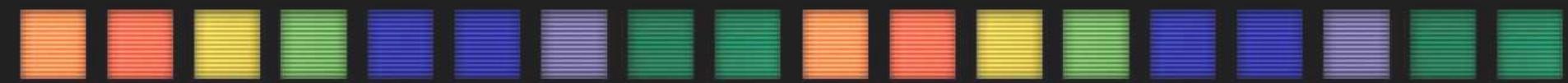




Introducción al Desarrollo Web

Ing. Marco Aedo López

CONCEPTOS BÁSICOS



Tema 2

PRESENTADO POR:
Ing. Marco Aedo López

Contacto:
maedol@unsa.edu.pe
marcoaedo@hotmail.com

Blog:
dutic.unsa.edu.pe
marcoaedo.blogspot.com

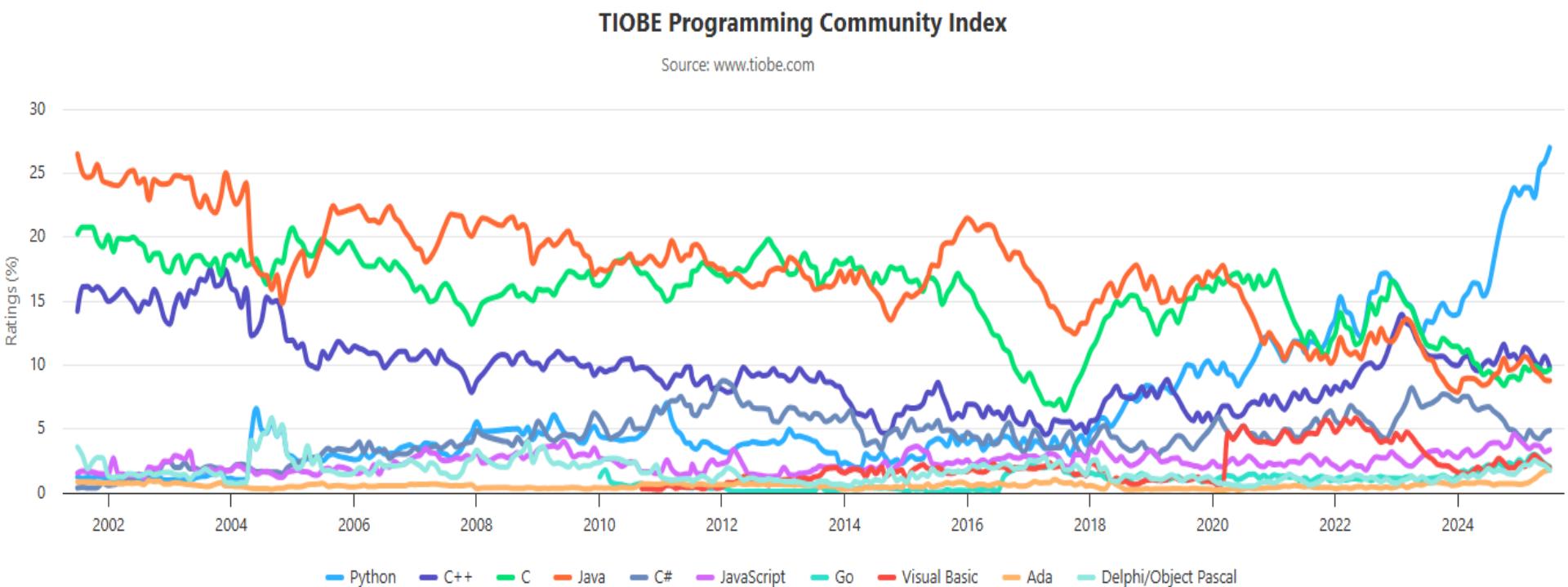
Objetivos

- Comprender los conceptos básicos relacionados a las herramientas utilizadas en el desarrollo web

Contenido

- Lenguajes de Programación
- Editores de Código
- Frameworks, Bibliotecas/Librerías
- Desarrollo Frontend (lado del cliente)
- Desarrollo Backend (lado del servidor)
- Desarrollo Fullstack
- Manejo de versiones con Git y GitHub
- WebAssembly
- Contenedores Docker

1. Lenguajes de Programación



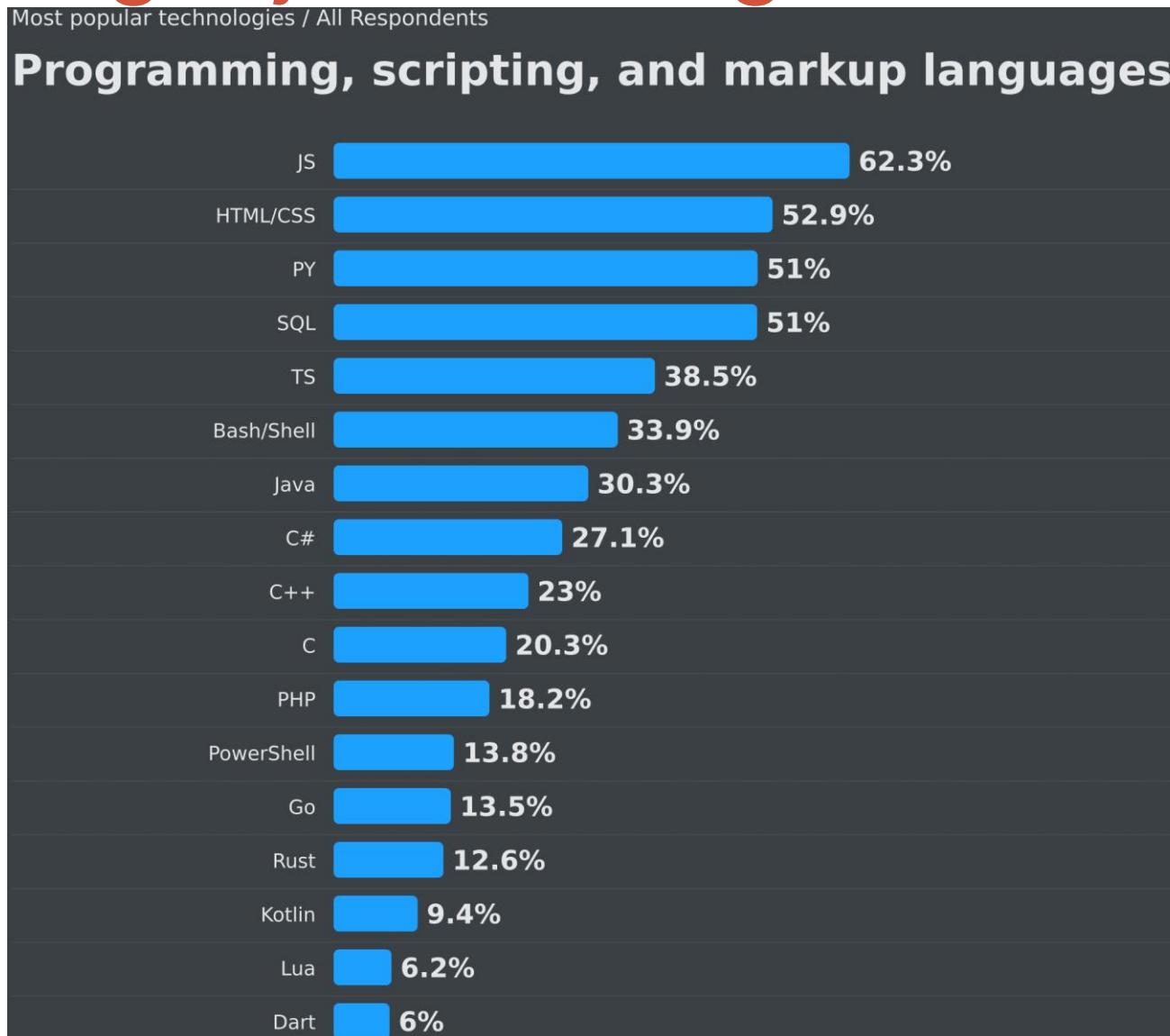
1. Lenguajes de Programación

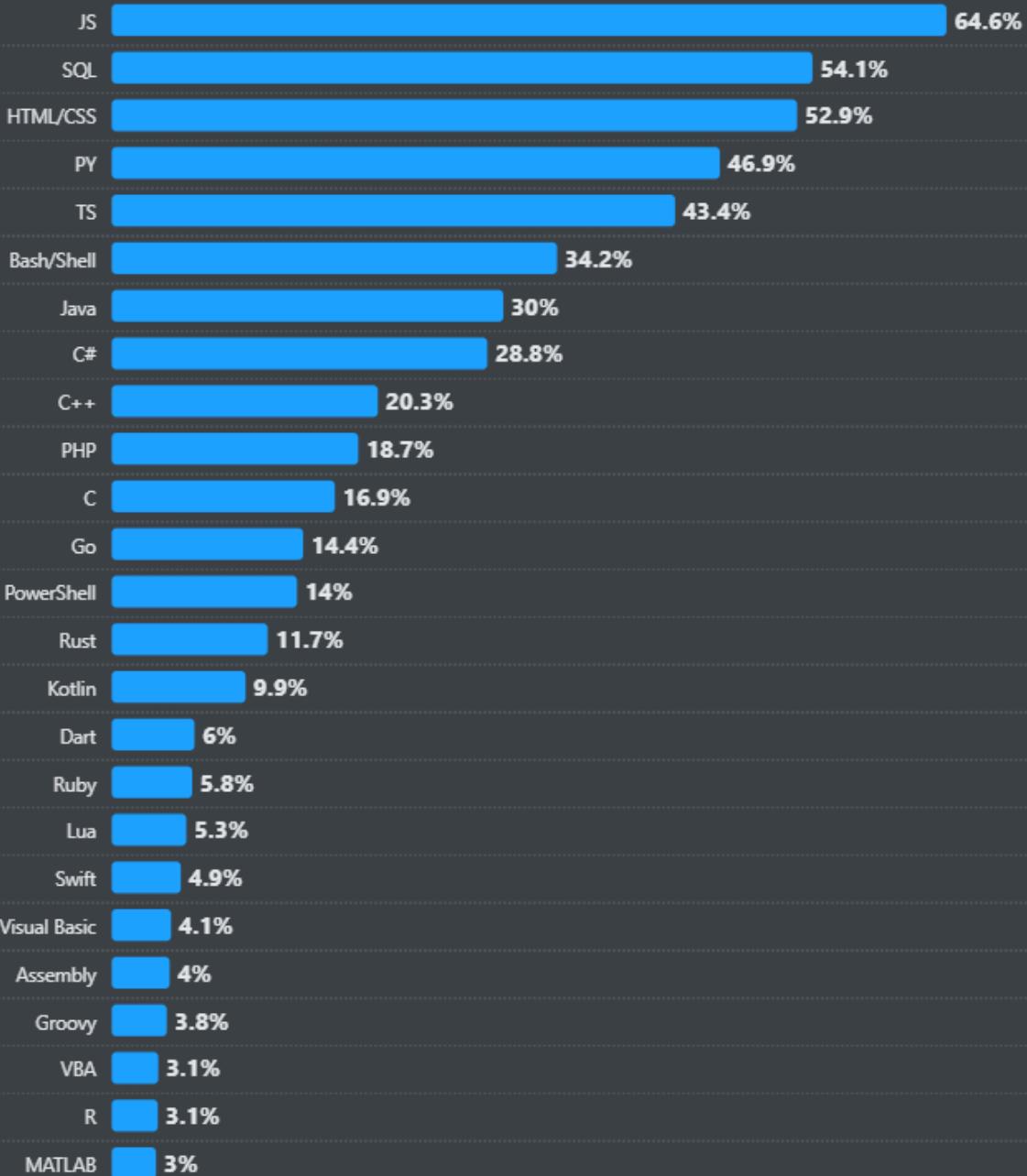
Jul 2025	Jul 2024	Change	Programming Language	Ratings	Change
1	1		 Python	26.98%	+10.85%
2	2		 C++	9.80%	-0.53%
3	3		 C	9.65%	+0.16%
4	4		 Java	8.76%	+0.17%
5	5		 C#	4.87%	-1.85%
6	6		 JavaScript	3.36%	-0.43%
7	7		 Go	2.04%	-0.14%
8	8		 Visual Basic	1.94%	-0.13%
9	24		 Ada	1.77%	+0.99%
10	11		 Delphi/Object Pascal	1.77%	-0.12%

1. Lenguajes de Programación

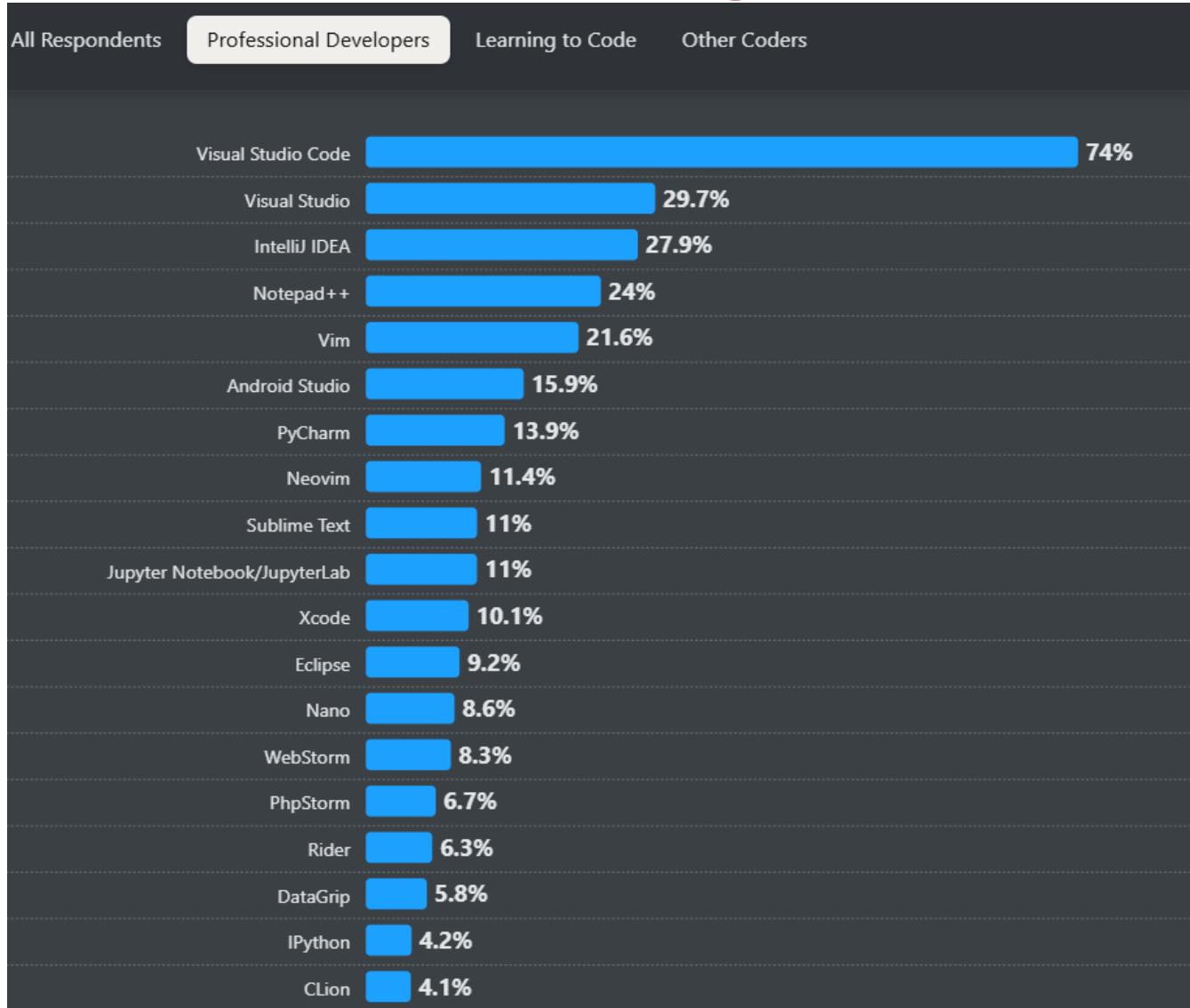
Most popular technologies / All Respondents

Programming, scripting, and markup languages





2. Editores de Código



3. Frameworks y Bibliotecas/Librerías

- El desarrollo web moderno se apoya en frameworks y librerías para acelerar el trabajo y garantizar buenas prácticas
- Un **framework** define cómo construir la aplicación
- Una **biblioteca/librería** es un conjunto de herramientas que usamos según necesidad

3. Frameworks y Bibliotecas/Librerías

- **Framework:** conjunto estructurado de componentes y reglas que guían el desarrollo de aplicaciones, promoviendo orden y reutilización de código
- Angular (JavaScript), Vue.js, Django (Python), Laravel (PHP), Spring Boot (Java), Express (JavaScript), Flask, etc.

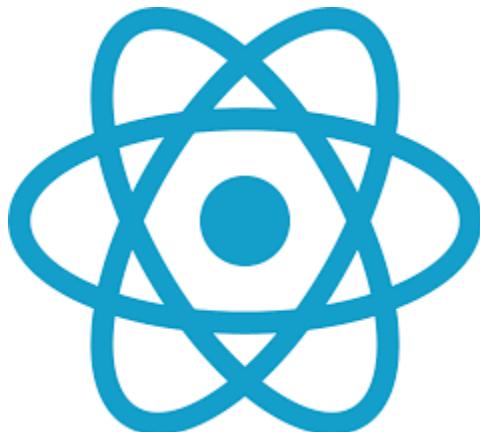


Express 



3. Frameworks y Bibliotecas/Librerías

- **Biblioteca/Librería:** Colección de funciones o recursos que se usan de forma puntual para resolver tareas específicas, sin imponer una estructura general
- React (JavaScript), Chart.js, JQuery, etc.

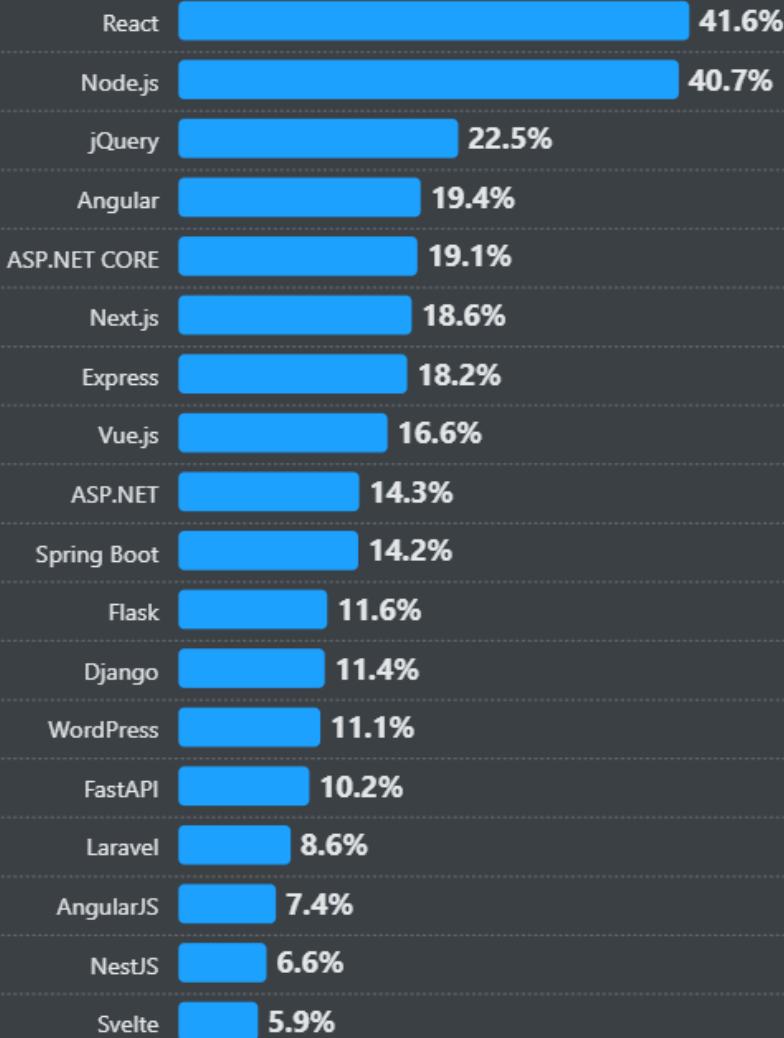


3. Frameworks y Bibliotecas/Librerías

Professional Developers

Learning to Code

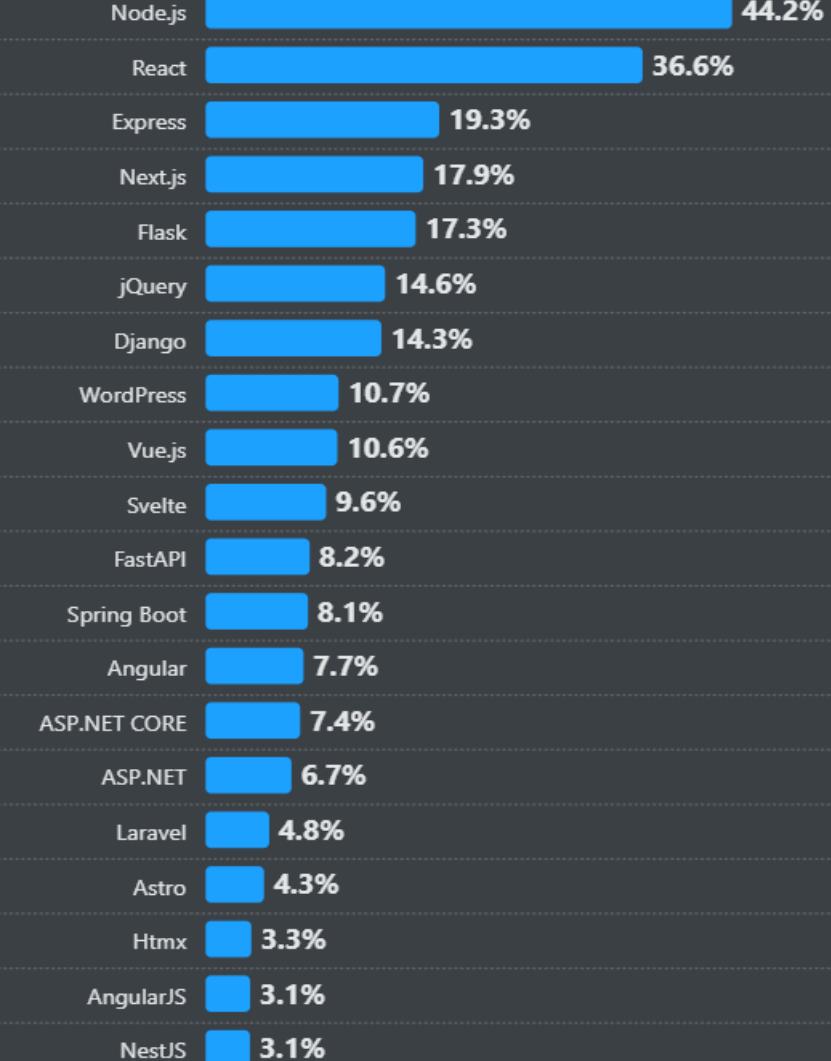
Other Coders



Professional Developers

Learning to Code

Other Coders



4. Desarrollo Frontend

- Lado del Cliente
- Parte del desarrollo web que se encarga de construir la interfaz con la que interactúa el usuario
- Lo que ves y usas en una página web o aplicación web: menús, botones, formularios, animaciones, colores, tipografía, etc.

HTML



CSS



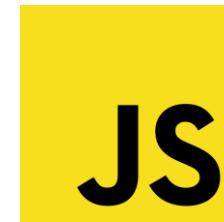
JavaScript



CSS

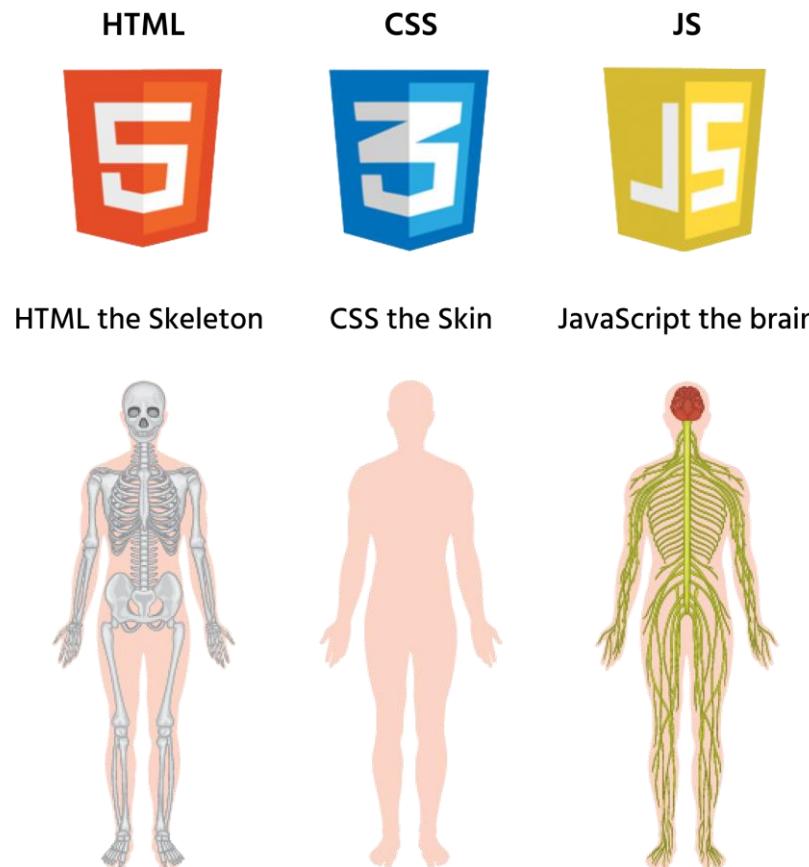


JS



4. Desarrollo Frontend

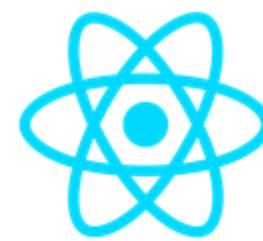
- HTML: estructura, contenido
- CSS: estilo, apariencia
- JavaScript: interactividad y lógica



4. Desarrollo Frontend



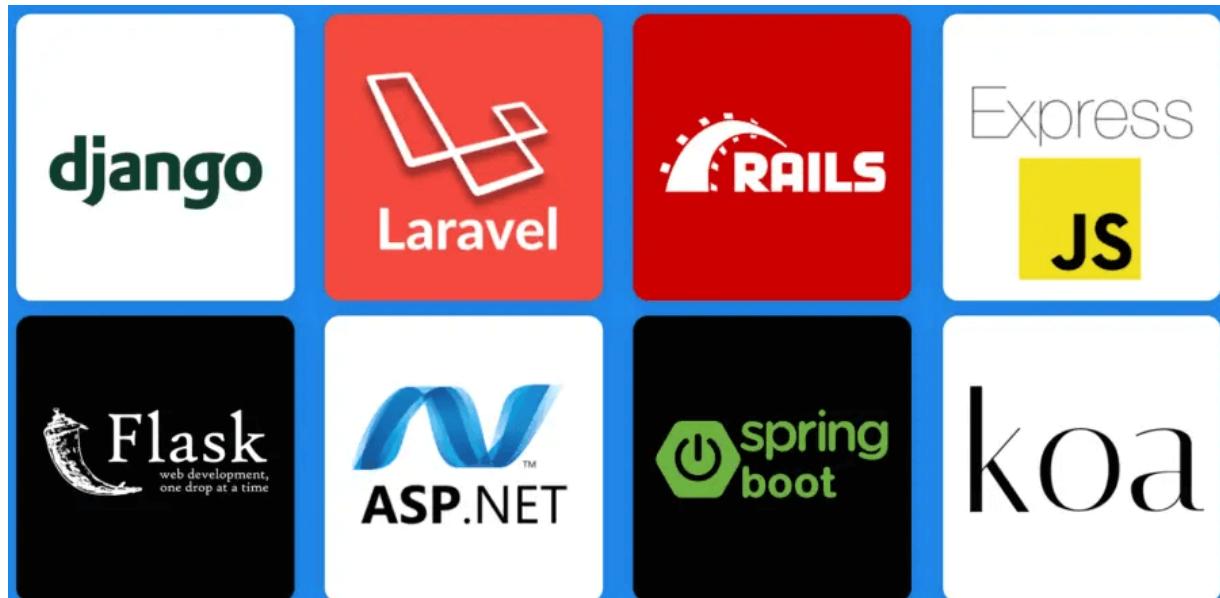
ANGULAR



REACT

5. Desarrollo Backend

- Lado del Servidor
- Parte del desarrollo web que se encarga de la lógica interna de una aplicación, procesamiento de datos y la comunicación con bases de datos y otros servicios
- No es visible directamente para el usuario final, pero es esencial para que la aplicación funcione
- Gestiona usuarios, permisos, sesiones, datos, seguridad, etc.



5. Desarrollo Backend

FRONTEND

parte que es visible, de la atención a los clientes y de coordinar con la cocina.

Principales funciones:

1. Tomar los pedidos de los clientes
2. Pasar los pedidos a la cocina.
3. Recibir los pedidos de la cocina.
4. Servir los pedidos a los clientes.



Principales desafíos:

Adaptarse a todo los tipos clientes.
Atender rápidamente a los clientes.
Lograr que los **clientes realicen pedidos**.

Se realizan pedidos a la cocina



La cocina entrega los pedidos

BACKEND

cocina, preparar los pedidos y dejarlos listos para que el front los sirva al cliente

Principales funciones:

1. Tomar los pedidos pasado por front.
2. Tomar los ingredientes
3. Preparar los pedidos.
4. Entregar los pedidos.

Principal desafíos:

Preparar de forma **segura**.
Prepara **varios pedidos a la vez**.
Disminuir el **tiempo de preparación**.

6. Desarrollo Full Stack

- Frontend + Backend
- Frontend: la parte visible para el usuario (interfaz, diseño, interactividad)
 - HTML, CSS, JavaScript, React, Angular, Vue, etc.
- Backend: la parte que procesa datos y lógica del servidor
 - Node.js, Express, Django, Laravel, Spring Boot, Bases de Datos, etc.



7. Otras Herramientas

- Git
 - Sistema de control de versiones
 - Permite guardar, seguir y gestionar los cambios en el código de un proyecto a lo largo del tiempo
 - Trabajar en equipo sin estropear el trabajo de otros
 - Deshacer cambios o volver a versiones anteriores
 - Llevar un historial completo del desarrollo



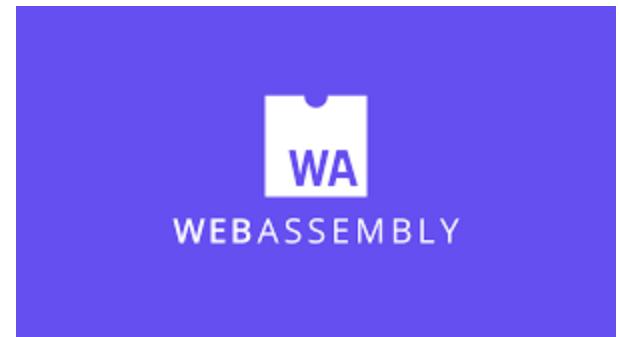
7. Otras Herramientas

- GitHub
 - Plataforma online (en la nube)
 - Aloja repositorios Git
 - Compartir código con otros desarrolladores
 - Trabajar en equipo con diferentes herramientas
 - Colaborar en proyectos de desarrollo de software



7. Otras Herramientas

- WebAssembly (Wasm)
 - Es un formato de código binario que permite ejecutar código de forma rápida en los navegadores web
 - Diseñado para ser un complemento de JavaScript, no un reemplazo
 - Permite que lenguajes como C, C++, Python, Rust, Go (y muchos otros) se compilen a WebAssembly
 - Así, ese código puede ejecutarse en la web con un rendimiento cercano al nativo
 - Velocidad y eficiencia



7. Otras Herramientas

- Contenedores Docker
 - Tecnología que permite empaquetar y ejecutar aplicaciones junto con todas sus dependencias (librerías, configuración, etc.) de forma aislada y portátil
 - Funcionan como “cajas ligeras” que incluyen sólo lo necesario para que una aplicación se ejecute, sin importar en qué sistema operativo se use
 - El Lenguaje de Programación X, con un framework Y y la base de datos Z, todos juntos



CONCEPTOS BÁSICOS



Gracias