**[goit-js-hw-08](https://baraban2003.github.io/goit-js-hw-08/)**

**Parcel boilerplate**

**Скрытые файлы**

Включите отображение скрытых файлов и папок в проводнике своей операционной системы, иначе вы не сможете выбрать и скопировать себе файлы настроек проекта, имена которых начинаются с точки.

**Зависимости**

На компьютере должена быть установлена LTS-версия [Node.js](https://nodejs.org/en/) со всеми дополнительными инструментами кроме **Chocolatey** - его ставить не нужно.

**Перед началом работы**

Один раз на проект установить все зависимости.

npm ci

**Разработка**

Запустить режим разработки.

npm run dev

Во вкладке браузера перейти по адресу [http://localhost:1234](http://localhost:1234/).

**Деплой**

Сборка будет автоматически собирать и деплоить продакшен версию проекта на GitHub Pages, в ветку gh-pages, каждый раз когда обновляется ветка main. Например, после прямого пуша или принятого пул-реквеста. Для этого необходимо в файле package.json отредактировать поле homepage и скрипт build, заменив имя\_пользователя и имя\_репозитория на свои.

**"homepage"**: "https://имя\_пользователя.github.io/имя\_репозитория",

**"scripts"**: {

**"build"**: "parcel build src/\*.html --public-url /имя\_репозитория/"

},

На всякий случай стоит зайти в настройки репозитория Settings > Pages и убедиться что продакшен версии файлов раздаются из папки /root ветки gh-pages.

Через какое-то время живую страницу можно будет посмотреть по адресу указанному в отредактированном свойстве homepage, например <https://goitacademy.github.io/parcel-project-template>.

**Файлы и папки**

* Все паршалы файлов стилей должны лежать в папке src/sass и импортироваться в src/sass/main.scss
* Изображения добавляйте в папку src/images, заранее оптимизировав их. Сборщик просто копирует используемые изображения чтобы не нагружать систему оптимизацией картинок, так как на слабых компьютерах это может занять много времени.