Professional HTML Page

The goal of this project is to **write a big HTML page** (only one page, not a website), that has a lot of structure and styles. This will make you investigate particular ways of doing things and certain tags and CSS properties, etc. Make it as big as you like, the idea is to invest in knowing the web platform more deeply. Challenge yourself to do what you don't know.

The theme of the page is **free**, but here are some suggestions:

- **Page clone:** write a clone of a web page you like (without cheating!). Use it as a model and try to make it as close as the style you see as possible (or even make changes that you like better).

- **CV page**: create a page showing your own CV (and upload it up somewhere!). Put your efforts into making it really nice and impressive. Research about CVs from web developers first, to get some inspiration.

- **Course notes**: Create a nice document with all the course notes, with good styles and structure. Try to make code samples collapsible, link relevant parts of the document with other parts, put an index (maybe generated with Javascript?). Insert samples boxes (maybe write them in `https://codesandbox.io` and embed them into your notes).

Deliverable

There is a deliverable on the campus website, in which you should give a link to a (surprise, surprise...) **GitHub repository** as soon as possible. At the deadline you should also **upload a ZIP file** with your finished project (so that the campus website has a copy as well, to keep a record of everything).

Mode of development

The suggested way of doing this project (and almost any other one), is:

- Start with a draft, and finish it very quickly. A prototype of the whole project, possibly badly styled or with incomplete content, etc. *Do a commit of this project as your initial version*.

- In successive iterations (e.g. commits), make it better in particular ways: more content in a specific section, expand content everywhere, correct typos everywhere, style specific parts in the whole document, etc. But at each step, the project should be "coherent" as a whole, it should not have only upgrades in some parts but not others. It should feel complete.

- In all iterations make an effort to find something to do that is relatively short and concrete (adding internal links, styling certain elements in some way, adding Javascript to manipulate elements, etc.), do it, and commit to the repository explaining what you did.

By working this way, every version of the project stands on its own, but you keep improving it on every session. This is a benefit of version control that transcends the technology and helps project management.