

PR03

Summary:

The purpose of this project is to exercise some of the core concepts learned in web development. Students will have 3 options to choose from for this last assignment. They can choose to either (1) build a game (2) build an infographic (3) build a full-stack application. It's recommended to pick the last option, the full-stack application, because students will learn more about the interactions that occur between the frontend and backend of web development.

OPTION I - GAME

Requirements:

Students can choose to build any game they want to code. Examples can include tic tac toe, connect 4, simon game, tetris, battleship...etc. The game must meet the basic use cases of the game selected.

Preview:

The game does not have to follow any particular visual style but here are some examples:



OPTION II - INFOGRAPHIC

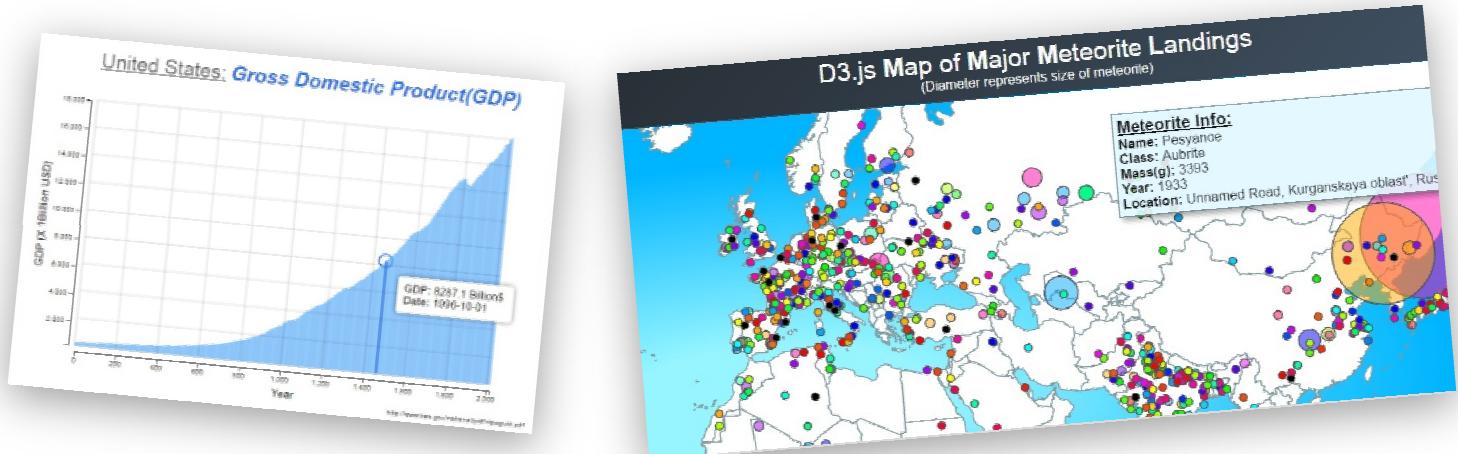
Requirements:

Students can also choose to build any infographic they want to code. Examples can include a chart to show the annual GDP, or number of people in space, or number of SpaceX flights. The infographic must use a get api (such as `$.getJSON()` or `axios.get()`) to collect data from a free data resource such as the link below, and display it on a chart. The chart can either be built using vanilla JavaScript or a popular infographic library like Chart.js or D3.js.

- <https://project-awesome.org/jdorfman/awesome-json-datasets#climate>

Preview:

The infographic does not have to follow any particular visual style but here are some examples:



OPTION III – FULL-STACK

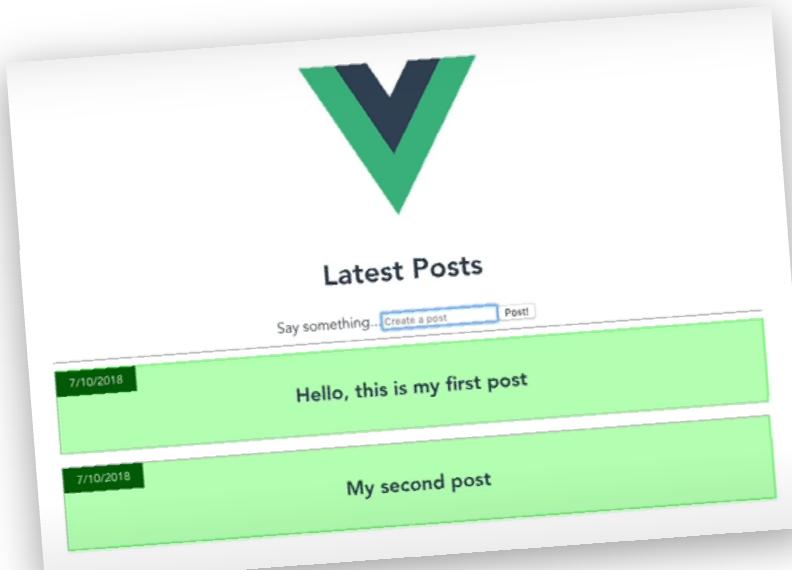
Requirements:

Students can also choose to build a full-stack application. The application could either be a utility similar to what is shown on the links below or any app like a game but must CRUD a database i.e. create, read, update, and delete data. It will have a frontend, backend, and database and hosted on the internet. The app can be completed by following the links below one line at a time, but students must change the styles of the frontend to make it look different.

- https://www.youtube.com/watch?v=j55fHUJqtyw&t=66s&ab_channel=TraversyMedia
- https://www.youtube.com/watch?v=X-JZ-QPAPuUs&ab_channel=TraversyMedia
- https://www.youtube.com/watch?v=W-b9KGwVECs&t=619s&ab_channel=TraversyMedia

Preview:

Screenshot of the tutorial from the links above:



SUBMISSIONS:

Grading:

The grading will be loosely ended, but will focus on how well the use cases are met, whether the app is bug-free, and the overall quality of the project compared to other students. It's important to test your app thoroughly. The application will be tested on chrome only and no other browsers.

Deadline:

The deadline is found under the 'end' attribute displayed on the header of the submission slab for 'pr03'. The project can be built using any fiddle or hosting service but the app must be successfully deployed on the internet if a fiddle is not used. Example hosting sites include:

- heroku
- render
- github pages
- netlify
- vercel
- firebase
- aws
- azure
- stackblitz
- codepen
- codesandbox

When the app is complete, paste the link on the submission slab of Apollo and hit submit.