

## Завдання 1 - бібліотека SimpleLightbox

Виконуй це завдання у файлах 01-gallery.html і 01-gallery.js. Розбий його на декілька підзавдань:

1. Додай бібліотеку [SimpleLightbox](#) як залежність проекту, використовуючи `npm` (посилання на CDN з твоєї минулої роботи більше не потрібне).
2. Використовуй свій JavaScript код з попередньої домашньої роботи, але виконай рефакторинг з урахуванням того, що бібліотека була встановлена через `npm` (синтаксис `import/export`).

Для того щоб підключити CSS код бібліотеки в проект, необхідно додати ще один імпорт, крім того, що описаний в документації.

```
// Описаний в документації
import SimpleLightbox from "simplelightbox";
// Додатковий імпорт стилів
import "simplelightbox/dist/simple-lightbox.min.css";
```

## Завдання 2 - відеоплеєр

HTML містить `<iframe>` з відео для Vimeo плеєра. Напиши скрипт, який буде зберігати поточний час відтворення відео у локальне сховище і, після перезавантаження сторінки, продовжувати відтворювати відео з цього часу.

```
<iframe
  id="vimeo-player"
  src="https://player.vimeo.com/video/236203659"
  width="640"
  height="360"
  frameborder="0"
  allowfullscreen
  allow="autoplay; encrypted-media"
></iframe>
```

Виконуй це завдання у файлах 02-video.html і 02-video.js. Розбий його на декілька підзавдань:

1. Ознайомся з [документацією](#) бібліотеки Vimeo плеєра.
2. Додай бібліотеку як залежність проекту через `npm`.
3. Ініціалізуй плеєр у файлі скрипта як це описано в секції [pre-existing player](#), але враховуй, що у тебе плеєр доданий як `npm` пакет, а не через CDN.
4. Вивчи документацію методу [on\(\)](#) і почни відстежувати подію [timeupdate](#) - оновлення часу відтворення.
5. Зберігай час відтворення у локальне сховище. Нехай ключем для сховища буде рядок `"videoplayer-current-time"`.

6. Під час перезавантаження сторінки скористайся методом [setCurrentTime\(\)](#) з метою відновлення відтворення зі збереженої позиції.
7. Додай до проекту бібліотеку [lodash.throttle](#) і зроби так, щоб час відтворення оновлювався у сховищі не частіше, ніж раз на секунду.

## Завдання 3 - форма зворотного зв'язку

HTML містить розмітку форми. Напиши скрипт, який буде зберігати значення полів у локальне сховище, коли користувач щось друкує.

```
<form class="feedback-form" autocomplete="off">
  <label>
    Email
    <input type="email" name="email" autofocus />
  </label>
  <label>
    Message
    <textarea name="message" rows="8"></textarea>
  </label>
  <button type="submit">Submit</button>
</form>
```

Виконуй це завдання у файлах 03-feedback.html і 03-feedback.js. Розбий його на декілька підзавдань:

1. Відстежуй на формі подію `input`, і щоразу записуй у локальне сховище об'єкт з полями `email` і `message`, у яких зберігай поточні значення полів форми. Нехай ключем для сховища буде рядок `"feedback-form-state"`.
2. Під час завантаження сторінки перевіряй стан сховища, і якщо там є збережені дані, заповнюй ними поля форми. В іншому випадку поля повинні бути порожніми.
3. Під час сабміту форми очищуй сховище і поля форми, а також виводь у консоль об'єкт з полями `email`, `message` та їхніми поточними значеннями.
4. Зроби так, щоб сховище оновлювалось не частіше, ніж раз на 500 мілісекунд. Для цього додай до проекту і використовуй бібліотеку [lodash.throttle](#).