**Міністерство освіти і науки України**

**Національний університет «Запорізька Політехніка»**

Кафедра програмних засобів

**Самостійна робота № 1**

з дисципліни «Soft skills, групова динаміка та комунікації» на тему:

«Створення презентацій»

Виконал(а):

студент групи КНТ-113сп І. А. Щедровський

Прийнял(а)

ст.викладач: Валерій ЛЬОВКІН

2024

**ЗМІСТ**

[1 ТЕМА ТА МЕТА РОБОТИ 3](#_Toc168558633)

[3 КОРОТКІ ТЕОРЕТИЧНІ ВІДОМОСТІ 5](#_Toc168558634)

[4 ТЕКСТ ПІДГОТОВЛЕНОЇ ДОПОВІДІ 7](#_Toc168558635)

[4.1 Що таке API 7](#_Toc168558636)

[4.2 Типи API 7](#_Toc168558637)

[4.3 Основні принципи роботи API 8](#_Toc168558638)

[4.4 Інструменти для роботи з API 10](#_Toc168558639)

[4.5 Проблеми API 11](#_Toc168558640)

[4.6 Підсумки 11](#_Toc168558641)

[5 ПРЕЗЕНТАЦІЯ 13](#_Toc168558642)

[6 ВИСНОВКИ 22](#_Toc168558643)

# ТЕМА ТА МЕТА РОБОТИ

Темою моєї роботи є «API: використання  
та аналіз існуючих рішень»

Метою цієї роботи є:

* вивчити основні можливості сучасних програм для підготовки презентацій;
* навчитися створювати презентації за допомогою засобів програм для підготовки презентацій.

1. **ЗАВДАННЯ ДО РОБОТИ**

2.1 Ознайомитися з основними теоретичними відомостями за  темою роботи, використовуючи дані методичні вказівки, лекційні матеріали та рекомендовану літературу.

2.2 Вивчити можливості однієї з сучасних програм підготовки  презентацій, зокрема дослідити пункти головного меню та навчитися наступному:

* створювати та зберігати нові презентації;
* використовувати шаблони слайдів та самостійно змінювати  оформлення;
* працювати з місцезаповнювачами;
* встановлювати ефекти анімації елементів слайду;
* встановлювати ефекти переходу при зміні слайдів;
* створювати елементи навігації за презентацією;
* вставляти в презентацію таблиці, графіки, малюнки, кліпи та  звукові файли, створені за допомогою інших програм.

2.3 Підготувати доповідь за темою, що відповідає індивідуальному завданню, обраному з таблиці А.1 за номером варіанту та узгодженому з викладачем.

2.4 Створити презентацію за підготовленою доповіддю, що  складається не менше ніж з 10 слайдів.

При створенні презентації обов’язково використовувати такі засоби програм підготовки презентацій:

* зміна кольорової схеми презентації;
* ефекти анімації;
* ефекти переходу;
* розгалуження;
* нотатки.

2.5 Оформити звіт з роботи.

* 1. Відповісти на контрольні запитання.

# КОРОТКІ ТЕОРЕТИЧНІ ВІДОМОСТІ

Формально підготовка презентацій дуже часто є частиною презентації результатів, ідей.

Відповідно програми, які направлені на створення презентацій,  надають інструменти, які придатні для розв’язання множини проблем.  У результаті таким чином можна представити основні результати звітів, проведеного аналізу даних, запропонованих ідей. Створення презентацій необхідно для візуалізації інформації, а візуалізація дозволяє  полегшити сприйняття даних, представити їх у наочному і більш про стому для сприйняття вигляді.

Сучасні офісні пакети включають і програмне забезпечення для  створення презентацій. Наприклад: PowerPoint – у Microsoft Office,  Keynote – у iWork, Impress – у LibreOffice.

Слайд (презентації) – це одна сторінка презентації, яка містить  візуальні елементи різного типу, а також текстові представлення, гіперпосилання тощо.

Нотатки слайдів містять текстові представлення основних позначок, які використовуються для допомоги доповідачу під час доповіді. Вони можуть містити основні структурні елементи доповіді щодо  конкретного слайду.

Фоном слайду може бути не тільки суцільна заливка, яка визначена білим кольором або іншим кольором заданим шаблоном, але і  така, яка відповідає потребам візуального відображення інших елементів на слайді та слайдах презентації загалом. Для цього може бути  задано інший потрібний колір фону, обрано візерунок або текстуру, встановлено рисунок через вибір відповідного файлу.

Описані вище блоки визначають розташування не тільки тексту,  але і будь-якого іншого об’єкта, поміщеного на слайд. Такі блоки називаються місцезаповнювачами. Вони визначають позицію і розмір  об’єкта. До об’єктів можуть належати заголовок, перелік (представляють текст), таблиця, діаграма, рисунок, відео тощо. При визначенні  макета слайда встановлюються позиції місцезаповнювачів. Відповідно  для кожного слайда можна обрати потрібний макет.

Будь-яку презентацію можна перетворити з кольорової до чор но-білої або презентації у відтінках сірого та навпаки. Такий варіант  може знадобитися тоді, коли презентацію потрібно роздрукувати. Але  перед створенням роздаткових матеріалів потрібно спочатку перевіри ти, як сама презентація буде виглядати у чорно-білому представленні. Для виконання цих дій необхідно використати спеціально призначені  кнопки з відповідними назвами з вкладки «Вид» у PowerPoint 2010 і  вище.

PowerPoint 2010 і вище містить широкі можливості роботи з зо браженнями та відеозаписами стосовно джерел таких медіаданих.  Джерелом медіаданих можуть бути відеозаписи з соціальних вебсай тів, де ці медіафайли безпосередньо знаходяться (найрозповсюджені ший варіант – YouTube). Для цього потрібно через вкладку «Встав лення» перейти за ланцюжком «Медіавміст» (або «Мультимедіа») → «Відео» → «Відео із вебсайту» (або «Відео із Інтернету»), де вказати  зовнішнє посилання на відео.

# ТЕКСТ ПІДГОТОВЛЕНОЇ ДОПОВІДІ

Доброго дня! Радий Вам представити свою презентацію на тему «API: Використання та аналіз існуючих рішень»

В цій презентації ми розглянемо що таке API, які є типи, їх основні принципи роботи та програми для їх використання та тестування.

## Що таке API

Інтерфейс програмування застосунків (або ж API)— це спосіб, завдяки якому дві або більше комп'ютерних програми можуть взаємодіяти між собою.

Спрощено, API є набором чітко визначених методів для взаємодії різних компонентів програмного забезпечення.

API забезпечує розробників інструментами для швидкої розробки програмного забезпечення, дозволяючи використовувати готові функції та сервіси.

Використання API спрощує інтеграцію сервісів і систем, що дозволяє створювати масштабовані додатки.

## Типи API

Всього є три основні типи API:

* Веб API
* Бібліотечні API
* Операційні API

До Веб API відносяться:

* REST (Representational State Transfer): Найпоширеніший тип веб API, який використовує стандарти HTTP з деякими правилами
* SOAP (Simple Object Access Protocol): Протокол, який використовує XML для обміну повідомленнями і забезпечує високий рівень безпеки та надійності.
* GraphQL: Запитова мова для API, яка дозволяє клієнтам запитувати тільки ті дані, які їм потрібні, що підвищує ефективність роботи.

Бібліотечні API забезпечують доступ до функцій та процедур, що знаходяться в програмних бібліотеках

Вони дозволяють розробникам використовувати вже готовий код для виконання конкретних задач, що значно спрощує і прискорює процес розробки програмного забезпечення.

Бібліотечні API часто використовуються для роботи з базами даних, обробки графіки, мережевого програмування та інших завдань.

Приклад: API бібліотеки jQuery, яка полегшує роботу з DOM в JavaScript.

Операційні API використовуються для взаємодії з операційною системою, надаючи доступ до її функцій   
та ресурсів.

Вони дозволяють розробникам створювати програми, які можуть виконувати завдання на рівні операційної системи, такі як управління файлами, обробка даних, робота з мережевими з'єднаннями та іншими системними ресурсами.

Приклад: Windows API, яке дозволяє розробникам створювати програми для операційної системи Windows.

## Основні принципи роботи API

Почнемо з того, що API повинні бути стандартизовані, абстрактні та модульні

Розглянемо ці принципи на прикладі з найбільш поширеним видом API – Веб API

Під стандартизованістю мається на увазі, що всі API повинні працювати схожим чином. Тобто мати запити та відповіді, використовувати методи та статуси відповідей HTTP

Під абстрактністю мається на увазі, що API не показують, як вони працюють всередині. Все що вони мають – входи та виходи. Можна привести аналогію з газовою плитою. У неї є вхід для газу та регулятори. Але ми не знаємо як вона працює всередині

Під модульністю мається на увазі те, що API можна розбити на маленькі незалежні API і робити до них запити окремо. Це може бути дуже корисно в деяких ситуаціях

Як я вже сказав основою Веб API є:

* запити та відповіді;
* методи HTTP;
* статуси відповідей HTTP;

Розглянемо запити та відповіді

* Коли клієнт (наприклад, браузер або мобільний додаток) хоче отримати або надіслати дані до сервера, він відправляє запит до API.
* Запит містить необхідну інформацію, таку як URL, метод HTTP, заголовки та, за потреби, тіло запиту.
* Сервер обробляє запит і відправляє відповідь, яка містить статусний код HTTP, заголовки та тіло відповіді з даними або повідомленням про помилку.

HTTP має наступні методи:

* GET - використовується для отримання даних з сервера. Наприклад, отримання списку користувачів або детальної інформації про конкретний об'єкт.
* POST - використовується для створення нових ресурсів на сервері. Наприклад, створення нового користувача або додавання нового запису.
* PUT - в икористовується для оновлення існуючих ресурсів на сервері. Наприклад, оновлення інформації про користувача.
* DELETE - в икористовується для видалення ресурсів з сервера. Наприклад, видалення користувача або запису.

А також HTTP має наступні статус коди, які показані на рисунку 4.1.

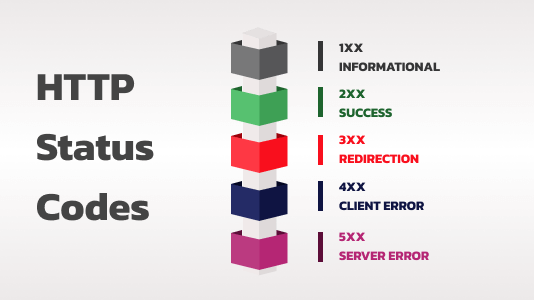


Рисунок 4.1 – HTTP Статус коди

## Інструменти для роботи з API

Тепер, коли ми знаємо як API працює ми можемо розглянути інструменти для зручної роботи з ним, а саме Postman та Swagger

Postman - потужний інструмент для тестування API, який дозволяє створювати, надсилати запити, переглядати відповіді, та автоматизувати тести.

Зручний для розробників завдяки інтуїтивно зрозумілому інтерфейсу і широким можливостям для налаштування запитів.

Зазвичай цей інструмент використовують саме для тестування API, перевірки різних запитів та роботи всього іншого

Swagger - інструмент для створення, генерування та візуалізації документації для REST API.

Дозволяє автоматично генерувати документацію з кодової бази або YAML/JSON файлів, що описують API.

Swagger UI дозволяє розробникам та користувачам взаємодіяти з API безпосередньо через веб-інтерфейс.

Swagger дуже зручний інструмент коли нам потрібно щоб з нашою API хтось взаємодіяв, наприклад, frontend команда. Ми не хочемо описувати документацію на нашу API самі, бо це довго та нудно, а також ми можемо щось забути, а ще це складно оновлювати.

Саме для цього і існує swagger – це автоматично генерована документація, яку можна віддати іншим командам та вони будуть розуміти як правильно з цим працювати

Якщо піти далі, то, наприклад, frontend може через swagger генерувати типи на основі яких він буде працювати, це приводить до зниження помилок та кількості ручних оновлень типів

## Проблеми API

Однією з проблем API є сумісність. Коли ми щось оновлюємо, то деякі клієнти які використовують нашу API можуть зламатись. Для цього ми можемо використовувати версіонування

В версіонуванні також можуть проблеми, наприклад, ми можемо забути зробити нову версію і нам потрібно одночасно підтримувати декілька версій, що дуже не зручно

Також нам потрібно підтримувати данні в різних форматах, від JSON до FormData та WebGL схем

Нам також потрібно подумати про захист від надмірних запитів, обробляти перевищення лімітів використання та сповіщувати про це клієнтів

І не потрібно забувати про безпеку! У нас повинна бути авторизація або аутентифікація, шифрування паролів та приватної інформації, а також захищена система від атак всілякого роду

## Підсумки

API дуже корисні для розробників, оскільки вони надають потужні інструменти для швидкої і ефективної розробки програмного забезпечення.

Використання API дозволяє інтегрувати різноманітні сервіси та системи, спрощуючи створення масштабованих і гнучких додатків.

Завдяки чітким стандартам, API забезпечують сумісність і узгодженість між різними компонентами.

Інтерфейси API роблять складні процеси доступними через прості виклики, що значно прискорює розробку і впровадження нових функцій.

# ПРЕЗЕНТАЦІЯ

Всі слайди презентації представлені на рисунках 5.1 – 5.18.

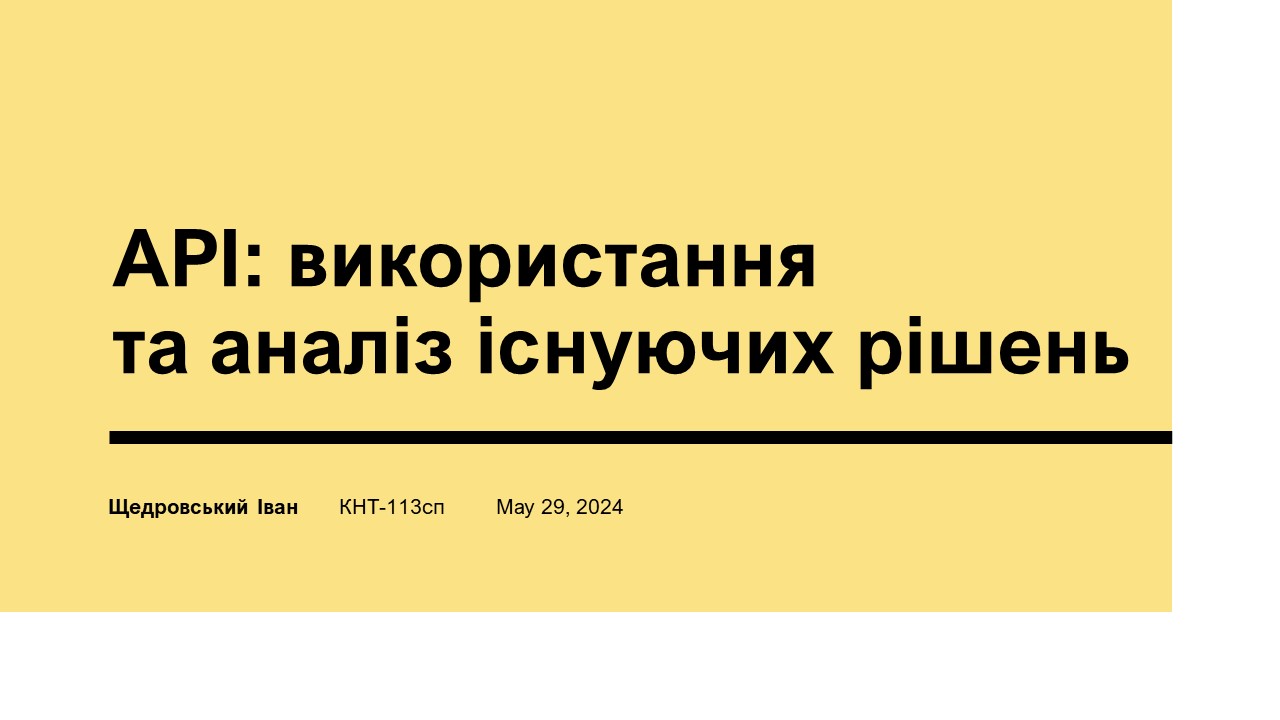


Рисунок 5.1 – Слайд презентації 1

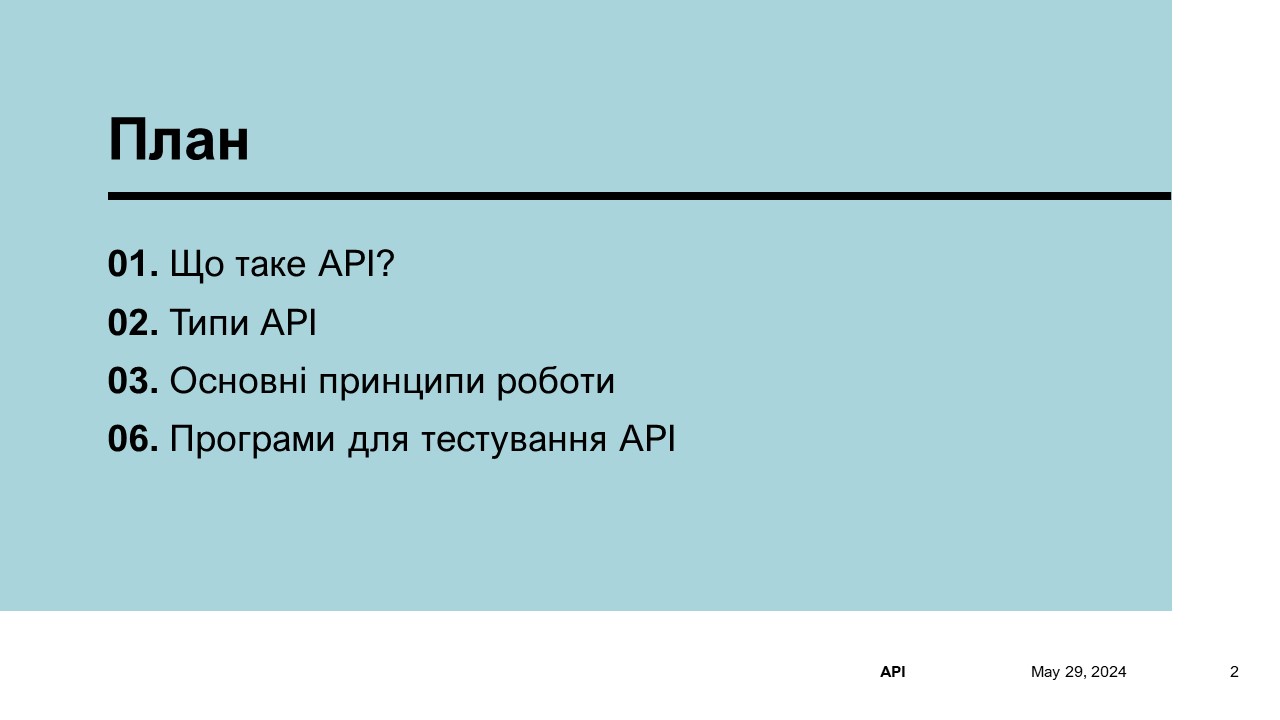


Рисунок 5.2 – Слайд презентації 2

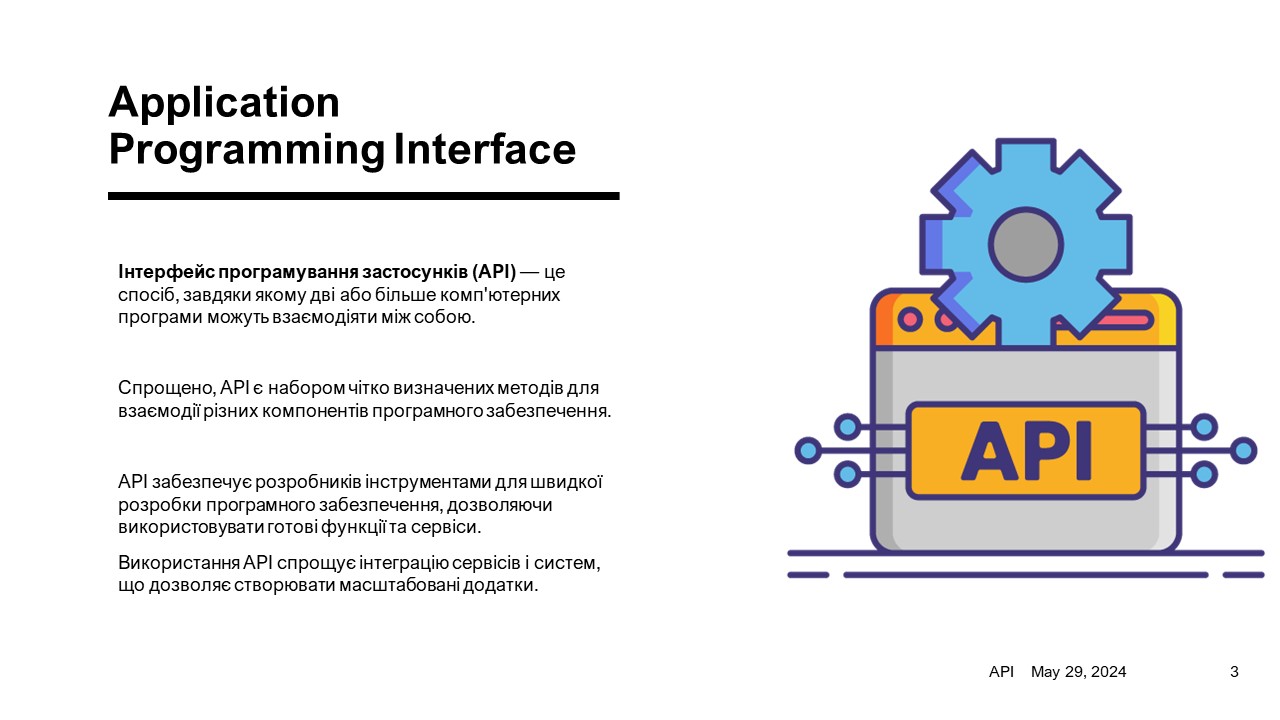
****

Рисунок 5.3 – Слайд презентації 3

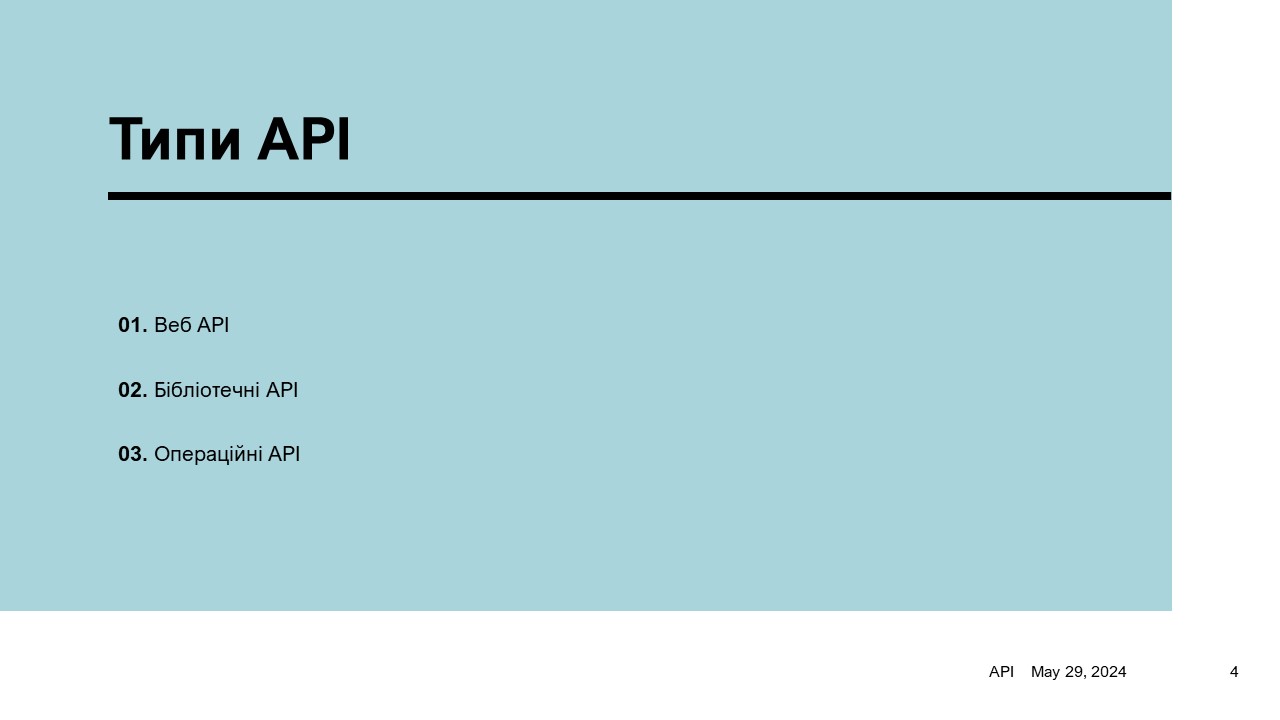
****

Рисунок 5.4 – Слайд презентації 4

****

Рисунок 5.5 – Слайд презентації 5

****

Рисунок 5.6 – Слайд презентації 6

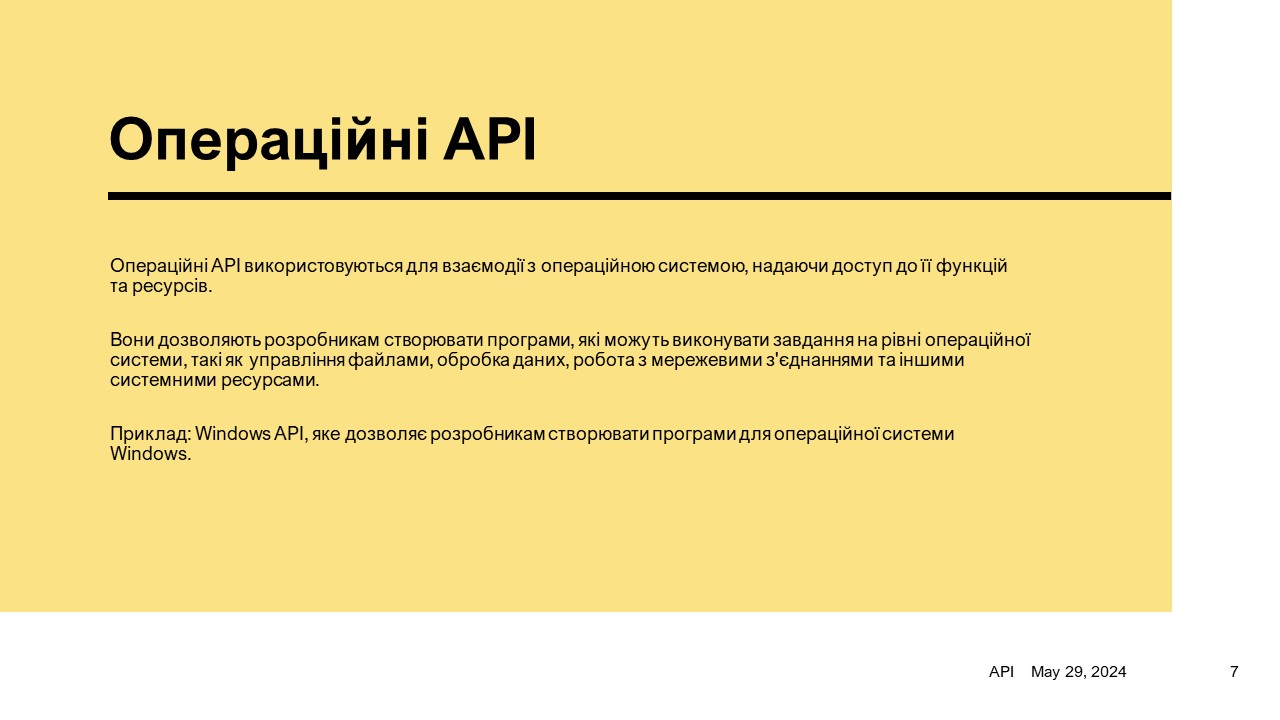
****

Рисунок 5.7 – Слайд презентації 7

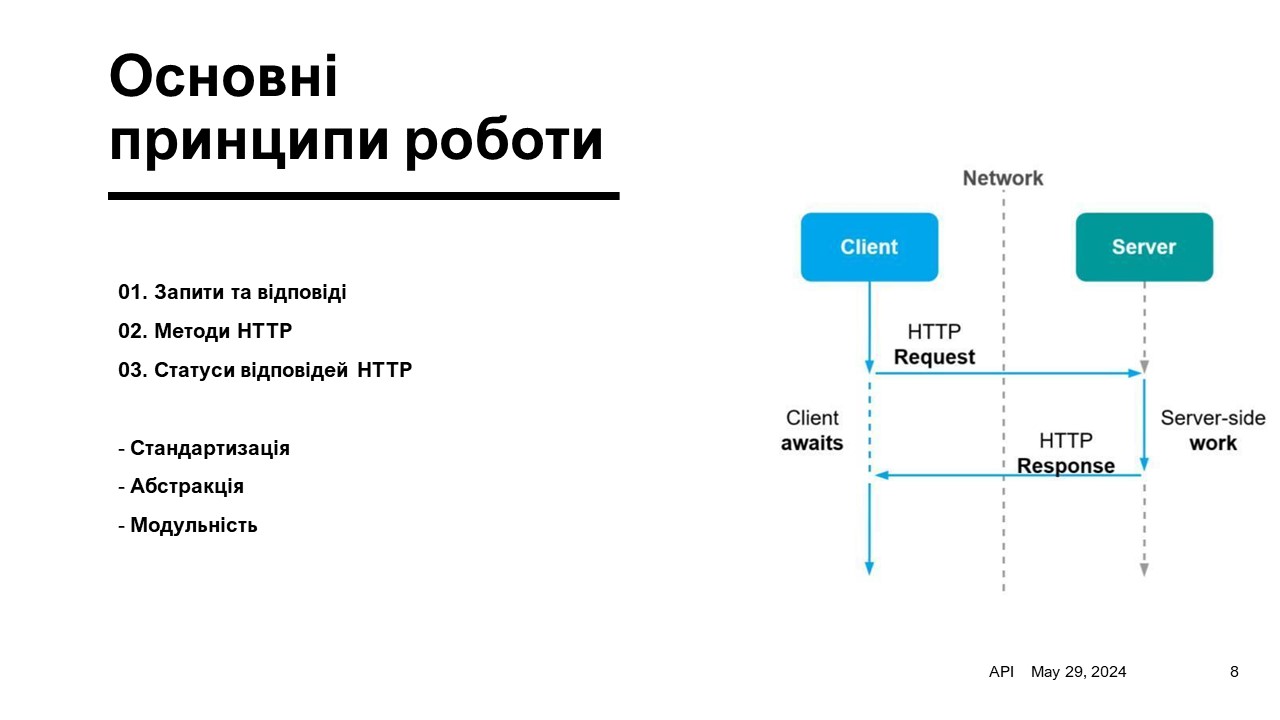
****

Рисунок 5.8 – Слайд презентації 8

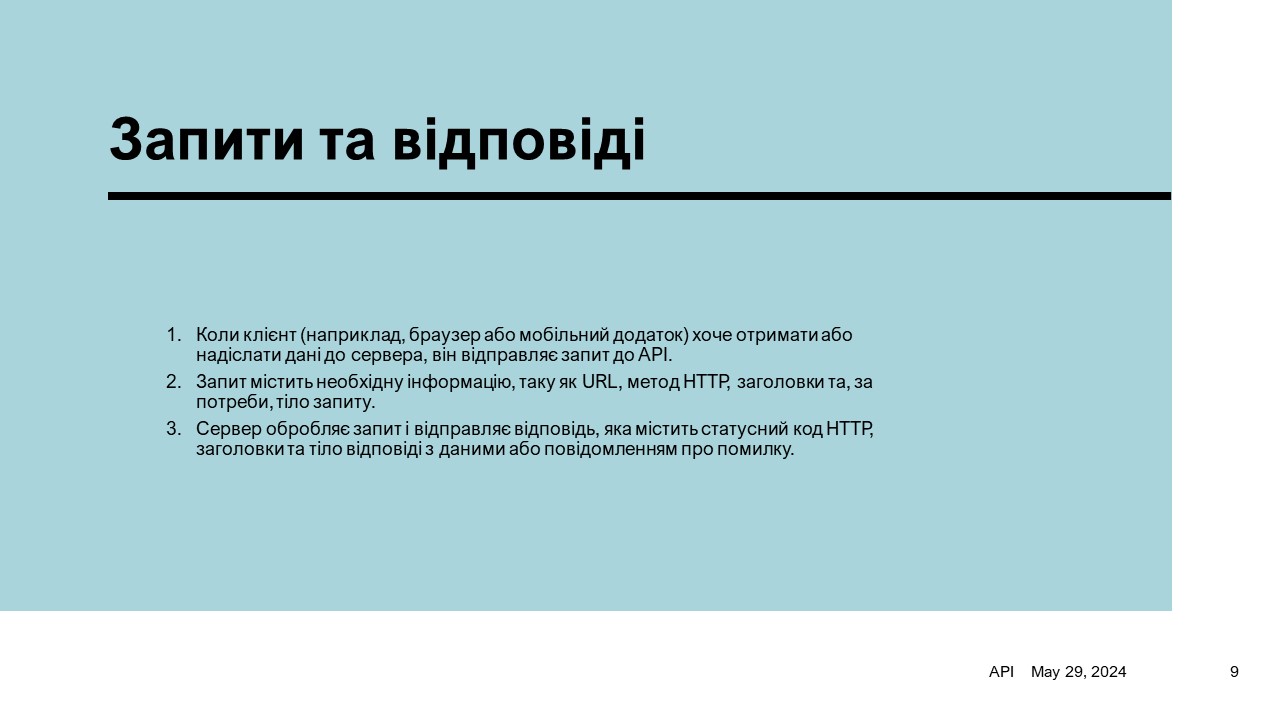
****

Рисунок 5.9 – Слайд презентації 9

****

Рисунок 5.10 – Слайд презентації 10

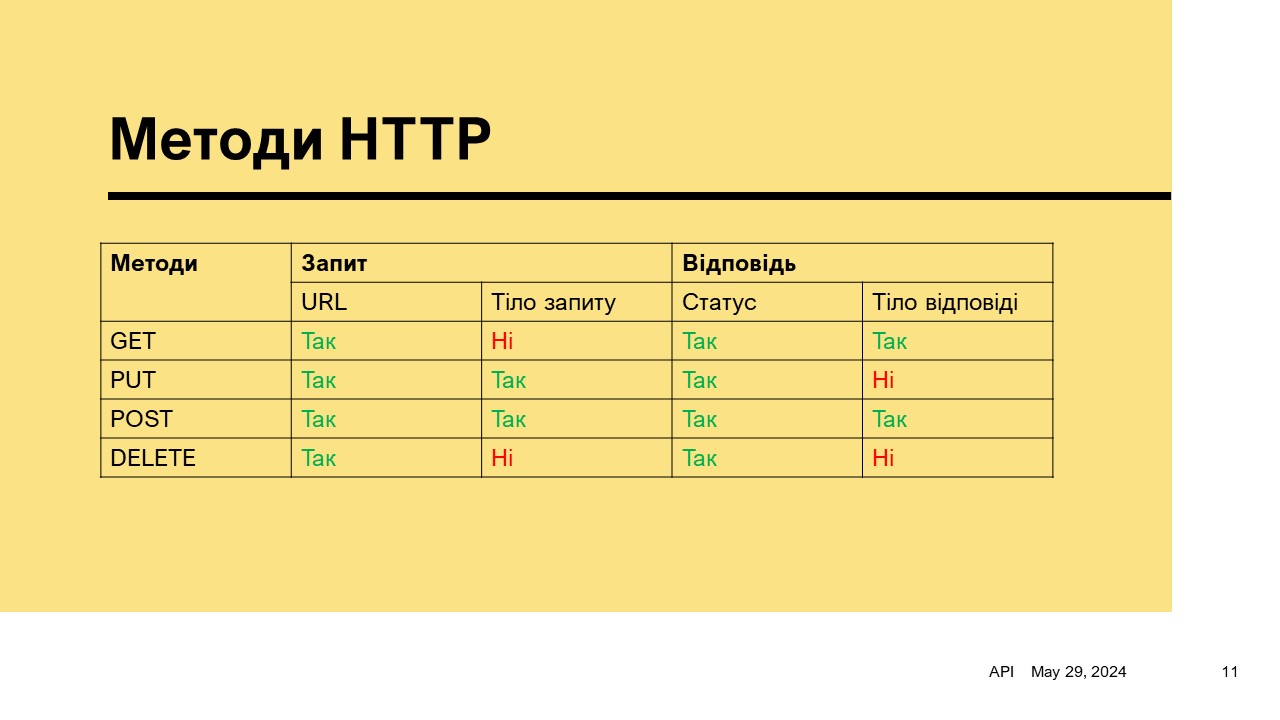
****

Рисунок 5.11 – Слайд презентації 11

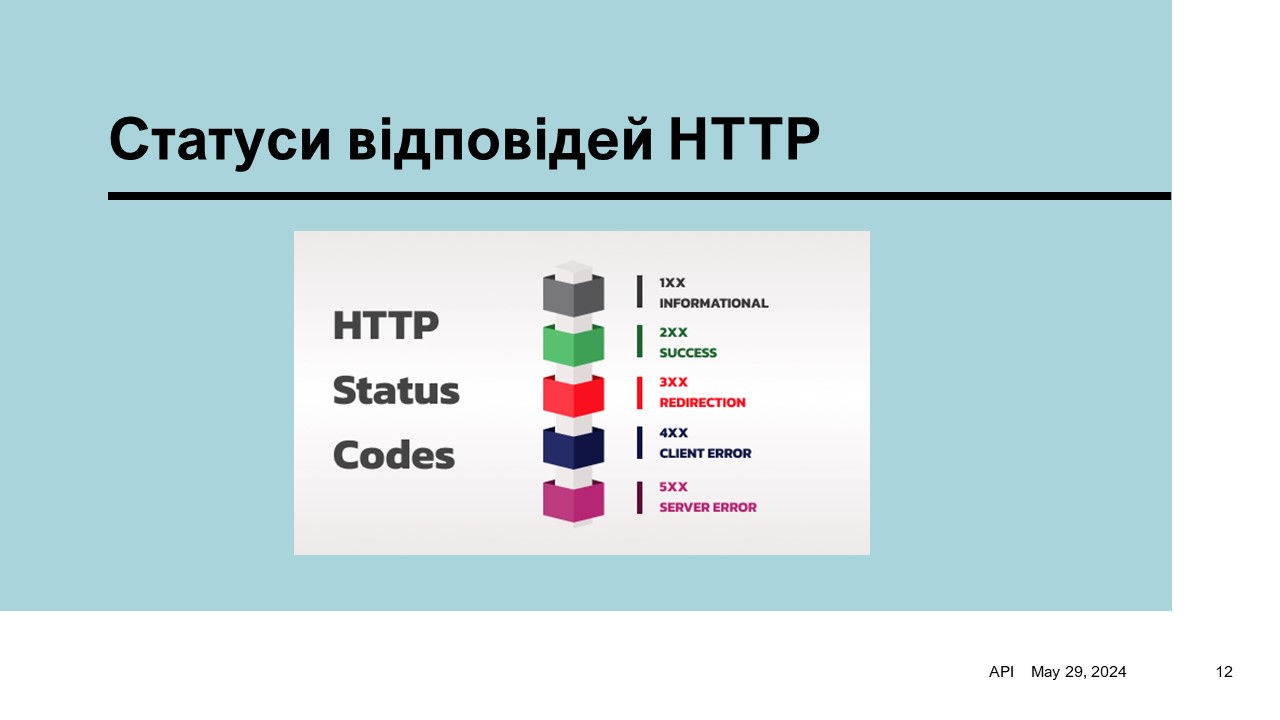
****

Рисунок 5.12 – Слайд презентації 12

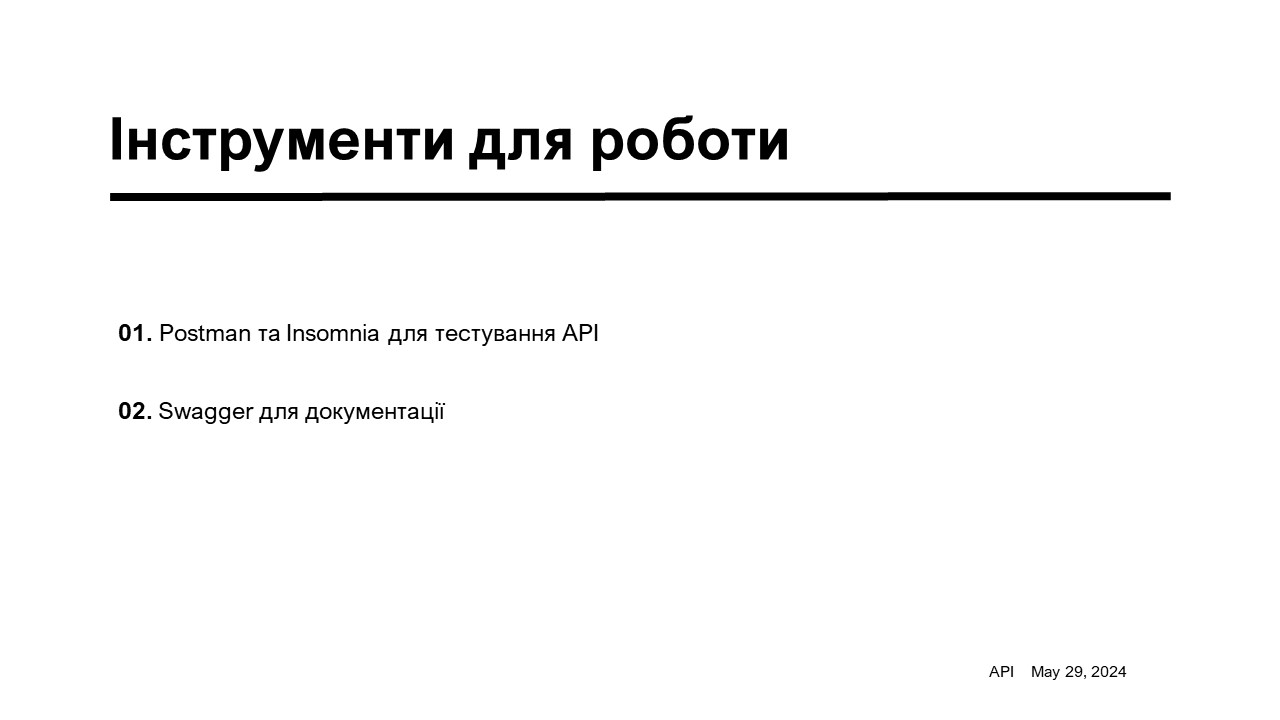
****

Рисунок 5.13 – Слайд презентації 13

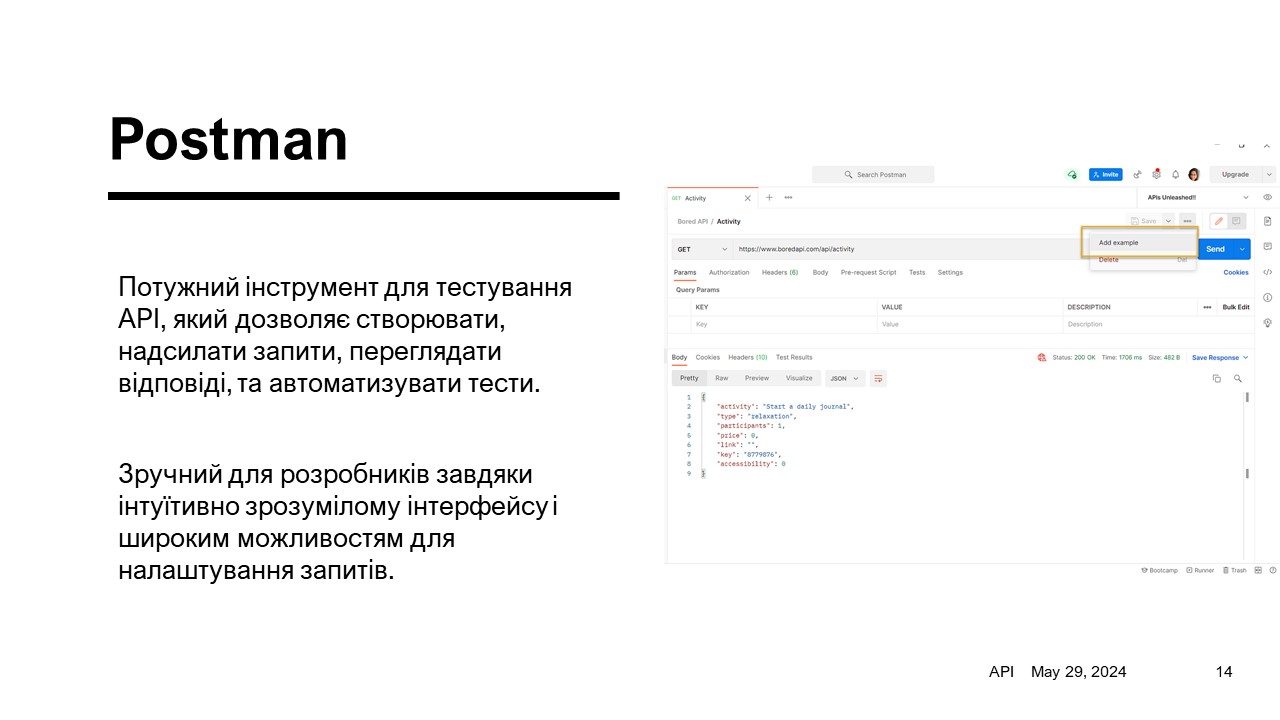
****

Рисунок 5.14 – Слайд презентації 14

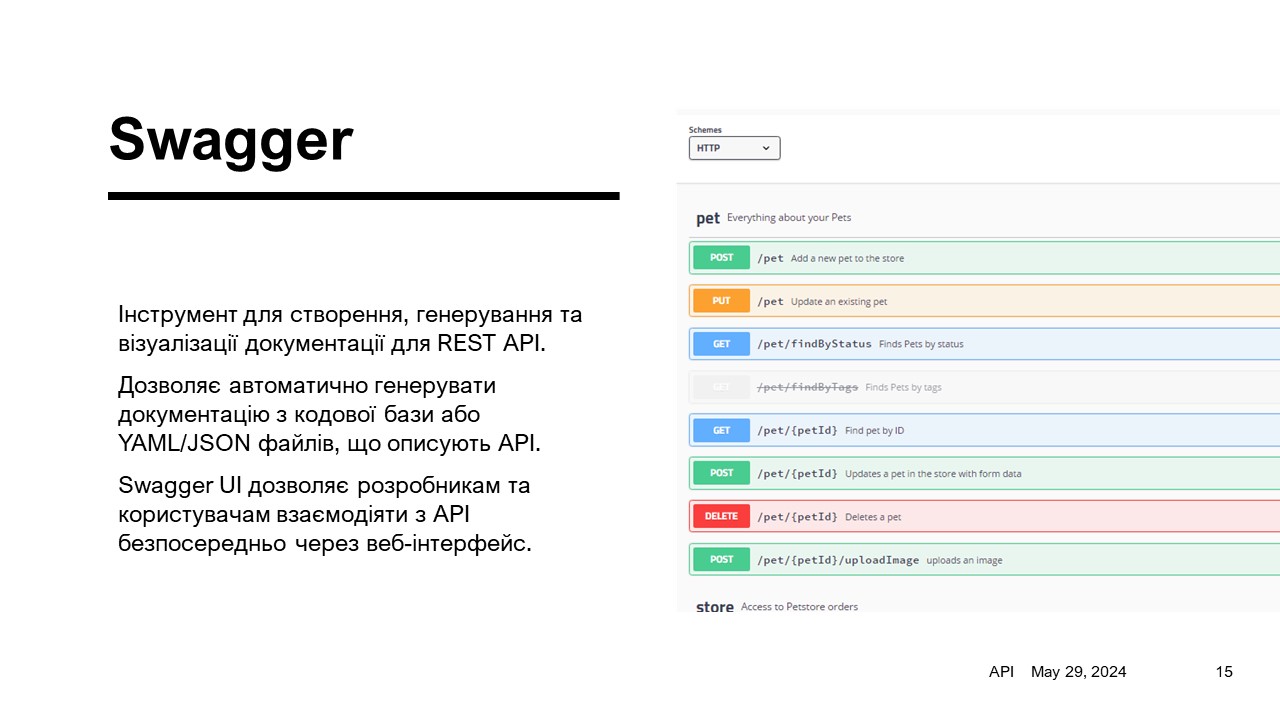
****

Рисунок 5.15 – Слайд презентації 15

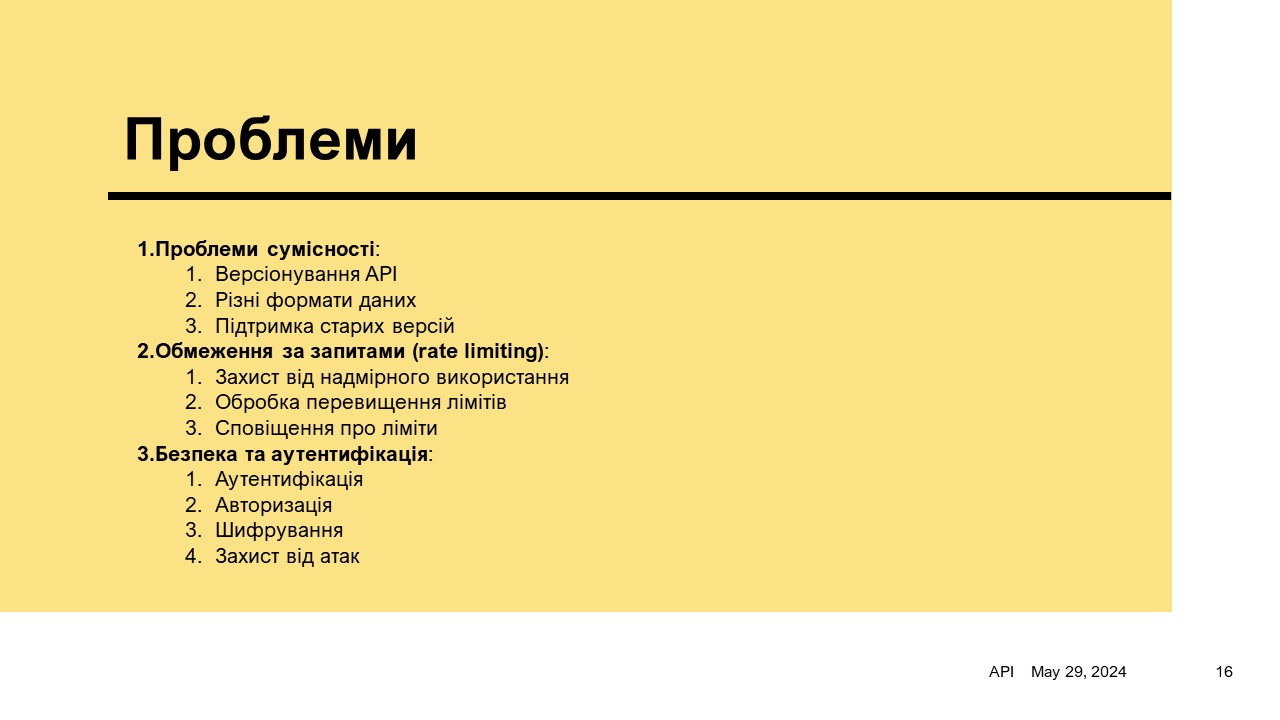
****

Рисунок 5.16 – Слайд презентації 16

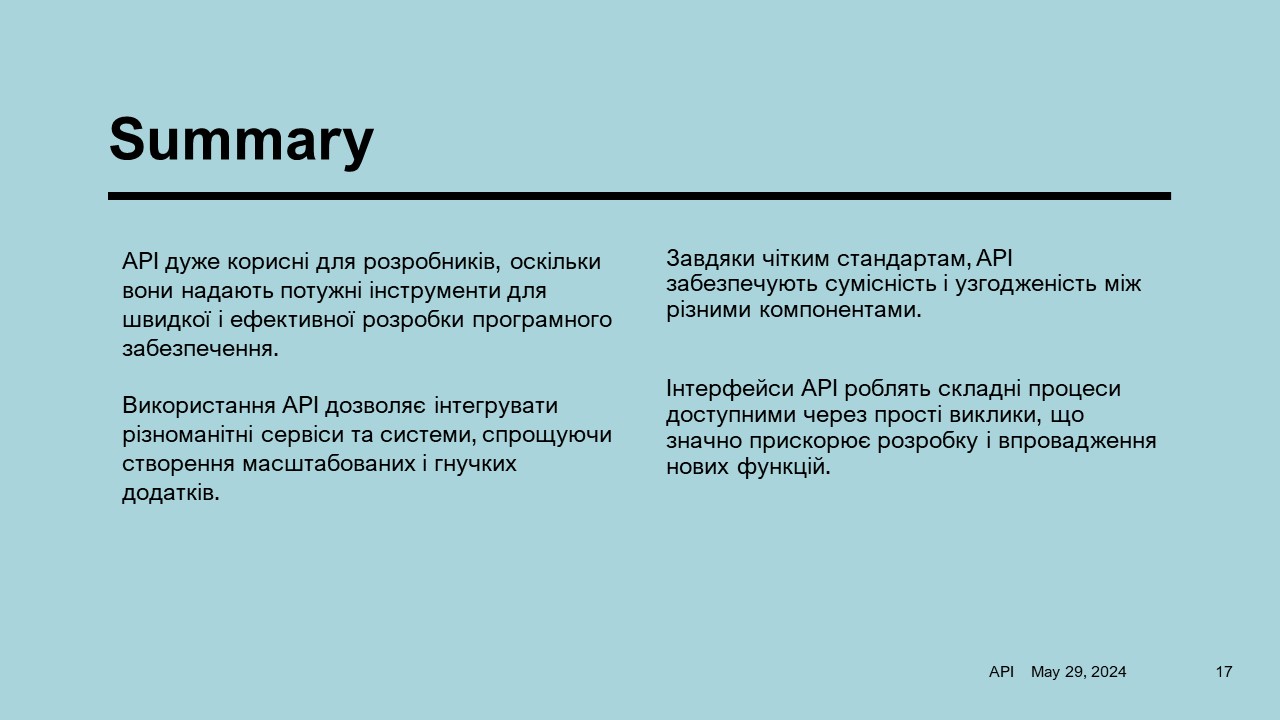
****

Рисунок 5.17 – Слайд презентації 17

****

Рисунок 5.18 – Слайд презентації 18

# ВИСНОВКИ

У рамках цієї роботи було вивчено основні можливості сучасних програм для підготовки презентацій.

Окрім теоретичного аналізу існуючих рішень, було проведено практичне навчання з використання цих засобів для створення презентацій.

Отримані знання та навички дозволяють ефективно використовувати сучасні інструменти для підготовки якісних та професійних презентацій, що є важливою складовою успішної комунікації та подання інформації.

Відповіді на запитання:

**Які додаткові можливості форматування тексту присутні в програмах підготовки презентацій?**

У програмах підготовки презентацій, таких як Microsoft PowerPoint, Google Slides, та інших, присутні наступні додаткові можливості форматування тексту:

* **Шрифти та розмір тексту**: вибір різних шрифтів, зміна розміру тексту.
* **Стиль шрифту**: жирний, курсив, підкреслений текст.
* **Колір тексту**: зміна кольору тексту, додавання градієнтів.
* **Виділення тексту**: маркування кольором виділення.
* **Вирівнювання**: вирівнювання по лівому, правому краю, центру, чи за шириною.
* **Інтервали**: зміна інтервалу між рядками та абзацами.
* **Нумеровані та марковані списки**: створення списків з номерами або маркерами.
* **Тінь та об’єм**: додавання тіні, об’ємних ефектів до тексту.
* **Стислий текст**: зменшення міжсимвольного інтервалу для щільнішого тексту.
* **Анімації та переходи**: додавання анімацій до текстових елементів.

**Для чого використовується підсумковий слайд?**

Підсумковий слайд використовується для:

* **Повторення основних моментів**: резюмування ключових точок презентації.
* **Висновків**: представлення висновків та основних результатів.
* **Заклику до дії**: підкреслення наступних кроків або дій, які повинні бути виконані.
* **Подяки**: вираження подяки аудиторії за увагу та участь.
* **Контактної інформації**: надання контактної інформації для подальшої комунікації.

**Як включити в підсумковий слайд тільки окремі слайди?**

Цей слайд був відключений в версіях старших за 2007 PowerPoint

**Як переключити презентацію до чорно-білого вигляду?**

Щоб переключити презентацію до чорно-білого вигляду в Microsoft PowerPoint:

* **Перейдіть до вкладки “Вид”**: Відкрийте вкладку “Вид” на стрічці інструментів.
* **Виберіть режим “Чорно-білий”**: Натисніть на “Чорно-білий” у розділі “Кольоровий/Чорно-білий”.
* **Налаштуйте вигляд**: Ви можете налаштувати, як саме виглядатимуть елементи у чорно-білому режимі, використовуючи інструменти на панелі “Чорно-білий”.

**Які елементи можна вставити в місцезаповнювач на слайді?**

У місцезаповнювач на слайді можна вставити такі елементи:

* **Текст**: будь-який текстовий контент.
* **Зображення**: фотографії, графіки, ілюстрації.
* **Таблиці**: структурування даних у вигляді таблиць.
* **Графіки та діаграми**: для візуального представлення даних.
* **Відео та аудіо**: мультимедійні елементи.
* **Смарт-арт**: графічні схеми для ілюстрації процесів чи відносин.
* **Іконки та символи**: додавання іконок для візуального акценту.
* **3D-моделі**: інтерактивні тривимірні моделі.