FRESNO STATE

Discovery. Diversity. Distinction.

Web programming (CSci 130)

Department of Computer Science

College of Science and Mathematics

California State University Fresno

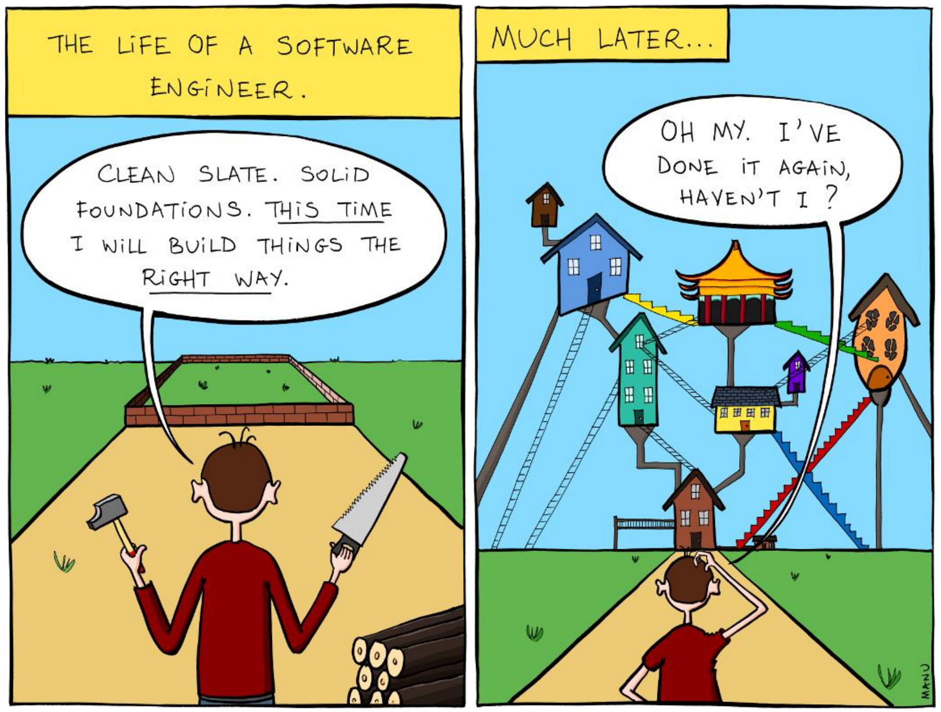
H. Cecotti

Learning outcomes

- Conclusion about Web Programming
    - Putting pieces together for success
        - In the final
        - In Job interviews
        - In your career
- Last minute questions about the project
    - Use software engineering practice to be more efficient!!!
    - Difficulties:
        - Server/Client communication
            - POST/GET
        - Asynchronous requests
            - AJAX
        - Syntax transferring data from one format to another
            - Description of objects using classes in JS and PHP

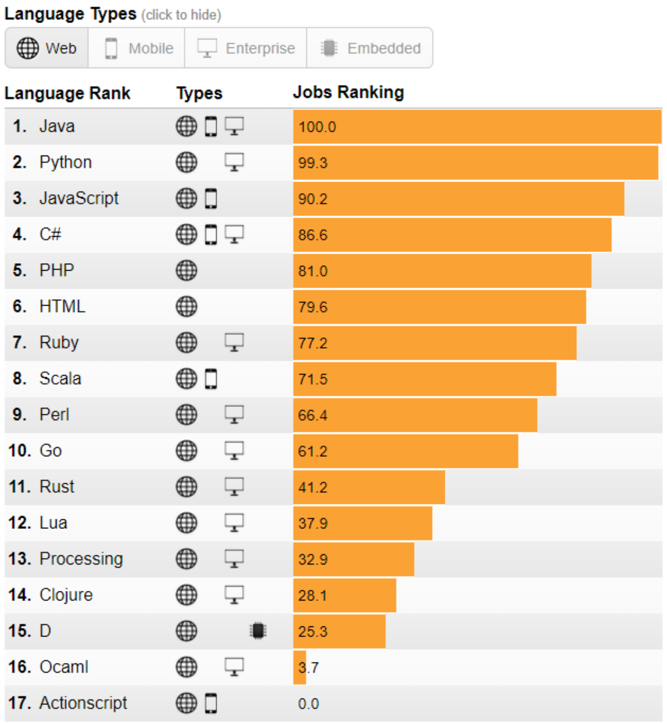
Project management

- Additional project outcomes
  - Don’t start late
  - Don’t expect it will work according to your “plan”
  - Main rule
    - 30% of the time to do 70% of the **estimated** work to do
    - 70% of the remaining time to do 30% of the **estimated** work to do
      - It takes more time to fix bugs, code refactoring


THE LIFE OF A SOFTWARE ENGINEER.
CLEAN SLATE. SOLID FOUNDATIONS. THIS TIME I WILL BUILD THINGS THE RIGHT WAY.
MUCH LATER...
OH MY. I'VE DONE IT AGAIN, HAVEN'T I ?
MANU

Web Programming

- IEEE Spectrum (July 2017)

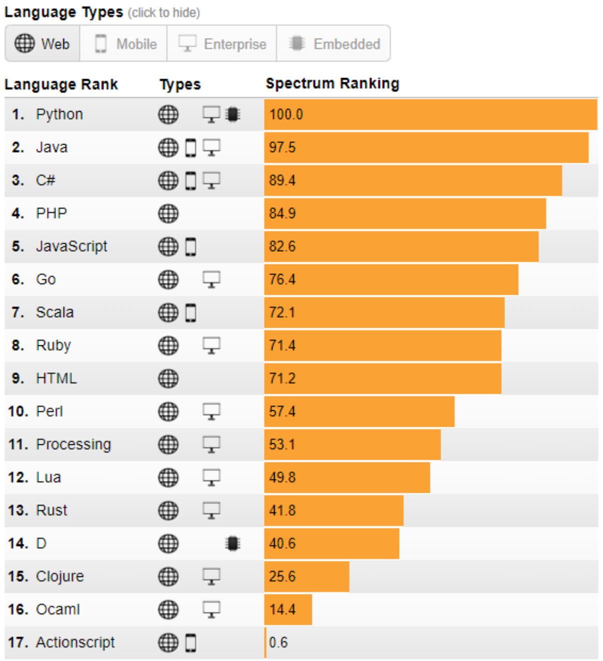


Language Types (click to hide)

🌐 Web | 📱 Mobile | 🖥 Enterprise | ⚙ Embedded

Language Rank	Types	Jobs Ranking
1. Java	🌐 📱 🖥	100.0
2. Python	🌐 🖥	99.3
3. JavaScript	🌐 📱	90.2
4. C#	🌐 📱 🖥	86.6
5. PHP	🌐	81.0
6. HTML	🌐	79.6
7. Ruby	🌐 🖥	77.2
8. Scala	🌐 📱	71.5
9. Perl	🌐 🖥	66.4
10. Go	🌐 🖥	61.2
11. Rust	🌐 🖥	41.2
12. Lua	🌐 🖥	37.9
13. Processing	🌐 🖥	32.9
14. Clojure	🌐 🖥	28.1
15. D	🌐 ⚙	25.3
16. Ocaml	🌐 🖥	3.7
17. Actionscript	🌐 📱	0.0

(July 2018)

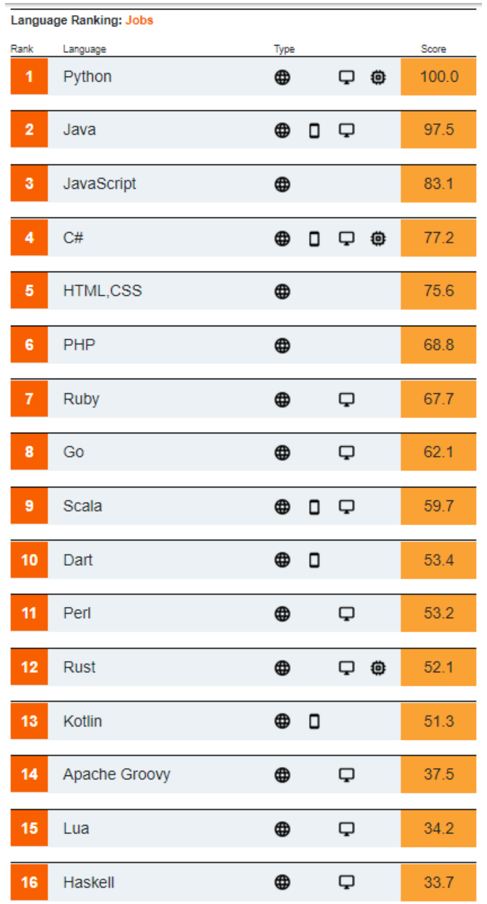


Language Types (click to hide)

🌐 Web | 📱 Mobile | 🖥 Enterprise | ⚙ Embedded

Language Rank	Types	Spectrum Ranking
1. Python	🌐 🖥 ⚙	100.0
2. Java	🌐 📱 🖥	97.5
3. C#	🌐 📱 🖥	89.4
4. PHP	🌐	84.9
5. JavaScript	🌐 📱	82.6
6. Go	🌐 🖥	76.4
7. Scala	🌐 📱	72.1
8. Ruby	🌐 🖥	71.4
9. HTML	🌐	71.2
10. Perl	🌐 🖥	57.4
11. Processing	🌐 🖥	53.1
12. Lua	🌐 🖥	49.8
13. Rust	🌐 🖥	41.8
14. D	🌐 ⚙	40.6
15. Clojure	🌐 🖥	25.6
16. Ocaml	🌐 🖥	14.4
17. Actionscript	🌐 📱	0.6

(July 2019)

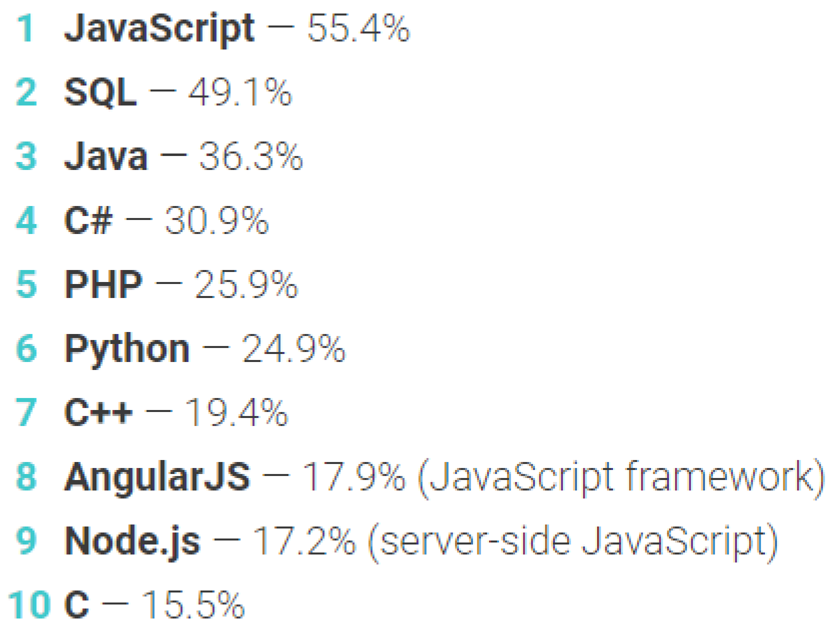


Language Ranking: Jobs

Rank	Language	Type	Score
1	Python	🌐 🖥 ⚙	100.0
2	Java	🌐 📱 🖥	97.5
3	JavaScript	🌐	83.1
4	C#	🌐 📱 🖥 ⚙	77.2
5	HTML,CSS	🌐	75.6
6	PHP	🌐	68.8
7	Ruby	🌐 🖥	67.7
8	Go	🌐 🖥	62.1
9	Scala	🌐 📱 🖥	59.7
10	Dart	🌐 📱	53.4
11	Perl	🌐 🖥	53.2
12	Rust	🌐 🖥 ⚙	52.1
13	Kotlin	🌐 📱	51.3
14	Apache Groovy	🌐 🖥	37.5
15	Lua	🌐 🖥	34.2
16	Haskell	🌐 🖥	33.7

Web Programming

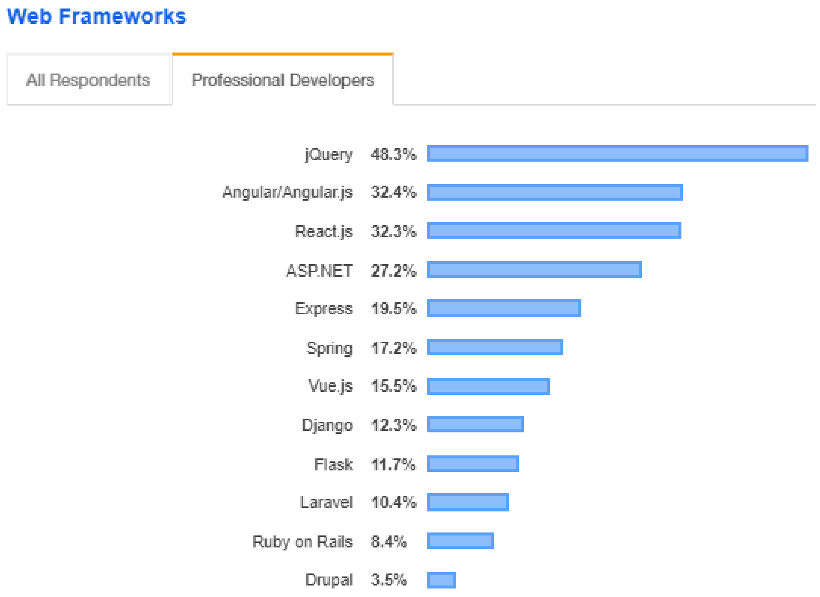
- Stack Overflow Developer Survey (2016)
  - ➢Most used technologies:

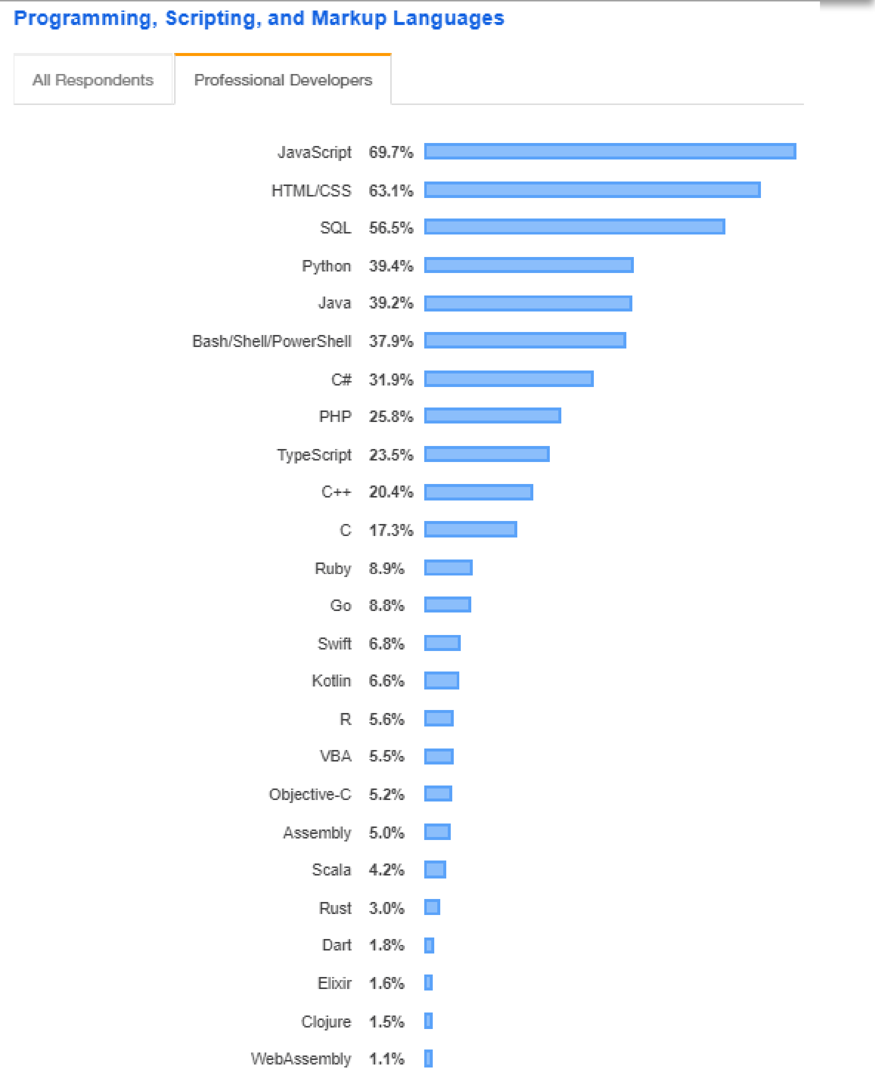


1 **JavaScript** — 55.4%
2 **SQL** — 49.1%
3 **Java** — 36.3%
4 **C#** — 30.9%
5 **PHP** — 25.9%
6 **Python** — 24.9%
7 **C++** — 19.4%
8 **AngularJS** — 17.9% (JavaScript framework)
9 **Node.js** — 17.2% (server-side JavaScript)
10 **C** — 15.5%

Web Programming

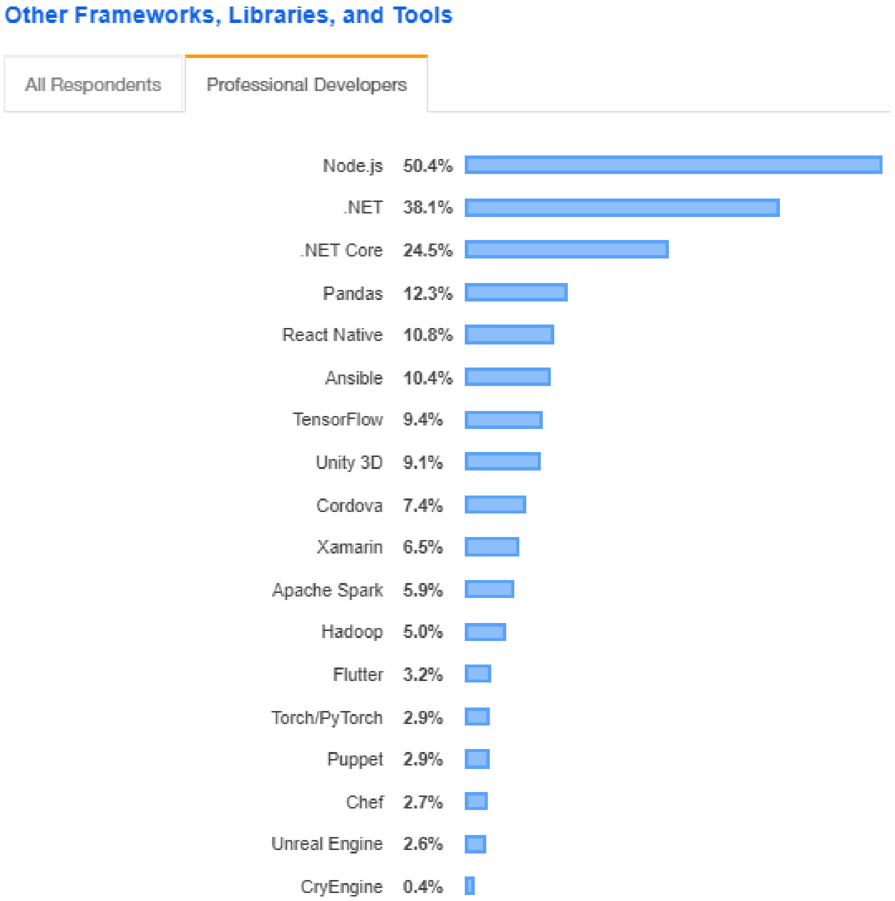
- Stackoverflow survey 2019

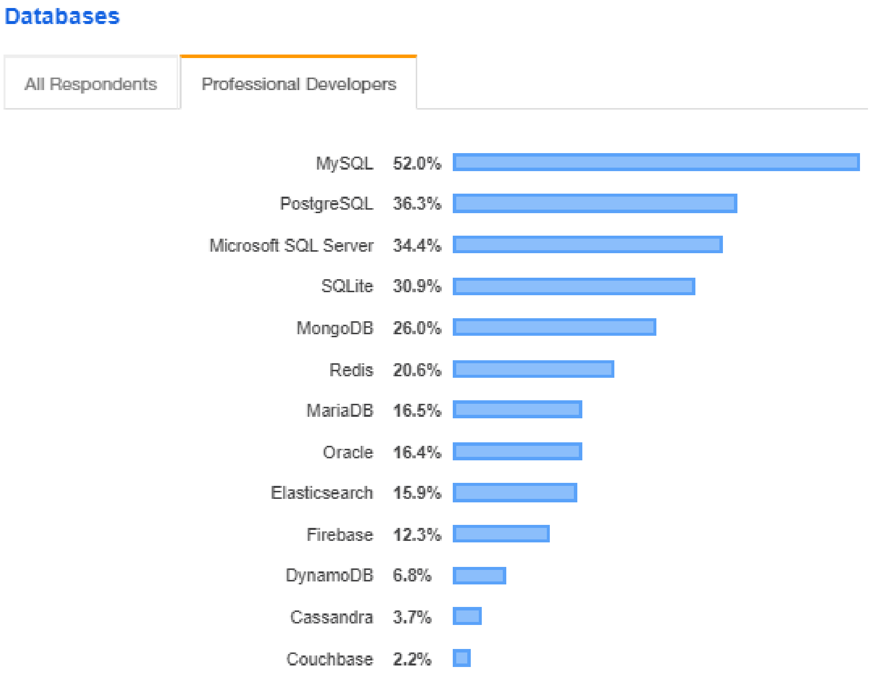




Web Programming

- Stackoverflow survey 2019



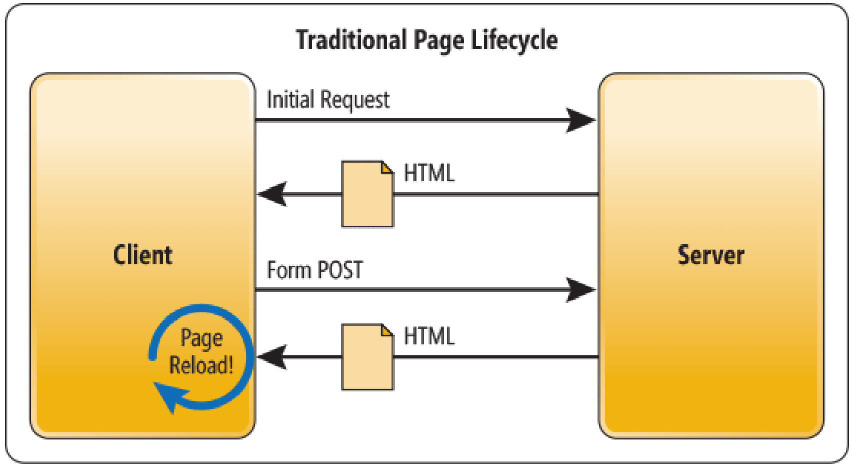


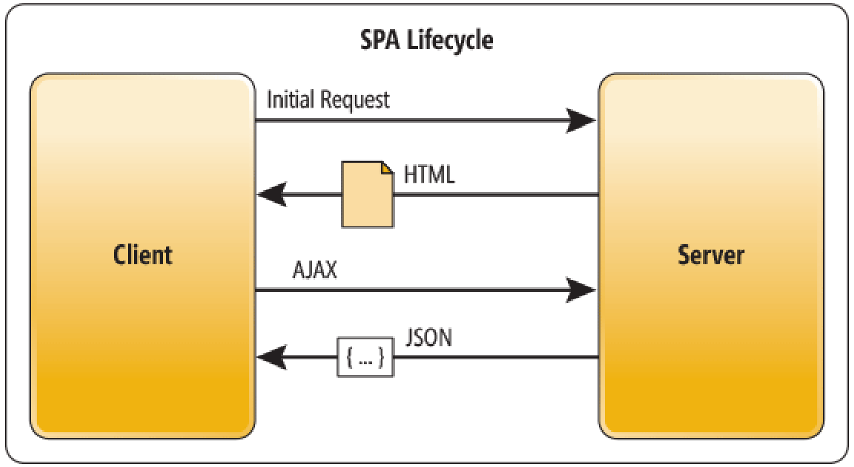
Web Programming

- Web pages
  - HTML: structure
  - CSS: Presentation
  - Javascript: Programming on the client side
    - Use the practice of the last versions (const, let, classes,…)
  - AJAX: Communication Client/Server through the XMLHttpRequest Object
  - PHP: Programming on the server side
- Principles
  - Separation of Concerns
    - Use of OOP
      - Classes (Objects, Properties, Methods)
    - Separation GUI/
      - Model View Controller (MVC)
- Paradigm shift
  - **Documents**/Pages → **Data**

Web Programming

- Where the work is happening?
  - ➢Balance between client-side and server-side
- **Traditional Page Lifecycle**
  - ➢POST → submit button
  - ➢More on the server side (PHP)
- **Single Page Application** (SPA)
  - ➢AJAX+JSON
  - ➢More on JS





Between what you learnt and what you know

- You may not know what you learnt
  - Awareness about what you are able to do
  - PHP/JS Frameworks
    - Minimum time to know how it works
    - Thanks to what you know in PHP and JS
- Job interview
  - “Can you do … ?”
  - **Yes.**

Rationale

- More frameworks?
  - “The measure of intelligence is the ability to change.”
— Albert Einstein
- Do not expect to use the same language and technologies all your life
  - Theoretical concepts
    - Not new
    - Mixture of old and new ideas
- If you use frameworks, read the documentation !!!
- Frameworks come and go
  - React developer = can only do React, no react → no job
  - JS developer = can do all JS frameworks

Mobile programming

- Evolution
  - ➢ Thin client layer → More emphasis on the client → Application on the client side
  - ➢ More clients → More challenges to manage concurrency on the server side
- Online presence for a company
  - ➢ Website
  - ➢ Social media
    - Facebook, Twitter, Instagram …

  - ➢ Mobile app
    - Android, iOS, …
    - C# with Xamarin
    - → Same principles with GUI connected to a database
      - MFC model
      - Separation UI / code to do something
    - Challenge
      - What goes on the client side (thin client?)
      - What goes on the server side (database, extra functions…)

About the final

- **Information**
  - 30 % of the final mark
  - Duration: 2h
  - Documents are not allowed
    - **Bring your laptop as the test is online**
    - No electronic devices (tablet, phone, smart wrist watch,…)
  - You will get reminders about the time during the exam
  - On Canvas
- **Structure** (fill the blank + multiple choices)
  - HTML: around 20 questions
  - CSS: around 16 questions
  - AJAX, JSON, PHP, COOKIE: around 25 questions
  - PHP-MySQL: around 15 questions
- **Go fast**: key definitions
- **Be precise with your answers!!!!**

About the final

- **Examples**
  - → Fill the blank
  - Definitions
  - Piece of code to complete
- Key concepts
- Everything from examples that were given **during the class or labs**
- **Be precise**
  - JSON (types: string, number,…   brackets: {} [])

About the final

- Description of classes and objects in JS and PHP
- AJAX
  - Send + POST/GET
- JSON
  - Serialize: to convert an object into that string
    - **JS**: JSON.stringify()
    - **PHP**: json_encode
  - Deserialize: to convert a string into an object
    - **JS**: JSON.parse()
    - **PHP**: json_decode()

Key elements

- What you can find on **Canvas**
  - Creation of a **dynamic table**
    - See lecture about DOM
    - Example: you get the output of a query from PHP+MySQL →
      - You must know how to present the table
  - Forms
  - AJAX + JSON, AJAX + MySQL
    - See in particular lectures from week 9,10,11
  - **Login**
  - Data browser with
    - JSON
    - MySQL

Positions

- **Front-End** Web Development
    - HTML, CSS3, JavaScript
- **Back-End** Web Development
    - PHP, MySQL, Python, Ruby
- **Full-stack**
    - Both the front-end and Back-End categories
    - Expertise in all layers of a website's development
        - Client, server, UX, hosting,…
- **Key skill**
    - To communicate with everybody, people of marketing ☺
    - You must be able to present, communicate with others, maintain your code

Conclusion

- From **documents**, web pages → **Data** (images, signals, documents)
  - “Forget about the web, because the web is everywhere”
  - Data scientists, Data engineers, Data miners,…
  - Focus on data
    - Query
    - Storage
    - Visualization
    - Information extraction
  - Connection to sensors to obtain Data
    - Internet of Things (IoT)
    - Data: not only Relational databases, XML, JSON
- Good luck for the final