GIT

- version control, collaborating and deployment
- industry standard

- keeps track of changes to code → history of project
- synchronize code between different people → store code on remote server (central version)
- test changes to code without losing original
- revert back to older versions of code

GIT COMMANDS

- `git clone <url>` downloads repository from remote server
- `git add <files>` include files next time we commit → track file
- `git commit -m "msg"` makes snapshot and saves repository
- `git status` see current status of repository
- `git push` upload changes to remote server
- `git pull` download latest changes from remote server

MERGE CONFICTS

```
<<<< HEAD
      ⋮        } local code/changes
======
      ⋮        } remote code/changes
>>>>
```

→ remove version you don't want

↳ commit changes

- git log — shows history of commits (commit hash)
- git reset --hard <commit> — reset code to old commit
  git reset --hard origin/master — reset code to version from origin (remote sever that was cloned)
- git reflog — shows history of all references

HTML

```
<!DOCTYPE html>
```
→ tell webbrowser that file is written in HTML 5

```
<html>
    <head>
        <title> Title </title>
    </head>
    <body>
        Hello World!
    </body>
</html>
```

metadata of page (not actually displayed)

→ end of page content

HTML TAGS

<h1> - <h6> headings from biggest to smallest

```
<ul>
    <li> First element </li>
    <li> Second element </li>
</ul>
```

unordered list with two list items

```
<ol>
    <li> First element </li>
    <li> Second element </li>
</ol>
```

ordered (= numbered list) with two list items

```
<img src="img.jpg">
```

↑ HTML attribute

insert image file, does not require an end tag (= self-contained tag)

```
<img src="img.jpg" height="200" width="200">
```

↑ defines size of image, optional attributes

→ specifying only one attribute preserves ratio

→ make responsive by using percentage (%)

```
<table>
  <tr>
    <th> Header </th>
  </tr>
  <tr>
    <td> Table Data </td>
  </tr>
</table>
```

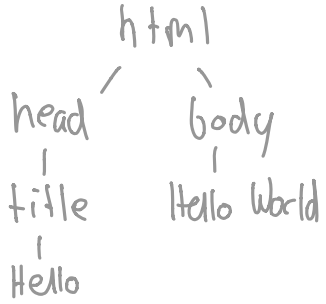
creates table with two rows and one column, including a header and table data

```
<form>
  <input type="text" placeholder="Placeholder" name="name">
  <button> Submit! </button>
</form>
```

↳ used to access value

↳ form used for user input, includes some type of input and button

- HTML files are like tree structures (Document Object Model)



CSS

- tags can include style attribute with CSS properties
  style="color: blue; text-align: center;"

- HEX codes are representation of RGB values in base 16
  ↳ 16 x 16 = 256 → two digits per color

<style> CSS </style> include styling inside HTML document
↳ usually placed with <head> tag

<link rel="stylesheet" href="style.css"> links and renders file of type stylesheet

CSS PROPERTIES

```
tag {
  property1: value1;
  property2: value2;
}
```

basic structure of CSS

- color — text color of element
- background-color — background color of element
- width — width of element
- height — height of element
- margin — spacing around element
- padding — internal spacing of element

- font-family — include multiple font-families in case one is unsupported
- font-size — how big font should show up
- font-weight — "thickness" of font
- border — defines thickness, type and color of border
- border-collapse — used to remove double borders (in tables)

tag1, tag2 {...} — apply same style to several tags

needs to be unique in whole HTML document

```
<div id="top>
    This is the <span class="name"> top </span> of my web page.
</div>
```

block element

inline element

groups several elements in class

#top {...} — ID selector

.name {...} — class selector

DEPLOY WEBPAGE TO INTERNET

- simple way using GithubPages
  ↳ Settings < GithubPages < Select Source (master branch of repository)
- pushed changes are automatically deployed to GithubPages