

## Інструкція для виконання лабораторної роботи №3

1. **Задаємо структуру папок** (створюємо ті, яких ще не було у попередній роботі)
  - `app/` - список файлів, ресурсів, картинок
    - i. `html/` – файли з розширенням `html`
      - `task_1_html.html`
      - `task_2_css.html`
    - ii. `css/` – файли з розширенням `css`
      - `task_2_css.css`
    - iii. `img/` – файли зображень
    - iv. `less/` – файли з розширенням `less` (код `css`-препроцесора `less`)
    - v. `sass/` – файли з розширенням `html` (код `css`-препроцесора `sass`)
    - vi. `scss/` – файли з розширенням `html` (код `css`-препроцесора `scss`)
    - vii. `index.html` – головний `html` файл
  - `build/` - готовий результат
  - `gulpfile.js` – список задач для `gulp 4` (файл)
2. **Написання `html`-документів та задання `css`-стилів** для `web`-сторінки.  
У файлах `index.html`, `task_2_css.html`, `task_2_css.css` прописати усе, як показано у зразку ([https://github.com/Dniproverts/Web\\_programming](https://github.com/Dniproverts/Web_programming)).
3. **Написання задач для `gulp 4`:**  
написати у файлі `gulpfile.js` усі задачі, як показано у зразку ([https://github.com/Dniproverts/Web\\_programming](https://github.com/Dniproverts/Web_programming)).
4. У консолі `git` прописуємо:  
`gulp build` (далі `enter`)      `// для збірки проекту`  
`gulp` (далі `enter`)      `// для тестування`  
`gulp deploy` (далі `enter`)      `// для відправки результату на github`
5. Відслідковуємо зміни -> `git status`
6. Додаємо усі відслідковані зміни до нової версії (коміту) -> `git add .`
7. Створюємо нову версію (коміт) -> `git commit -m "Коментар до задачі"`
8. Зливаємо зміни на віддалений сервер -> `git push`

Корисні посилання:

- 1) [https://github.com/Dniproverts/Web\\_programming](https://github.com/Dniproverts/Web_programming)
- 2) <https://www.youtube.com/watch?v=hft4XYApT44&t=251s>
- 3) <https://www.w3schools.com/cssref/>
- 4) <https://fonts.google.com/>