обробці такої відповіді?

Завдання 4.2. Реагування на витік API-ключа (35 балів)

- Визначте 5 кроків реагування на ситуацію з витоком API-ключа у порядку пріоритетності.
- Поясніть, де правильно зберігати API-ключ при розробці локально та після публікації на Netlify.
- Що таке файл .gitignore і який запис необхідно додати для захисту файлів із секретними даними?

Завдання 4.3. Проектування безпечної архітектури (30 балів)

Намалюйте (схематично або описово) архітектуру «WeatherBoard», де API-ключ OpenWeatherMap ніколи не потрапляє до браузера кінцевого користувача. Опишіть роль кожного компоненту: браузер клієнта, Netlify Function (або AWS Lambda), OpenWeatherMap API.

Індивідуальні варіанти (самостійна робота)

Оберіть варіант. Для заданого API-сценарію виконайте Завдання 4.1-4.2.

Вар.	API та сценарій	Статус для аналізу	Проблема безпеки
1	API НБУ: отримання курсів валют	200 (успіх) та 404 (валюта не знайдена)	Ключ у localStorage браузера
2	GitHub API: список репозиторіїв	200 та 403 (ліміт запитів)	Токен у git-коміті
3	PokeAPI: дані про покемонів	200 та 404 (ID не існує)	Ключ у src атрибуті script тегу

4	OpenWeatherMap: погода у місті	200 та 401 (немає ключа)	Ключ у README.md репозиторію
5	JSONPlaceholder: CRUD для постів	201 (POST) та 404 (DELETE)	Пароль бази даних у config.js
6	Stripe API (тестовий режим): платежі	200 та 400 (невірні дані)	Secret key у відкритому репозиторії
7	NASA API: фото дня	200 та 429 (перевищено ліміт)	API ключ у коментарі CSS
8	Telegram Bot API: надсилання повідомлень	200 та 500 (помилка сервера)	Bot token у JavaScript-файлі

4.3. Тест для самоперевірки (Розділ 4)

1. Студент отримав відповідь від API зі статусом 400 Bad Request. Що це означає?

А. Сервер тимчасово недоступний Б. Запит не знайдений на сервері В. Клієнт надіслав некоректний запит (наприклад, невірний формат даних або відсутнє обов'язкове поле) Г. Потрібна авторизація

Правильна відповідь: В - 400 Bad Request означає помилку на стороні клієнта: неправильний синтаксис, відсутні поля або невірний формат.

2. Яка різниця між HTTP-методами GET та POST?

А. GET швидший, POST надійніший Б. GET отримує дані (нічого не змінює), POST створює або передає дані для обробки В. GET лише для браузерів, POST лише для мобільних додатків Г. GET використовується для авторизації, POST - для всього іншого

Правильна відповідь: Б - основна семантична різниця: GET є ідемпотентним запитом даних, POST - передачею даних для обробки або створення ресурсу.

3. Студент отримав лист: «Терміново! Ваш AWS-акаунт заблоковано. Перейдіть за посиланням та введіть дані картки». Що слід зробити?

А. Перейти за посиланням і перевірити акаунт - це може бути справжнє повідомлення Б. Це фішинг: відкрити AWS Console напям (aws.amazon.com), не переходячи за посиланням у листі В. Надіслати відповідь з проханням пояснити ситуацію Г. Ігнорувати і видалити акаунт

Правильна відповідь: Б - термінові вимоги та прохання ввести платіжні дані є класичними ознаками фішингу. Перевіряти акаунт слід лише через пряме введення URL.

4. Чому пароль «Kov_basa!2025#drum» значно надійніший за «password2025»?

А. Він довший та містить різні типи символів (літери, цифри, спецсимволи), що збільшує простір перебору Б. Він містить українські слова, що ускладнює злам В. Він більш 8 символів - цього достатньо Г. Перший пароль неможливо запам'ятати, тому він надійніший

Правильна відповідь: А - надійність пароля визначається довжиною та різноманітністю символів: кожен додатковий тип символів збільшує час перебору у тисячі разів.

5. Студент додав рядок .env до файлу .gitignore. Що це означає на практиці?

А. Файл .env буде видалено з комп'ютера Б. Git ігноруватиме файл .env і не включатиме його до комітів - ключі залишаться лише локально В. Файл .env буде зашифровано перед завантаженням Г. Файл .env буде замінено порожнім файлом у репозиторії

Правильна відповідь: Б - .gitignore вказує Git ігнорувати перелічені файли. Файл .env залишається на локальному комп'ютері, але не потрапляє до репозиторію.

РОЗДІЛ 5. БЕЗСЕРВЕРНІ ОБЧИСЛЕННЯ ТА AWS CLI

Відповідає лабораторним роботам No 11-12

5.1. Теоретичні питання для самоперевірки

- Що таке Serverless (безсерверна) архітектура? У якому сенсі «без серверів» не означає відсутності серверів?
- Порівняйте традиційний сервер та AWS Lambda за моделлю оплати та масштабуванням.
- Що таке обробник події (lambda_handler) у Lambda-функції? Поясніть параметри event та context.
- Що таке CloudWatch? Яку інформацію можна знайти у рядку REPORT логів Lambda?
- Що таке AWS CLI? Назвіть три переваги роботи через CLI порівняно з веб-консоллю.
- Що таке Access Key і Secret Access Key у AWS? Чому їх не можна публікувати у відкритому доступі?
- Поясніть різницю між командами aws s3 cp та aws s3 sync.
- Що таке Free Tier у AWS Lambda? Скільки запитів на місяць надається безкоштовно?

5.2. Додатковий практичний кейс: Автоматизація звітів компанії TechReports

Контекст кейсу

Грудень 2025 р. Компанія TechReports щодня генерує 50 текстових звітів розміром від 2 до 10 КБ кожен. Раніше DevOps-інженер вручну завантажував їх до S3 через веб-консоль - це займало 30-40 хвилин щодня. Технічний директор поставив завдання: автоматизувати процес і скоротити ручну роботу до нуля. Наявні інструменти: AWS CLI, AWS Lambda, S3-бакет reports-techreports-2025.

Виміряні показники Lambda після тестового запуску:

Параметр	Тест 1	Тест 2	Тест 3
Duration (фактичний час)	245 мс	189 мс	312 мс
Billed Duration	300 мс	200 мс	400 мс
Max Memory Used	38 МБ	35 МБ	42 МБ
Memory Size (налаштування)	128 МБ	128 МБ	128 МБ

Завдання до кейсу

Завдання 5.1. Аналіз метрик Lambda (30 балів)

- Розрахуйте середній фактичний час виконання та середній Billed Duration за трьома тестами.
- Поясніть різницю між Duration та Billed Duration: чому AWS округлює час у більшу сторону?
- Оцініть ефективність використання пам'яті: яка частка виділеної пам'яті (128 МБ) реально використовується? Чи є сенс зменшити Memory Size до 64 МБ?
- Розрахуйте приблизну щомісячну вартість Lambda, якщо функція викликається 50 разів на день (1 500 разів на місяць) зі середнім Billed Duration 300 мс при тарифі 0,0000002 USD/запит + 0,00001667 USD за кожні 1 ГБ-секунду.

Завдання 5.2. Проектування CLI-скрипта (35 балів)

Напишіть послідовність команд AWS CLI (bash або Windows batch), яка:

- завантажує ВСІ файли з локальної папки /reports до S3-бакету reports-techreports-2025/daily/ за допомогою sync;
- перевіряє вміст бакету після синхронізації командою ls;
- видаляє файли, старіші за 30 днів (описово, якщо не знаєте точну команду).

Вкажіть, де безпечно зберігати Access Key для автоматичного запуску скрипта за розкладом.

Завдання 5.3. Lambda-функція для перевірки звітів (35 балів)

Напишіть псевдокод Lambda-функції на Python, яка:

- отримує на вхід об'єкт event з полями: report_name (рядок) та file_size_kb (число);
- повертає statusCode: 200 та повідомлення «Звіт прийнято», якщо розмір до 10 КБ включно;

- повертає statusCode: 400 та повідомлення «Звіт перевищує ліміт», якщо розмір більше 10 КБ;
- включає перевірку: якщо поле report_name відсутнє - повертати statusCode: 400 та «Назва звіту відсутня».

Індивідуальні варіанти (самостійна робота)

Оберіть варіант. Виконайте Завдання 5.1-5.3 для свого сценарію.

Вар.	Сценарій автоматизації	Кількість викликів/міс	Billed Duration (мс)	Пам'ять (МБ)
1	Обробка форми зворотного зв'язку	500	150	128
2	Стиснення зображень при завантаженні до S3	2 000	800	256
3	Щоденна розсилка email-дайджестів	30	1 200	128
4	Валідація даних форми реєстрації	10 000	50	64
5	Генерація PDF-квитанцій для замовлень	3 000	2 000	512
6	Моніторинг доступності сайту кожні 5 хвилин	8 640	200	128
7	Синхронізація даних між двома S3-бакетами	300	400	256
8	Автоматичне перейменування файлів за шаблоном	1 000	100	128

5.3. Тест для самоперевірки (Розділ 5)

1. Lambda-функція виконалась за 245 мс, але у логах вказано Billed Duration: 300 мс. Чому?

А. Lambda округлює вгору до найближчих 100 мс Б. 55 мс займає ініціалізація функції (cold start) В. AWS нараховує мінімум 300 мс незалежно від тривалості Г. Billed Duration і Duration завжди відрізняються на 55 мс

Правильна відповідь: А - Lambda округлює фактичний час виконання вгору до найближчого кратного 1 мс (раніше до 100 мс), що пояснює різницю.

2. Яка команда AWS CLI скопіює файл report.txt з локального комп'ютера до S3-бакету my-bucket у папку reports/?

А. aws s3 sync report.txt s3://my-bucket/reports/ Б. aws s3 cp report.txt s3://my-bucket/reports/report.txt В. aws s3 upload report.txt s3://my-bucket/reports/ Г. aws copy report.txt my-bucket/reports/

Правильна відповідь: Б - команда `cp (copy)` використовується для копіювання окремого файлу, `sync` - для синхронізації цілої папки.

3. Компанія хоче, щоб Lambda-функція автоматично запускала щоразу, коли новий файл завантажується до S3. Що для цього потрібно налаштувати?

А. Постійно запущений EC2-сервер, що перевіряє бакет Б. S3 Event Notification або S3 Trigger у Lambda - конфігурацію, що запускає функцію на подію PutObject В. CloudWatch Alarm для S3 Г. Netlify Webhook

Правильна відповідь: Б - S3 може надсилати події (triggers) до Lambda при завантаженні файлу. Це класична подійно-орієнтована безсерверна архітектура.

4. Студент виконав `aws configure`. Де на комп'ютері зберігаються введені Access Key та Secret Access Key?

А. У файлі `credentials.txt` на робочому столі Б. У файлах `~/.aws/credentials` та `~/.aws/config` (у прихованій папці домашньої директорії) В. Лише у пам'яті терміналу - після закриття видаляються Г. У реєстрі Windows

Правильна відповідь: Б - AWS CLI зберігає облікові дані у `~/.aws/credentials` (Unix/Mac) або `%USERPROFILE%\aws\credentials` (Windows). Цю папку не можна публікувати.

5. Яка основна перевага команди `aws s3 sync` над `aws s3 cp` при регулярному резервному копіюванні папки?

А. `sync` є швидшою командою для одного файлу Б. `sync` автоматично завантажує лише змінені або нові файли, не дублюючи вже завантажені В. `sync` шифрує файли перед завантаженням Г. `sync` є безкоштовним, `cp` - платним

Правильна відповідь: Б - `sync` порівнює локальні та хмарні файли і передає лише зміни, що суттєво економить трафік і час при щоденних резервних копіях.

ТЕСТИ ДЛЯ САМОПЕРЕВІРКИ ЗНАНЬ. (Розділи 2-3: GitHub Pages, Netlify, Amazon S3)

Для кожного питання оберіть одну правильну відповідь (А, Б або В).

Питання та варіанти відповіді	Правильна відповідь
1. Що таке Git у контексті веб-розробки? А. Хмарний хостинг для статичних сайтів від Microsoft Б. Розподілена система контролю версій для відстеження змін у коді В. Мова програмування для автоматизації деплою	Б - Git - розподілена система контролю версій, що дозволяє відстежувати зміни, повертатись до попередніх версій і працювати в команді.
2. Яка правильна послідовність команд для публікації змін на GitHub? А. <code>git commit</code> → <code>git add</code> → <code>git push</code>	В - Спочатку <code>git add</code> (додати до індексу), потім <code>git commit</code> (зафіксувати),

Б. git push → git commit → git add В. git add → git commit → git push	потім git push (відправити на сервер).
3. Навіщо репозиторій для GitHub Pages повинен мати назву виду username.github.io? А. Це зручне ім'я без технічної необхідності Б. GitHub автоматично публікує такий репозиторій за адресою https://username.github.io В. Лише такі репозиторії можна зробити публічними	Б - GitHub Pages вимагає саме такої назви, щоб автоматично опублікувати сайт за адресою https://username.github.io.
4. Що відбувається після команди git add . (з крапкою)? А. Всі файли видаляються з репозиторію Б. Всі змінені файли поточної директорії та підпапок додаються до індексу (staging area) В. Створюється новий порожній репозиторій	Б - Крапка означає «поточна папка і все всередині». Файли потрапляють до staging area і будуть включені до наступного коміту.
5. Студент отримав помилку 403 при виконанні git push. Найімовірніша причина: А. Файли занадто великі для GitHub Б. Немає прав запису до репозиторію або не налаштовано автентифікацію (токен/SSH) В. Репозиторій заблоковано через перевищення трафіку	Б - Помилка 403 (Forbidden) означає відсутність прав. Потрібно перевірити Personal Access Token або SSH-ключ.
6. Що означає термін «commit» у Git? А. Відправка файлів на хостинг Б. Зафіксований знімок стану файлів у певний момент часу з унікальним ідентифікатором В. Видалення старих версій файлів	Б - Коміт - це знімок (snapshot) стану репозиторію в певний момент. Кожен коміт має унікальний хеш і опис змін.
7. Чому чутливі до регістру назви файлів (style.css vs Style.CSS) є проблемою при публікації на GitHub Pages? А. GitHub забороняє великі літери у назвах файлів Б. Windows ігнорує регістр, але сервери Linux є чутливими - файл не знайдено, сайт відкривається без стилів В. CSS-файли не підтримуються на GitHub Pages	Б - Локально на Windows index.html знаходить і Style.CSS, і style.css. На Linux-сервері GitHub ці файли різні, тому сайт «ламається».
8. Що таке «Deploy Preview» у Netlify? А. Резервна копія останнього успішного деплою Б. Тимчасова URL-адреса для перегляду змін з Pull Request до злиття у головну гілку В. Метод оплати за хостинг через попередній перегляд рахунку	Б - Deploy Preview генерує унікальну URL для кожного PR, що дозволяє переглянути зміни у продакшн-середовищі без публікації на основний сайт.
9. Яка функція Netlify дозволяє найшвидше відновити сайт після помилкового деплою? А. git revert з наступним git push Б. Rollback - публікація попередньої версії одним кліком у розділі Deploys В. Видалення репозиторію та повторне завантаження файлів	Б - Rollback у Netlify Dashboard займає ~30 секунд і не потребує роботи з Git - лише вибрати попередній успішний деплой і натиснути «Publish deploy».
10. Що необхідно додати до тегу <form>, щоб Netlify автоматично обробляв форму? А. Атрибут action з URL серверного скрипту Б. Атрибути data-netlify="true" та name, а також приховане поле form-name	Б - Без data-netlify та прихованого поля form-name Netlify не розпізнає форму під час збірки і не

В. Тер <code><script></code> з підключенням Netlify SDK	буде обробляти відправлені дані.
11. Скільки часу займає автоматичне оновлення сайту на Netlify після git push? А. 10–15 хвилин Б. 30–90 секунд В. 1–2 години	Б - CI/CD pipeline Netlify від отримання push до публікації нової версії займає в середньому 30–90 секунд.
12. Чим Netlify відрізняється від GitHub Pages? А. Netlify підтримує лише платні акаунти Б. Netlify надає вбудований CI/CD, форми, функції та 100 ГБ/міс трафіку проти 10 ГБ у GitHub Pages В. GitHub Pages підтримує серверний код, Netlify - ні	Б - Netlify має більший набір функцій: CI/CD, Forms, Functions, Rollback, більший ліміт трафіку (100 ГБ vs 10 ГБ у GitHub Pages).
13. Що таке гілка (branch) у Git? А. Копія репозиторію на іншому сервері Б. Незалежна лінія розробки, що дозволяє працювати над функцією без впливу на основний код В. Резервна копія, яка створюється автоматично щодня	Б - Гілка дає змогу розробляти нові функції або виправляти помилки ізольовано, а потім злити (merge) зміни в основну гілку.
14. Яку команду використовують для клонування існуючого репозиторію з GitHub на локальний комп'ютер? А. git init Б. git clone <URL> В. git fetch origin	Б - git clone створює локальну копію всього репозиторію разом з усією історією комітів.
15. Що таке файл .gitignore? А. Файл з налаштуваннями інтерфейсу GitHub Б. Файл, у якому перелічені шляхи та маски файлів, які Git не повинен відстежувати та включати до комітів В. Журнал усіх проігнорованих помилок у коді	Б - .gitignore захищає від випадкового завантаження секретних даних (.env), артефактів збірки (node_modules/) та системних файлів.
16. Що таке AWS Free Tier і скільки він діє? А. Безмежний безкоштовний рівень для студентів Б. Набір безкоштовних ресурсів AWS для нових акаунтів терміном 12 місяців від реєстрації В. Знижка 50% на перший місяць використання AWS	Б - AWS Free Tier надає нові акаунти безкоштовними ресурсами (EC2 t2.micro, S3 5 ГБ та ін.) протягом 12 місяців.
17. Яка перша дія рекомендується після реєстрації в AWS для забезпечення безпеки акаунту? А. Запустити EC2-інстанс Б. Налаштувати Multi-Factor Authentication (MFA) для кореневого акаунту В. Завантажити AWS CLI	Б - Root user має необмежені права на всі ресурси. MFA є базовим захистом від несанкціонованого доступу.
18. Що таке IAM у AWS? А. Сервіс для зберігання файлів у хмарі Б. Identity and Access Management - сервіс керування правами доступу користувачів та сервісів до ресурсів AWS В. Інструмент моніторингу продуктивності хмарних додатків	Б - IAM дозволяє створювати користувачів, групи та ролі з мінімально необхідними правами, реалізуючи принцип найменших привілеїв.
19. Що таке Amazon S3-бакет? А. Віртуальний сервер для запуску застосунків Б. Контейнер для зберігання об'єктів (файлів) у хмарі з унікальним ім'ям в межах всього AWS	Б - S3-бакет - це контейнер об'єктів у хмарі. Ім'я бакету унікальне глобально.

В. База даних для зберігання структурованих даних	Об'єкти (файли) не мають обмежень за типом.
20. Студент завантажив файл до S3 і отримав помилку Access Denied при відкритті публічного URL. Найімовірніша причина: А. Файл завантажено у неправильний регіон Б. Не налаштовано Bucket Policy для публічного доступу або увімкнено Block Public Access В. URL неправильно сформовано браузером	Б - За замовчуванням S3 блокує публічний доступ. Для статичного хостингу потрібно вимкнути Block Public Access і додати Bucket Policy з s3:GetObject.
21. Що таке Bucket Policy в Amazon S3? А. Правила іменування файлів у бакеті Б. JSON-документ, що визначає права доступу до бакету та його об'єктів для користувачів і сервісів В. Налаштування максимального розміру файлів у бакеті	Б - Bucket Policy - це ресурсна IAM-політика у JSON, що керує доступом до бакету. Для публічного сайту додається Action: s3:GetObject з Principal: *.
22. Для якого типу даних найдоцільніше використовувати клас зберігання S3 Glacier Instant? А. Файли сайту, що завантажуються тисячами користувачів щодня Б. Архівні дані рідкого доступу, що можуть знадобитись у надзвичайній ситуації В. Логи сервера за поточний день	Б - Glacier Instant оптимальний для архівів з рідким, але потенційно терміновим доступом (резервні копії, медичні записи). Вартість ~0.004 USD/ГБ.
23. Що відбудеться зі старою версією файлу report.pdf в S3, якщо версіювання увімкнено і завантажуватиметься новий файл з такою ж назвою? А. Стара версія буде автоматично видалена Б. Стара версія збережеться з унікальним Version ID і буде доступна для відновлення В. Завантаження нового файлу буде заблоковано	Б - Версіювання S3 зберігає всі версії об'єкта. Кожна версія отримує унікальний Version ID. Попередню версію можна відновити у будь-який момент.
24. Яку різницю між класами S3 Standard та S3 Standard-IA (Infrequent Access)? А. Standard-IA зберігає файли лише у одному регіоні, Standard - у всіх Б. Standard-IA дешевший за зберігання, але дорожчий за кожне звернення і підходить для рідко використовуваних даних В. Standard-IA не підтримує версіювання	Б - Standard (0.023 USD/ГБ) - для часто використовуваних даних. Standard-IA (0.0125 USD/ГБ) дешевший, але кожне читання коштує додатково.
25. Навіщо при налаштуванні AWS Budget обирається шаблон «Zero spend budget»? А. Щоб автоматично видаляти всі ресурси при перевищенні ліміту Б. Щоб отримати сповіщення одразу при першому нарахуванні будь-якої суми (навіть 0.01 USD) В. Щоб отримати додатковий безкоштовний ліміт ресурсів	Б - Zero spend budget захищає від несподіваних витрат - сповіщення надходить при першому нарахуваному центі, поки не вийшли за межі Free Tier.
26. Що таке CDN (Content Delivery Network) у контексті AWS CloudFront? А. Система резервного копіювання даних по всьому світу Б. Географічно розподілена мережа серверів, що зберігають копії контенту ближче до кінцевого користувача В. Протокол для шифрування трафіку між S3 та браузером	Б - CDN скорочує затримку: замість звернення до S3 у США, користувач з України отримує файл з найближчого CDN-вузла у Європі.

27. Яка команда AWS CLI дозволяє перелічити вміст S3-бакету? А. <code>aws s3 list my-bucket</code> Б. <code>aws s3 ls s3://my-bucket/</code> В. <code>aws s3 show --bucket my-bucket</code>	Б - Команда <code>aws s3 ls</code> з префіксом <code>s3://</code> виводить список файлів та папок у вказаному бакеті або підпапці.
28. Що таке принцип найменших привілеїв (Least Privilege) у AWS IAM? А. Надавати всім користувачам однакові мінімальні права Б. Надавати кожному користувачу або сервісу лише ті права, які необхідні для виконання конкретних завдань В. Забороняти доступ до AWS без VPN	Б - Least Privilege обмежує потенційну шкоду від компрометації акаунту або помилки. Наприклад, IAM-користувач для S3 не повинен мати права запуску EC2.
29. Яка різниця між командами <code>aws s3 cp</code> та <code>aws s3 sync</code>? А. <code>cp</code> копіює лише один файл, <code>sync</code> - синхронізує цілу папку, завантажуючи лише нові та змінені файли Б. <code>cp</code> є безкоштовним, <code>sync</code> - платним В. <code>sync</code> шифрує дані, <code>cp</code> - ні	А - <code>cp</code> призначений для копіювання конкретного файлу, <code>sync</code> - для синхронізації директорій, завантажуючи лише дельту (нові/змінені).
30. Студент хоче налаштувати статичний веб-хостинг на S3. Які два обов'язкові кроки потрібно зробити? А. Включити версіонування та налаштувати Lifecycle Rules Б. Увімкнути Static Website Hosting у налаштуваннях бакету та відкрити публічний доступ через Bucket Policy В. Підключити CloudFront та налаштувати SSL-сертифікат	Б - Для статичного хостингу S3 необхідно: 1) увімкнути Static Website Hosting з вказанням <code>index.html</code>; 2) вимкнути Block Public Access та додати Bucket Policy.
31. Що таке Lifecycle Rule в Amazon S3? А. Правило для автоматичного резервного копіювання бакету Б. Автоматична політика переміщення або видалення об'єктів через певний час (наприклад, перемістити у Glacier через 90 днів) В. Правило іменування файлів для автоматичного сортування	Б - Lifecycle Rules автоматизують управління вартістю: старі файли переміщуються у дешевші класи (Standard → Standard-IA → Glacier) або видаляються.
32. Чому не можна завантажувати Access Key та Secret Access Key AWS у публічний репозиторій GitHub? А. GitHub блокує завантаження файлів з такими назвами Б. Боти сканують GitHub у реальному часі і можуть отримати доступ до AWS-акаунту, що призведе до великих витрат В. AWS автоматично анулює такі ключі	Б - Відомі випадки збитків у тисячі доларів - боти знаходять ключі протягом хвилин після публікації та запускають ресурси для майнінгу або спам-розсилок.
33. Що таке region (Region) у AWS? А. Окремий тип підписки AWS для певних країн Б. Географічна зона з кількома дата-центрами (Availability Zones), де AWS розгортає свої сервіси В. Налаштування мови інтерфейсу AWS Console	Б - AWS має понад 30 регіонів (eu-central-1, us-east-1 тощо). Ресурси ізольовані між регіонами. Для Free Tier рекомендується us-east-1.
34. Де безпечно зберігати Access Key для автоматичного запуску AWS CLI-скриптів за розкладом? А. У файлі <code>config.txt</code> поряд зі скриптом Б. У змінних середовища (environment variables) або через <code>aws configure</code> у файлі <code>~/.aws/credentials</code> В. У коментарі на початку bash-скрипту	Б - Credentials файл (<code>~/.aws/credentials</code>) або змінні середовища є безпечнішими за жорстке кодування у скрипті, особливо якщо скрипт може потрапити до Git.

35. Яку перевагу дає версіонування (Versioning) S3 порівняно зі звичайним зберіганням? А. Файли займають менше місця завдяки стисненню версій Б. Захист від випадкового видалення або перезапису - будь-яку версію об'єкта можна відновити В. Автоматичний переклад файлів між мовними версіями	Б - Версіонування дозволяє відновити будь-яку попередню версію файлу або видаленого об'єкта. Корисне при помилковому перезаписі або ransomware-атаці.
36. Студент виконав команду <code>git commit -m "fix"</code>. Що не так з цим комітом? А. Команда синтаксично невірна Б. Опис «fix» не є інформативним - незрозуміло, що саме виправлено і чому В. Команда не може бути виконана без <code>git add</code> перед нею	Б - Правило написання комітів: опис повинен пояснювати «що і чому». Приклад хорошого опису: «fix: виправлено відображення меню на мобільних пристроях».
37. Що таке Netlify CI/CD? А. Платна функція для автоматичного тестування JavaScript-коду Б. Автоматичний pipeline, що збирає та публікує сайт після кожного <code>git push</code> без ручних дій В. Система резервного копіювання для Netlify-сайтів	Б - CI/CD (Continuous Integration/Continuous Deployment) у Netlify автоматично будує сайт при кожному <code>push</code> до підключеного репозиторію.
38. Чим відрізняються S3 Standard-IA та S3 Glacier з точки зору швидкості доступу? А. Standard-IA надає доступ протягом хвилин, Glacier - миттєво Б. Standard-IA надає миттєвий доступ, Glacier (окрім Glacier Instant) - від кількох хвилин до годин В. Обидва класи надають однакову швидкість доступу	Б - Standard-IA - миттєво. Glacier Flexible Retrieval - 1-12 годин, Glacier Deep Archive - до 48 годин. Glacier Instant - миттєво, але дорожчий за зберігання.
39. Студент підключив репозиторій до Netlify, але автооновлення не спрацьовує після <code>push</code>. Найімовірніша причина: А. Netlify підтримує лише Manual Deploy Б. Не надано доступ Netlify до репозиторію у налаштуваннях GitHub (webhook не активовано) В. Репозиторій занадто великий для автоматичного деплою	Б - Для CI/CD Netlify реєструє webhook у GitHub. Якщо права не надано або webhook видалено - автоматичний деплой не спрацьовує.
40. Яке максимальне обмеження трафіку для сайту на GitHub Pages безкоштовного плану? А. 5 ГБ/місяць Б. 10 ГБ/місяць з м'яким обмеженням В. 100 ГБ/місяць	Б - GitHub Pages рекомендує не перевищувати 10 ГБ/місяць трафіку та 1 ГБ розміру репозиторію. Netlify надає 100 ГБ/місяць на безкоштовному плані.
41. Що таке Pull Request (PR) на GitHub? А. Команда для завантаження змін з GitHub на локальний комп'ютер Б. Механізм пропозиції змін з однієї гілки в іншу з можливістю перегляду коду командою перед злиттям В. Запит до GitHub API для отримання даних репозиторію	Б - Pull Request дозволяє команді переглянути та обговорити зміни перед їх злиттям у головну гілку - основа командної розробки.
42. Що станеться з даними у S3, якщо зупинити оплату або закрити AWS-акаунт? А. Дані зберігатимуться безкоштовно протягом 5 років Б. Після закриття акаунту всі дані в S3 будуть видалені назавжди В. Дані автоматично переміщуються до Glacier безкоштовно	Б - При закритті AWS-акаунту всі ресурси, включаючи S3-бакети та їх вміст, видаляються без можливості відновлення.

	Перед закриттям дані потрібно зберегти.
43. Для чого використовується команда git status? А. Показує статистику завантажень репозиторію на GitHub Б. Відображає стан робочої директорії: які файли змінені, доданому до staging area або не відстежуються В. Перевіряє підключення до GitHub	Б - git status є першою командою для діагностики: показує змінені, staged та untracked файли перед виконанням git add та git commit.
44. Чим відрізняється публічний (public) бакет S3 від приватного (private)? А. Публічний бакет платний, приватний - безкоштовний Б. Публічний бакет доступний будь-якому користувачу Інтернету без автентифікації; приватний - лише авторизованим В. Публічний бакет підтримує версіонування, приватний - ні	Б - Публічний S3-бакет використовується для статичних сайтів або публічних ресурсів. Приватний - для резервних копій, конфіденційних даних, де потрібна IAM-автентифікація.
45. Що таке AWS Budget і для чого він використовується студентом? А. Інструмент для планування архітектури хмарного застосунку Б. Сервіс контролю витрат з можливістю отримання email-сповіщень при досягненні порогу витрат В. Функція автоматичного масштабування ресурсів при перевантаженні	Б - AWS Budget дозволяє встановити ліміт витрат і отримати сповіщення. Для студентів рекомендується Zero Spend Budget - сповіщення при першому нарахованому центі.
46. Що означає команда git log у Git? А. Записує помилки Git у системний журнал Б. Показує хронологічний список комітів з їх хешами, авторами та повідомленнями В. Авторизує користувача у GitHub	Б - git log відображає всю історію комітів. Корисно для пошуку конкретного коміту, до якого потрібно повернутись, або для перегляду хто і коли робив зміни.
47. Студент хоче переглянути зміни між поточним станом файлу та останнім комітом. Яку команду використати? А. git compare HEAD Б. git diff В. git show --changes	Б - git diff показує строкові відмінності між робочою директорією та останнім комітом (або між довільними комітами).
48. Яку проблему вирішують Netlify Forms для статичних сайтів? А. Додають анімацію до HTML-форм Б. Дозволяють отримувати дані форм без серверного коду або бекенду, обробляючи їх на боці Netlify В. Автоматично валідують поля форми у реальному часі	Б - Статичний сайт не має серверного коду. Netlify Forms обробляє відправлені дані та зберігає їх у Dashboard - форма «оживає» без написання бекенду.
49. Яке значення атрибуту type у тезі input для поля електронної пошти? А. type="mail" Б. type="email" В. type="text" class="email"	Б - type="email" активує вбудовану браузерну валідацію формату адреси та відповідну клавіатуру на мобільних пристроях.
50. Що означає скорочення CI/CD у контексті сучасного веб-деплою? А. Client Interface / Content Delivery	Б - CI/CD - практика автоматичного тестування (CI) та публікації (CD) кожного оновлення коду.

Б. Continuous Integration / Continuous Deployment - безперервна інтеграція та безперервне розгортання В. Code Inspection / Code Distribution	Netlify та GitHub Actions є прикладами CI/CD для статичних сайтів.
---	--

ТЕСТИ ДО САМОПРЕВІРКИ ЗНАНЬ. (Розділи 4-5: REST API, Кібербезпека, AWS Lambda, AWS CLI)

Для кожного питання оберіть одну правильну відповідь (А, Б або В).

Питання та варіанти відповіді	Правильна відповідь
1. Що таке REST API? А. Мова програмування для розробки веб-застосунків Б. Архітектурний стиль взаємодії між клієнтом і сервером через HTTP з використанням стандартних методів В. Система управління базами даних	Б - REST (Representational State Transfer) - стандарт взаємодії. Клієнт надсилає HTTP-запити (GET/POST/PUT/DELETE), сервер повертає JSON або XML.
2. Що означає HTTP-статус-код 404? А. Запит успішно оброблено Б. Запитаний ресурс не знайдено на сервері В. Сервер тимчасово недоступний	Б - 404 Not Found - сервер доступний, але запитаний ресурс (URL, ідентифікатор) не існує. Потрібно перевірити правильність URL або ID.
3. Для чого використовується HTTP-метод GET? А. Для створення нового ресурсу на сервері Б. Для отримання (читання) даних без зміни стану сервера В. Для видалення ресурсу	Б - GET є ідемпотентним - багаторазовий виклик не змінює стан. Використовується для отримання списків, деталей об'єкта тощо.
4. Студент отримав відповідь API зі статусом 401. Що потрібно зробити? А. Перевірити підключення до Інтернету Б. Додати або виправити API-ключ/токен авторизації у заголовку запиту В. Чекати, доки сервер перезавантажиться	Б - 401 Unauthorized означає відсутність або недійсний токен автентифікації. Потрібно передати API-ключ у заголовку Authorization або як параметр запиту.
5. Яка різниця між HTTP-статусами 200 та 201? А. Різниць немає - обидва означають успіх Б. 200 OK - успішне читання або оновлення; 201 Created - успішне створення нового ресурсу В. 200 - часткова відповідь, 201 - повна відповідь	Б - 200 повертається на GET/PUT. 201 повертається на POST і означає, що новий ресурс створено. У відповіді зазвичай є Location з URL нового об'єкта.
6. Що таке JSON? А. Мова розмітки для створення веб-сторінок Б. Текстовий формат обміну даними на основі JavaScript-об'єктів, зручний для передачі між клієнтом та сервером В. Система шифрування даних у REST API	Б - JSON (JavaScript Object Notation) - легкий текстовий формат: {"key": value}. Підтримується всіма мовами. Є стандартом обміну даних у REST API.
7. Що таке фішинг? А. Метод шифрування трафіку між клієнтом і сервером Б. Вид соціальної інженерії, при якому зловмисник імітує довіреного суб'єкта (банк, AWS) для викрадення даних	Б - Фішинг маскується під офіційні листи або сайти. Ознаки: терміновість, загрози, посилання на

В. Тип SQL-ін'єкції у веб-форми	підозрілі домени, прохання ввести паролі.
8. Чому API-ключі не можна зберігати у JavaScript-файлах, що публікуються на GitHub? А. JavaScript не підтримує рядкові константи для ключів Б. Боти сканують публічні репозиторії і можуть викрасти ключі протягом хвилин для несанкціонованого використання API В. GitHub автоматично шифрує та видаляє такі файли	Б - Крім того, ключ у JS клієнтського коду видимий у DevTools браузера будь-кому. Для захисту ключів використовують серверні функції (Netlify Functions, Lambda).
9. Що таке двофакторна аутентифікація (2FA)? А. Вхід з двох різних браузерів для підтвердження особи Б. Додатковий рівень захисту, що вимагає підтвердження особи другим фактором (ОТР-код) окрім пароля В. Система, яка дозволяє увійти без пароля через biometrics	Б - 2FA унеможливлює вхід лише за паролем. Навіть якщо пароль скомпрометовано, зловмисник не отримає доступ без другого фактора (ОТР-застосунок, SMS).
10. Де правильно зберігати API-ключ при розробці на Netlify у продакшені? А. У файлі api_keys.json у корені проекту Б. У змінних середовища (Environment Variables) у налаштуваннях Netlify - вони доступні лише серверним функціям В. У HTML-атрибуті data-key тегу script	Б - Environment Variables Netlify передаються лише серверним Netlify Functions і не потрапляють до браузера. Це безпечний спосіб зберігання секретів.
11. Студент отримав відповідь API: {} (порожній JSON-об'єкт) на GET-запит. Що це означає? А. Сервер повернув помилку і не зміг обробити запит Б. Запитаний ресурс не існує або сервер повернув порожній результат - фронтенд повинен обробити цей стан В. JSON-відповідь зашифрована і потребує декодування	Б - Порожній об'єкт {} не є помилкою сервера - ресурс просто відсутній або порожній. Фронтенд повинен перевіряти наявність даних перед відображенням.
12. Що таке файл .env і для чого він використовується? А. Файл конфігурації браузера Б. Файл для зберігання змінних середовища (API-ключів, паролів) локально без включення до Git В. Файл зі стилями для development-середовища	Б - .env зберігає секретні налаштування у форматі KEY=VALUE. Обов'язково додається до .gitignore, щоб не потрапити до публічного репозиторію.
13. Для чого використовується HTTP-метод DELETE? А. Для очищення кешу браузера Б. Для видалення вказаного ресурсу на сервері В. Для скасування попереднього POST-запиту	Б - DELETE є стандартним REST-методом для видалення ресурсу. Наприклад, DELETE /posts/5 видаляє пост з id=5.
14. Що означає статус 400 Bad Request від API? А. Сервер недоступний через технічні проблеми Б. Клієнт надіслав некоректний запит: невірний формат, відсутні обов'язкові поля або невалідні дані В. Потрібна авторизація для доступу до ресурсу	Б - 400 - помилка на стороні клієнта. Потрібно перевірити структуру тіла запиту, обов'язкові поля та формат даних (наприклад, рядок замість числа).
15. Яку ознаку фішингового листа ілюструє повідомлення: «Терміново! Ваш акаунт заблокований. Перейдіть за посиланням і введіть дані картки»? А. Лист написаний правильною граматикою - це ознака офіційного повідомлення	Б - Легітимні компанії не вимагають платіжних даних через лист. Перевірити стан акаунту потрібно лише через пряме введення URL в адресному рядку.

Б. Штучна терміновість і вимога ввести платіжні дані - класичні ознаки фішингу В. Прохання перейти за посиланням є нормальною практикою для будь-яких компаній	
16. Чому SMS-коди є менш надійними для 2FA порівняно з TOTP-застосунком? А. SMS-коди діють 60 секунд, TOTP - 30 секунд Б. SMS-коди вразливі до SIM-swapping та перехоплення, TOTP-коди генеруються локально без передачі мережею В. SMS потребує підключення до Інтернету, TOTP - ні	Б - SIM-swapping дозволяє зловмиснику отримати SMS-код. TOTP (Google Authenticator, Authy) генерує коди локально і не вразливий до перехоплення.
17. Що таке Postman і для чого він використовується? А. Хмарний сервіс для зберігання API-ключів Б. Інструмент для тестування API-запитів: відправки HTTP-запитів і перегляду відповідей без написання коду В. Бібліотека JavaScript для роботи з REST API	Б - Postman дозволяє тестувати GET/POST/PUT/DELETE запити, переглядати заголовки, тіло відповіді та статус-коди без написання клієнтського коду.
18. Яка перша дія при виявленні витоку API-ключа у публічному репозиторії? А. Видалити репозиторій Б. негайно анулювати (revoke) скомпрометований ключ та згенерувати новий В. Встановити пароль на репозиторій	Б - Перший пріоритет - анулювати ключ, щоб запобігти подальшому використанню. Видалення коміту не є достатнім - боти вже могли зберегти ключ.
19. Що таке заголовок Authorization у HTTP-запиті? А. Ім'я файлу, який запитується з сервера Б. HTTP-заголовок для передачі токена або API-ключа для автентифікації запиту В. Поле для вказання автора REST API	Б - Authorization: Bearer <token> або Authorization: ApiKey <key> - стандартні способи передачі облікових даних у REST API без включення їх в URL.
20. Чому пароль «P@ssw0rd!2026#Tech» надійніший за «password123»? А. Він написаний з великої літери Б. Він довший і поєднує великі/малі літери, цифри та спецсимволи, що збільшує простір перебору в мільярди разів В. Він містить рік, що є унікальним для кожного користувача	Б - Складність пароля визначається ентропією. Кожен тип символів множить час brute-force. «password123» є у топ-100 найгірших паролів - зламується за секунди.
21. Що таке безсерверна (serverless) архітектура? А. Архітектура, де застосунок запускається без будь-яких серверів Б. Модель, де провайдер (AWS) керує серверною інфраструктурою, а розробник платить лише за час виконання функцій В. Архітектура лише для статичних сайтів без динамічного контенту	Б - Serverless не означає відсутності серверів - вони є, але керуються провайдером. Розробник пише функції і платить лише за фактичні виклики.
22. Що таке AWS Lambda? А. Сервіс для зберігання великих файлів у хмарі Б. Безсерверний обчислювальний сервіс, що виконує код у відповідь на події без управління серверами В. Мова програмування від AWS для хмарної розробки	Б - AWS Lambda виконує функції (Python, Node.js, Java) у відповідь на тригери: HTTP через API Gateway, завантаження до S3, розклад CloudWatch Events тощо.
23. Lambda-функція виконалась за 245 мс, але у логах вказано Billed Duration: 300 мс. Чому? А. 55 мс займає ініціалізація контейнера (cold start)	Б - Lambda округлює Duration вгору: 245 мс → 300 мс (до найближчих 100 мс для старих, або до 1 мс для нових функцій). Це

Б. AWS Lambda округлює час виконання вгору до найближчих 100 мс (або 1 мс у нових версіях) В. AWS нараховує мінімум 300 мс незалежно від тривалості	враховується при розрахунку витрат.
24. Що таке параметр event у функції lambda_handler(event, context)? А. Об'єкт з метаданими Lambda-контейнера Б. Словник із даними, що передаються функції від триггера (HTTP-запит, S3-подія тощо) В. Список усіх попередніх викликів функції	Б - event містить вхідні дані: для API Gateway - HTTP-запит (path, body, headers), для S3 - інформацію про завантажений файл. context містить метадані виконання.
25. Яка основна перевага AWS Lambda порівняно з постійно запущеним EC2-сервером для обробки нерегулярних запитів? А. Lambda підтримує більше мов програмування ніж EC2 Б. Lambda тарифікується лише за фактичний час виконання - EC2 нараховується постійно, навіть у простої В. Lambda підключена до Інтернету, EC2 - ні	Б - Для нерегулярного навантаження Lambda значно дешевша: платите лише за фактичні виклики. EC2 коштує ~0.0116 USD/год навіть без навантаження.
26. Що таке AWS CloudWatch у контексті Lambda? А. Сервіс відеоспостереження для дата-центрів AWS Б. Сервіс моніторингу та логування: зберігає логи Lambda-функцій, метрики виконання та дозволяє налаштовувати алерти В. Інструмент для перегляду витрат на AWS-ресурси	Б - CloudWatch Logs зберігає stdout Lambda. Рядок REPORT у логах містить Duration, Billed Duration, Max Memory Used - ключові метрики для оптимізації.
27. Яка команда AWS CLI копіює окремий файл до S3-бакету? А. aws s3 sync report.txt s3://my-bucket/ Б. aws s3 cp report.txt s3://my-bucket/reports/report.txt В. aws s3 upload report.txt my-bucket	Б - aws s3 cp - команда для копіювання окремого файлу. sync призначений для синхронізації цілої папки. Шлях у S3 вказується з префіксом s3://.
28. Де AWS CLI зберігає Access Key після виконання aws configure? А. У файлі credentials.txt на робочому столі Б. У прихованій папці ~/.aws/credentials (Linux/Mac) або %USERPROFILE%\aws\credentials (Windows) В. Лише у пам'яті терміналу - після закриття видаляються	Б - ~/.aws/credentials є стандартним місцем зберігання облікових даних CLI. Файл не повинен потрапляти до Git-репозиторію.
29. Яка ключова перевага aws s3 sync над aws s3 cp при регулярному резервному копіюванні? А. sync шифрує файли перед завантаженням Б. sync завантажує лише нові та змінені файли, не дублюючи вже синхронізовані В. sync є безкоштовним, cp - платним	Б - sync порівнює локальні та хмарні файли за розміром і датою. Для щоденних резервних копій це суттєво економить час та вартість передачі даних.
30. Що таке тригер (trigger) у AWS Lambda? А. Помилка, що зупиняє виконання функції Б. Подія або сервіс (S3, API Gateway, CloudWatch Events), що запускає виконання Lambda-функції В. Параметр для встановлення ліміту пам'яті функції	Б - Тригери дозволяють будувати подійно-орієнтовану архітектуру: завантаження файлу до S3 → Lambda обробляє файл автоматично без ручного запуску.
31. Скільки безкоштовних запитів надає AWS Lambda на місяць у рамках Free Tier? А. 100 000 запитів Б. 1 000 000 запитів В. 10 000 запитів	Б - AWS Lambda Free Tier включає 1 мільйон безкоштовних запитів та 400 000 ГБ-секунд обчислень на місяць - незалежно від того, минув рік Free Tier чи ні.

32. Що таке cold start у AWS Lambda? А. Запуск Lambda-функції о 6 ранку за розкладом Б. Додаткова затримка при першому виклику функції після простою - AWS ініціалізує новий контейнер В. Помилка при запуску функції без достатньої пам'яті	Б - Cold start виникає при першому виклику або після тривалого простою. Може додавати 100-500 мс. Зменшується Provisioned Concurrency або збільшенням пам'яті.
33. Що поверне Lambda-функція з statusCode: 400? А. Успішне виконання без даних Б. Помилку клієнта - наприклад, відсутні обов'язкові поля або невалідні дані у запиті В. Помилку сервера при виключенні у коді функції	Б - 400 повертається навмисно у логіці функції при невалідних вхідних даних. 500 - неочікувана помилка. 200 - успіх.
34. Що таке Netlify Functions і як вони пов'язані з AWS Lambda? А. Платний додаток Netlify для хмарних обчислень Б. Serverless-функції Netlify, що виконуються на базі AWS Lambda під капотом і дозволяють безпечно виконувати серверний код В. Плагіни для автоматизації деплою на Netlify	Б - Netlify Functions спрощують роботу з Lambda: не потрібен AWS-акаунт, функції живуть у папці /netlify/functions проекту і деплоються автоматично.
35. Яка команда AWS CLI переглядає список файлів у певній папці S3-бакету? А. aws s3 dir s3://my-bucket/folder/ Б. aws s3 ls s3://my-bucket/folder/ В. aws s3 show my-bucket --path folder/	Б - aws s3 ls з URI s3://bucket/prefix/ показує вміст папки. Без префіксу - список усіх бакетів. Прапор --recursive виводить всі файли рекурсивно.
36. Що міститься у рядку REPORT у логах AWS Lambda (CloudWatch)? А. Список всіх змінних середовища функції Б. Метрики виконання: Duration, Billed Duration, Memory Size, Max Memory Used, Init Duration В. Повний вміст об'єкта event, переданого функції	Б - Рядок REPORT надає ключові метрики для оптимізації функції: якщо Max Memory Used значно менше Memory Size - можна зменшити виділену пам'ять і знизити витрати.
37. Для чого використовується AWS API Gateway у зв'язі з Lambda? А. Для зберігання результатів виконання Lambda-функцій Б. Для прийому HTTP-запитів із браузера та передачі їх Lambda-функції як тригер В. Для шифрування трафіку між Lambda та S3	Б - API Gateway надає Lambda публічний HTTP/HTTPS endpoint. Браузер → API Gateway → Lambda → відповідь. Це стандартна архітектура serverless REST API.
38. Студент налаштував Lambda-тригер на S3 подію PutObject. Що відбудеться після завантаження файлу до бакету? А. Файл буде переміщено до Glacier Б. AWS автоматично запустить Lambda-функцію і передасть їй інформацію про завантажений файл через об'єкт event В. S3 надішле email-повідомлення на адресу адміністратора	Б - S3 Event Notification → Lambda - класична безсерверна архітектура. event['Records'][0]['s3']['object']['key'] містить ім'я завантаженого файлу.
39. Що таке Access Key ID та Secret Access Key у AWS CLI? А. Логін та пароль для веб-консолі AWS Б. Пара облікових даних для програмного доступу до AWS API через CLI або SDK без веб-консолі В. Ключі для шифрування файлів у S3	Б - Access Key ID (публічний ідентифікатор) + Secret Access Key (секретний) - аналог логін/пароль для AWS API. Secret Key показується лише один раз при створенні.

40. Яка максимальна тривалість виконання Lambda-функції за замовчуванням? А. 15 секунд Б. 3 секунди (налаштування за замовчуванням, максимум 15 хвилин) В. 1 хвилина	Б - За замовчуванням timeout Lambda = 3 секунди. Максимальний - 15 хвилин. Для довгих завдань потрібно збільшити timeout у налаштуваннях функції.
41. Що таке Serverless Framework або SAM (AWS Serverless Application Model)? А. Безкоштовний план AWS для студентів Б. Інструменти для визначення та деплою безсерверних застосунків як коду (Infrastructure as Code) В. Бібліотека Python для написання Lambda-функцій	Б - SAM/Serverless Framework дозволяють описати Lambda, API Gateway, S3 у YAML-файлі та деплоїти автоматично. Це Infrastructure as Code для serverless.
42. Що відбудеться, якщо Lambda-функція перевищить встановлений timeout? А. Функція продовжить виконання у фоновому режимі Б. Виконання буде примусово зупинено і у CloudWatch з'явиться помилка Task timed out В. AWS автоматично збільшить timeout для цього виклику	Б - При перевищенні timeout Lambda повертає помилку. Якщо це API Gateway виклик - клієнт отримає 502/504. Потрібно або оптимізувати код, або збільшити timeout.
43. Як розрахувати вартість AWS Lambda за місяць при 1000 викликів зі середнім Billed Duration 200 мс і 128 МБ пам'яті? А. Множити кількість викликів на тариф за запит: $1000 \times 0.0000002 \text{ USD}$ Б. Рахувати обидві складові: вартість запитів (1000×0.0000002) + вартість обчислень ($1000 \times 0.2 \text{ сек} \times 0.125 \text{ ГБ} \times 0.00001667 \text{ USD/ГБ-сек}$) В. Платити фіксований тариф 0.01 USD/місяць	Б - Lambda тарифікується за двома складовими: кількість запитів ($0.0000002 \text{ USD/запит}$) та обчислювальний час у ГБ-секундах ($0.00001667 \text{ USD/ГБ-сек}$).
44. Що таке змінні середовища (Environment Variables) у Lambda? А. Параметри, що вказують у якому регіоні AWS виконується функція Б. Пари ключ-значення, що налаштовуються у консолі та доступні у коді через os.environ - спосіб передачі секретів без жорсткого кодування В. Налаштування операційної системи контейнера Lambda	Б - Environment Variables Lambda (os.environ['DB_URL']) - безпечний спосіб передавати API-ключі, URLs, конфігурацію без включення у код функції.
45. Що таке Provisioned Concurrency у AWS Lambda? А. Ліміт кількості одночасних викликів функції Б. Функція попереднього прогріву контейнерів Lambda для усунення cold start при критичних до затримки застосунках В. Автоматичне масштабування Lambda при пікових навантаженнях	Б - Provisioned Concurrency тримає N контейнерів завжди готовими до виклику - cold start відсутній. Використовується для API з вимогами до latency < 100 мс.
46. Яку команду AWS CLI використовують для синхронізації локальної папки /reports із S3-бакетом? А. aws s3 cp /reports s3://my-bucket/reports/ --recursive Б. aws s3 sync /reports s3://my-bucket/reports/ В. aws s3 upload --dir /reports s3://my-bucket/	Б - aws s3 sync автоматично визначає нові та змінені файли і завантажує лише їх. При регулярних резервних копіях це значно ефективніше ніж cp --recursive.
47. Яку роль відіграє IAM Role у AWS Lambda? А. Визначає, хто може викликати Lambda-функцію ззовні	Б - Execution Role - IAM Role, що призначається Lambda. Якщо

Б. Надає Lambda-функції права доступу до інших AWS-сервісів (S3, DynamoDB, CloudWatch) під час виконання В. Встановлює ліміт на кількість виконань функції на день	функція записує до S3, Role повинна мати s3:PutObject. Принцип Least Privilege застосовується і тут.
48. Що таке DLQ (Dead Letter Queue) у AWS Lambda? А. Черга пріоритетних завдань для Lambda Б. Сервіс для зберігання невдалих викликів Lambda, що дозволяє аналізувати та повторно обробляти помилки В. Резервна Lambda-функція, що запускається при збоях основної	Б - DLQ (SQS або SNS) отримує повідомлення про невдалі виклики Lambda після вичерпання спроб повтору. Дозволяє не втратити дані та дослідити причину збою.
49. Для чого використовується прапор --delete у команді aws s3 sync? А. Видаляє S3-бакет після синхронізації Б. Видаляє з S3 файли, яких немає у локальній папці - забезпечує точну дзеркальну копію В. Видаляє локальні файли після успішного завантаження до S3	Б - aws s3 sync --delete робить S3 точною дзеркальною копією джерела: файли, видалені локально, видаляються і в S3. Використовувати обережно.
50. Що таке AWS SAM CLI і яку задачу він вирішує? А. Інтерфейс командного рядка для управління S3-бакетами Б. Інструмент для локального тестування Lambda-функцій і деплою serverless-застосунків без завантаження на AWS В. CLI для управління IAM-користувачами та правами	Б - SAM CLI (sam local invoke, sam local start-api) дозволяє запускати Lambda локально для відлагодження, що суттєво прискорює цикл розробки.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Проектування інформаційних систем: методичні вказівки /Уклад. О.В.Ізмайлова, – Київ : КНУБА, 2023. – 28 с. <https://repository.knuba.edu.ua/items/f1ee0cf1-e830-43a7-b33e-dc2af6f40378>
2. Ніколюк П. К. Моделювання систем : навч. посіб. для здобувачів вищої освіти спеціальності 122 Комп'ютерні науки. Вінниця: ДонНУ, 2023. 228 с.
3. Міца О.В., Лавер В.О. Системний аналіз : навч.-метод. посіб. / О.В. Міца, В.О. Лавер. Ужгород : вид-во ПП «АУТДОР - ШАРК», 2021. 63 с.
4. Ізмайлова, О. В. Проектування інформаційних систем: навч. посібник : для студ. галузі знань 12 "Інформаційні технології" - Київ : КНУБА, 2022. - 87 с.
5. Литвин В. В., Пасічник В. В., Шаховська Н. Б. Проектування інформаційних систем: Навчальний посібник. Львів: «Магнолія-2006», 2021. 380 с.
6. Новіков Ф. В. Техніко-економічне обґрунтування сучасних технологій виробництва : навч. посібник / Ф. В. Новіков, Д. Ф. Новіков, О. А. Єрмоленко та ін. – Дніпро: ЛІРА, 2022. – 256 с. Режим доступу: <http://www.repository.hneu.edu.ua/handle/123456789/28117>
7. Сучасні інформаційні технології та системи [Електронний ресурс] : монографія / Н. Г. Аксак, Л. Е. Гризун, О. В. Щербаков [та ін.] ; за заг.ред. Пономаренка В. С. – Харків : ХНЕУ ім. С. Кузнеця, 2022. – 270 с. Режим доступу: <http://repository.hneu.edu.ua/handle/123456789/29233>
8. Теорія систем та системний аналіз : курс лекцій / В. В. Тютюнник, О. О. Писклакова. – Харків : Друкарня Мадрид, 2020. – 108 с.

9. Рубан І. В. Особливості створення системи підтримки прийняття антикризових рішень в умовах невизначеності вхідної інформації при надзвичайних ситуаціях / І. В. Рубан, В. В. Тютюник, О. О. Тютюник // Сучасні інформаційні технології у сфері безпеки та оборони. – 2021. – №1(40). – С. 75-84. Режим доступу: <http://repository.hneu.edu.ua/handle/123456789/25600>

10. Виклюк Я. І. Моделювання складних систем : посібник / Я. І. Виклюк, Р.М. Камінський, В. В. Пасічник. – Львів : Вид-во «Новий Світ –2000», 2020. – 404 с., 2023. – 39 с.

11. Зелінська О. В. Інформаційні системи та технології в галузі : навч. посіб. / О. В. Зелінська, Н. А. Потапова, Л. О. Волонтир. – Вінниця : ВНАУ, 2020. – 253 с

12. Крєневич А. П. Алгоритми і структури даних : підручник / А. П. Крєневич. – Київ : ВПЦ «Київський Університет», 2021. – 200 с

13. Методологія інформаційних систем та баз даних: теоретичний і практичний підходи : навч. посіб. / уклад. Ю. О. Ушенко, М. Л. Ковальчук, М. С. Гавриляк, А. Л. Нєгрич. – Чернівці : Чернівецький нац. ун-т ім. Ю. Федьковича, 2021. – 240 с.

14. Мінухін С. Метод збільшення продуктивності Apache Spark на основі сегментування даних і налаштувань конфігураційних параметрів [Електронний ресурс] / С. Мінухін, Н. Коптілов // Сучасний стан наукових досліджень та технологій промисловості. – 2024. – № 1(27). – С. 128–139. – Режим доступу : <https://doi.org/10.30837/ITSSI.2024.27.128>

15. Kief Morris. Infrastructure as Code. O'Reilly Media, 2020 – 462 p. Michael Kreef, Mitchell Hashimoto. Terraform Cookbook: Efficiently Define, Launch, and Manage Infrastructure as Code Across Various Cloud Platforms. Packt Publishing, 2020 – 366 p.

16. Thomas Kurian Tekanat. Datadog Cloud Monitoring Quick Start Guide. Packt Publishing, 2021 – 318 p.

17. Brendan Burns, Joe Beda, Kelsey Hightower, Lachlan Evenson. Kubernetes: Up and Running, 3rd Edition. O'Reilly Media, 2022 – 540 p.

18. Raphael Hertzog, Jim O’Gorman, Mati Aharoni, Joe O’Gorman. Kali Linux Revealed: Mastering the Penetration Testing Distribution. OffSec Press, 2021 – 346 p.

Інтернет-ресурси:

1. Академія web розробки MDN [Електронний ресурс]. – Режим доступу: <https://developer.mozilla.org/>

2. HTML Tutorial [Електронний ресурс]. – Режим доступу: <https://www.tutorialspoint.com/html/index.htm>

3. CSS Tutorial [Електронний ресурс]. – Режим доступу: <https://www.tutorialspoint.com/css/index.htm>

4. Javascript Tutorial [Електронний ресурс]. – Режим доступу: <https://www.tutorialspoint.com/javascript/index.htm>

5. SQL Tutorial [Електронний ресурс]. – Режим доступу: <https://www.tutorialspoint.com/sql/index.htm>

6. Навчальна платформа Coursera, Secure Software Design [Електронний ресурс]. – Режим доступу до ресурсу: <https://www.coursera.org/programs/program-natsional-nii-tiekhnichnii-univiersitiet-ukrayini-kiyivs-kii/specializations/secure-software-design>

7. Навчальна платформа Coursera, Information Systems [Електронний ресурс]. – Режим доступу до ресурсу: <https://www.coursera.org/programs/program-natsional-nii-tiekhnichnii-univiersitiet-ukrayini-kiyivs-kii/specializations/information-systems>

8. Навчальна платформа Udemy, Software Architecture & System Design Practical Case Studies [Електронний ресурс]. – Режим доступу до ресурсу: <https://ua.udemy.com/course/software-architecture-system-design-practical-case-studies>
9. Навчальна платформа Udemy, Software Architecture & Design of Modern Large Scale Systems [Електронний ресурс]. – Режим доступу до ресурсу: <https://ua.udemy.com/course/software-architecture-design-of-modern-large-scale-systems/>
10. Навчальна платформа Udemy, Software Architecture & Technology of Large Scale Systems [Електронний ресурс]. – Режим доступу до ресурсу: <https://ua.udemy.com/course/developer-to-architect/>
11. Навчальна платформа Wintellectnow, Architecting Distributed Cloud Applications [Електронний ресурс]. – Режим доступу до ресурсу: <https://www.wintellectnow.com/Home/SeriesDetail?seriesId=architecting-distributed-cloud-applications>

Інформаційні ресурси

1. Електронна бібліотека ЛьвДУВС: URL: <https://www1.lvduvs.edu.ua/elektronnyi-kataloh>
2. Державна науково-технічна бібліотека України. URL: <http://www.gntb.n-t.org>.
Сайт національної бібліотеки України імені В.І.Вернадського. URL: <http://www.nbuv.gov.ua>
3. Бібліотека ПДАУ. URL: <https://www.pdaa.edu.ua/content/biblioteka>
4. Atlassian solutions. URL: <https://www.atlassian.com/>
5. Bitbucket - The code collaboration platform for modern software teams. URL: <https://bitbucket.org/>
6. Docker. URL: <https://www.docker.com/>
7. drawSQL - Design, visualize and collaborate on entity relationship diagrams for your databases. URL: <https://drawsql.app/>
8. git – fast version control. URL: <https://git-scm.com/>
9. GitHub - Where the world builds software. URL: <https://github.com/>
10. GitLab – the DevOps Platform. URL: <https://about.gitlab.com/>
11. Kubernetes. URL: <https://kubernetes.io/>
12. Trello. URL: [ps://trello.com/home](https://trello.com/home)
13. Visual Studio Code. URL: <https://code.visualstudio.com/>