

REPASO

DEV.F
DESARROLLAMOS(PERSONAS);

dev

Repaso

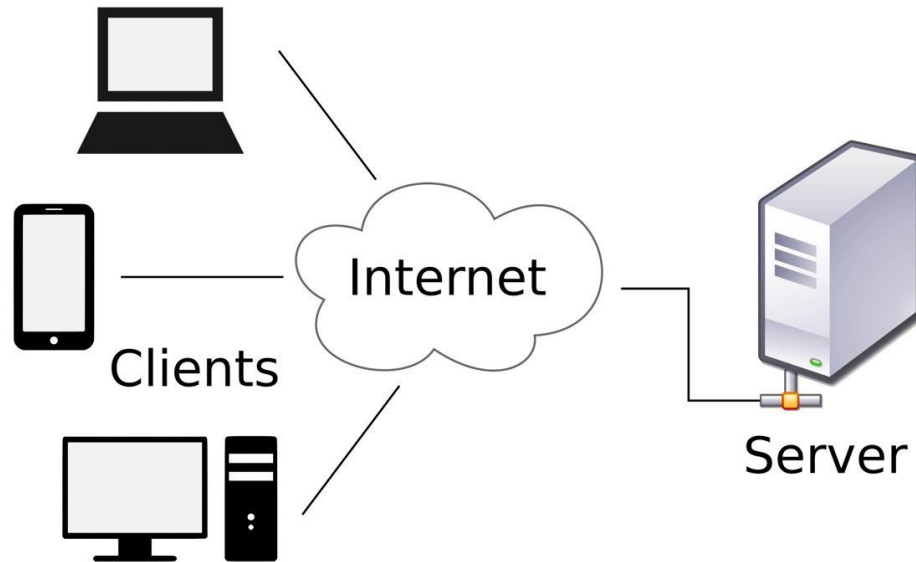
- Funciones, parámetros.
- Comandos en terminal.
- Git y github.
- Prototype y POO.
- ES6 (acceso a objetos y arreglos, arrow function, destructuring).

Conceptos de servidores web

DEV.F
DESARROLLAMOS(PERSONAS);

dev

Intro al modelo cliente servidor



Intro al concepto de URL

Uniform resource identifier, sirve para acceder a un recurso físico o abstracto por Internet. A continuación se muestran los elementos mínimos de una URL:

protocolo://dominio:port/path?

<https://www.tutorialesprogramacionya.com/javascriptya/nodejsya/detalleconcepto.php>

Localhost

Es la dirección de mi propio computador, también se le conoce como dirección loopback.

Localhost: <http://localhost:8080>

Loopback: <http://127.0.0.1:8080>

Mime Types

Los mime types indican la naturaleza y formato de los archivos que son transmitidos en la respuesta de una solicitud web.

- [Listado.](#)

Semantic Versioning

DEV.F
DESARROLLAMOS(PERSONAS);

dev

Semantic Versión

Es un conjunto simple de reglas y requerimientos que dictan cómo asignar e incrementar los números de la versión de un software. Evitan la pérdida de versiones y mejoran la gestión de dependencias.

1.2.3-beta.1+meta



The diagram illustrates the components of the Semantic Versioning string '1.2.3-beta.1+meta'. Horizontal lines are placed under each part of the string: '1', '2', '3', 'beta.1', and '+meta'. Vertical lines connect these underlines to labels below: 'Major' for '1', 'Minor' for '2', 'Patch' for '3', 'Pre-release' for 'beta.1', and 'Metadata' for '+meta'.

Major Minor Patch Pre-release Metadata

Funcionamiento de semantic versión

Dado un número de versión **MAYOR.MENOR.PARCHE**, se incrementa:

- La versión **MAYOR** cuando realizas un cambio incompatible en el proyecto.
- La versión **MENOR** cuando añades funcionalidad que compatible con versiones anteriores.
- La versión **PARCHE** cuando reparas errores compatibles con versiones anteriores.

MAYOR.MENOR.PARCHE = MAJOR.MINOR.PATCH.

Ejemplo: 1.2.1

Entendiendo conceptos

- Servicio \approx endpoint \approx recurso \approx servicio web.
- API \approx API REST \approx API Restful.
- Request = Petición = Solicitud.
- Response = Respuesta.
- Path \approx Dirección \approx Url \approx URI.
- File = Fichero = Archivo.
- Terminal = Línea de comandos = CLI = Bash.
- Script \approx Programa \approx Conjunto de líneas de código o instrucciones.
- Entidad (BD) \approx Clase (Programación) \approx Recurso (API).

