

## Інструкція для виконання лабораторної роботи №5, 6

### 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
    - task\_3\_js.html
    - task\_4\_site.html
  - ii. css/ – файли з розширенням css
    - task\_2\_css.css
    - task\_3\_js.css
    - task\_4\_site.css
    - scrollbar.css
  - iii. js/ – файли з розширенням js
    - task\_3\_js.js
    - snow.js
    - jquery-3.6.0.js
    - task\_4\_js.js
  - iv. data/ – папки та файли з даними для вашого сайту
  - v. img/ – файли зображень
  - vi. less/ – файли з розширенням less (код css-препроцесора less)
  - vii. sass/ – файли з розширенням html (код css-препроцесора sass)
  - viii. scss/ – файли з розширенням html (код css-препроцесора scss)
  - ix. 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\_4\_site.html, task\_4\_site.css прописати так як показано у зразку **АЛЕ ДЛЯ СВОГО САЙТУ** з підключенням необхідних модулів, скриптів і т.д.

([https://github.com/Dniproverts/Web\\_programming](https://github.com/Dniproverts/Web_programming)).

### 3. **Написання js-документів** для web-сторінки.

У файлах task\_4\_site.js прописати так, як показано у зразку **АЛЕ ДЛЯ СВОГО САЙТУ** з підключенням необхідних модулів, скриптів і т.д.

([https://github.com/Dniproverts/Web\\_programming](https://github.com/Dniproverts/Web_programming)).

4. У консолі git встановити усі необхідні модулі.
5. 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)).
6. 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
7. Відслідковуємо зміни -> `git status`
8. Додаємо усі відслідковані зміни до нової версії (коміту) -> `git add .`
9. Створюємо нову версію (коміт) -> `git commit -m "Коментар до задачі"`
10. 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=TZSY6rDUDrE&t=312s>
- 3) <https://mdbootstrap.com/docs/standard/navigation/headers/>