

OIM3690 - Web Technologies

2022 Fall

Session 26 (12/01)



Today's Agenda

1. Welcome/News/Announcements
2. Responsive Design and Bootstrap
3. **Introducing React** by Gibson Njine
4. Course Review
5. Session 26
 - i. Creating a web app using **Python**
 - ii. Revisit **How web works**

Welcome/News/Announcements

1. Project **Final**: due **Tuesday 12/06**

- I am available to meet to discuss about your project this week and next week.

2. Class Schedule:

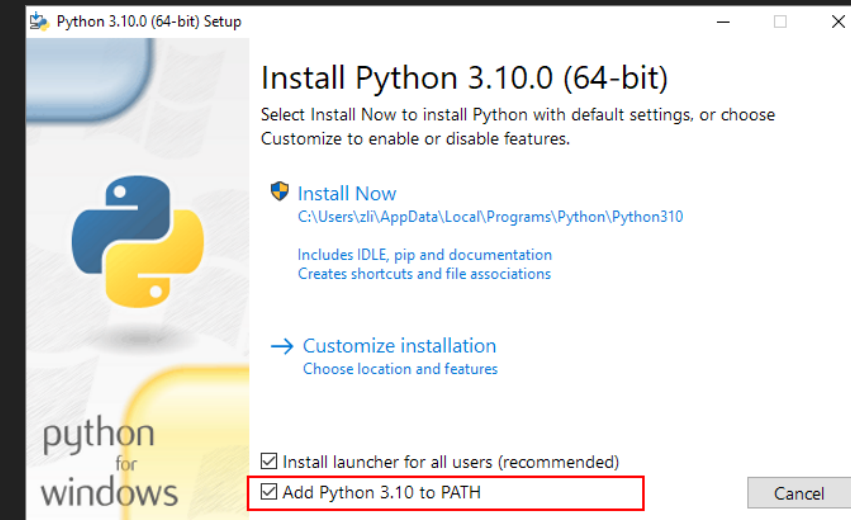
- Today: **Class Overview**/Creating Web App using **Python**

3. Grading Scheme:

- Participation/Exercises/Quizzes: 40%
- Graded Homework: 30%
- Term Project: 30%

Get Python

1. Go to <https://www.python.org/downloads/>. Click "*Download Python 3.11*".
 - **Be very careful for the next step!!**
2. After downloading, follow the instruction to finish installation.
 - On the first screen of installation, **enable** the "**Add Python 3.11 to PATH**" option.
 - If you forget the step above, you have to **uninstall** Python in Settings and **reinstall**.



Social Time

Answer the following question(s):

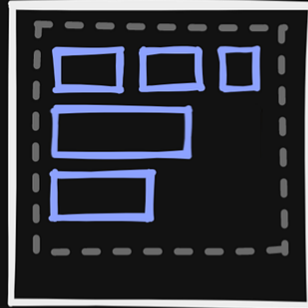
1. Do you plan to learn more on **Web Dev** before graduation (or in near future)?
2. Do you plan to learn other **programming language(s)** before graduation (or in near future)?
3. Do you plan to **keep your project website** after end of semester?
4. What question(s)/mystery(-ies) has the class answered/clarified for you? What are you still wondering about?

Responsive Design and Bootstrap

1. Introduction to Bootstrap
2. Bootstrap demo: [App Landing Page](#)

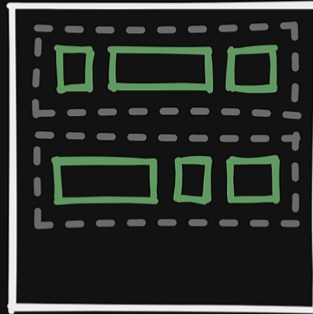
CSS Layout Modes

flow



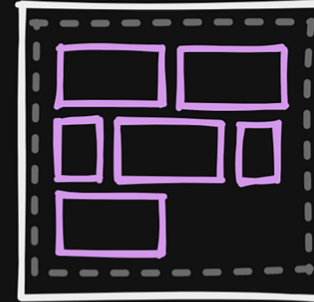
this is the default.
elements flow like text

flexbox



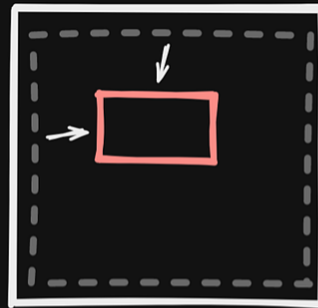
one-dimensional layout
system

grid



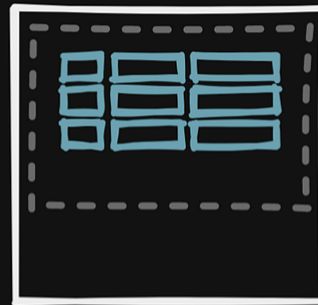
two-dimensional layout
system

positioned



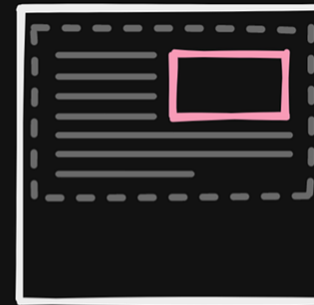
extract from normal flow
to a precise position

table



for structured tabular
data

float



for floating images inside
of text. occasionally used
for layouts by masochists

Introducing React by Gibson Njine

Progress in 14 Weeks

- 14 weeks ago:
 - you never wrote **HTML/CSS/JavaScript**
 - you knew **nothing** or **little** about programming
 - you never used **VS Code/GitHub**
 - you never created/owned a **website**
- **Today:**
 - well, let us take a look...

What we have learned in this semester...

1. HTML
2. CSS
3. JavaScript
4. and many other stuff
 - i. Git and GitHub
 - ii. VS Code
 - iii. Responsive Design, i.e. Bootstrap
 - iv. SEO
 - v. Domain and Hosting
 - vi. ...

Recommendations

- [W3Schools](#)
- [MDN Web Docs](#)
- GitHub [Explore](#) and [popular topics](#)
- [r/webdev](#)
- Many MOOCs
 - Coursera/udacity/edx/lynda/Khan...
 - YouTube Channels, e.g. [freecodecamp](#), [Traversy Media](#)
 - Udemy
 - [Full Stack Open Course](#)
 - [The Odin Project](#)

What else to learn

- More Web - [A visual overview of useful skills to learn as a web developer](#)
 - [CMS](#), i.e. [WordPress](#)
 - More JavaScript, i.e. [node.js](#)
 - Web application framework, i.e. [React](#), [Vue.js](#)
 - YouTube: [Web Development In 2022 - A Practical Guide](#)
- Others - [roadmap.sh](#)
 - Other programming languages
 - [Python/Java/TypeScript/Go/Rust/Solidity...](#)
 - 2022 [Stack Overflow Developer Survey](#)
 - UI/UX Design
 - Becoming a product manager

Being a Product Manager (PM)

- Understand demand
- Collect demand
- Analyze demand
- Convert demand to tasks
- Project management
- ...repeat after launching the product

Being a PM: Front-end PM/Feature PM/UI PM

- Technical skill sets:
 - UI, i.e. Figma/Sketch
 - A/B testing
 - Front-end experience
 - Data Analytics
- Interacting with other roles:
 - UI/UX Designer
 - Engineers
 - Data Analyst/Data Scientists
 - User Researchers
 - Users

Being a PM: Back-end PM/PM-Technical

- Technical skill sets:
 - Depending on technologies of the product
 - i.e. cloud infrastructure, database products, APIs, ML algorithm
 - Data
 - Basic Coding/Software Development
- Interacting with other roles
 - UI/UX Designer
 - Engineers
 - Data Analyst/Data Scientists
 - User Researchers
 - Users

Session 26 - Creating a web app using *Python*

Introduction to Web Application - Flask

- What is **Flask**?
 - Flask is a micro web framework written in Python.
- Popular sites using Flask:
 - Whole website: **Pinterest, Twilio**
 - In tech stack:
 - Reddit
 - LinkedIn
 - Netflix
 - Lyft
 - ...

Flask - Routing

- What is routing?

Modern web applications use meaningful URLs to help users. Users are more likely to like a page and come back if the page uses a meaningful URL they can remember and use to directly visit a page.

- Examples:

- <https://www.MySuperAwesomeVlog.com/new>
- <https://www.MySuperAwesomeVlog.com/1>
- <https://www.MySuperAwesomeVlog.com/1/edit>

Session 26 - Creating a web app using *Python*

1. Installation

```
> python -m pip install flask  
# macOS/Linux users:  
> python3 -m pip install flask
```

2. Creating our first Flask application

3. Turn on **Debug Mode**

4. Resources:

- [Flask Quickstart](#)
- [Flask Web App Demo](#)
- [HTML for Flask](#)
- [10 Best Flask Tutorials For Beginners in 2022](#)

Thank you