# APIS

Son datos de un backend. Datos que pueden ser propios o no. Por ej, puedo consultar el estado del clima.

* La API siempre me **devuelve un JSON**
* **Me permite compartir información con el cliente externos sin darle acceso a mi servidor.**
* **Es una consulta a la base de datos y parsear la info.**
* **Por ej. En mi base de datos tengo datos de las direcciones de farmacias, info del dueño, de sus datos personales, pero sólo quiero acceder a las direcciones de las farmacias.**
* **¿CÓMO armo la API? Con el ejemplo anterior, Hago la consulta a la base de datos:**

1. **Select dirección from farmacias**
2. **obtengo un choclazo de info**
3. **a esa info la filtro con php y la trabajo como un array**
4. **php me permite convertir ese array a JSON**

* **eso es una API. Ya puedo compartir con el mundo exterior info de la base de datos, para que el mundo pueda consumir esa información.**

Es acceder a la base de datos pero sólo a información parcial.

El 80% de las API del mundo están en formato JSON de JAVA.

API REST: necesita que me comunique con esta base de datos de una forma muy particular.

# JSON

* **es un String (**al igual que en PHP)
* **El objeto JSON empieza y termina con comillas simples**
* **Las propiedades van con comillas dobles** (esto los diferencia del objeto literal)
* **JAVA tiene en su objeto JSON dos métodos que me permiten trabajar con el parseo de esa información:**

1. **PARSE**
2. **STRINGIFY**

* ***PARSE:* recibe un string y lo transforma en objeto literal.**
* ***STRINGIFY:* hace lo contrario.**

Es lo que tiene JAVA para usar librerías.

Como escribo un objeto en JAVA

{nombre:”Patricia”}

Si quiero agregarle …

{nombre: “Patricia”, edad:25, mascotas: [“gato”,”perro”]}

Como convertir nuestro objeto literal en formato JSON: agregar las comillas que están en rojo.

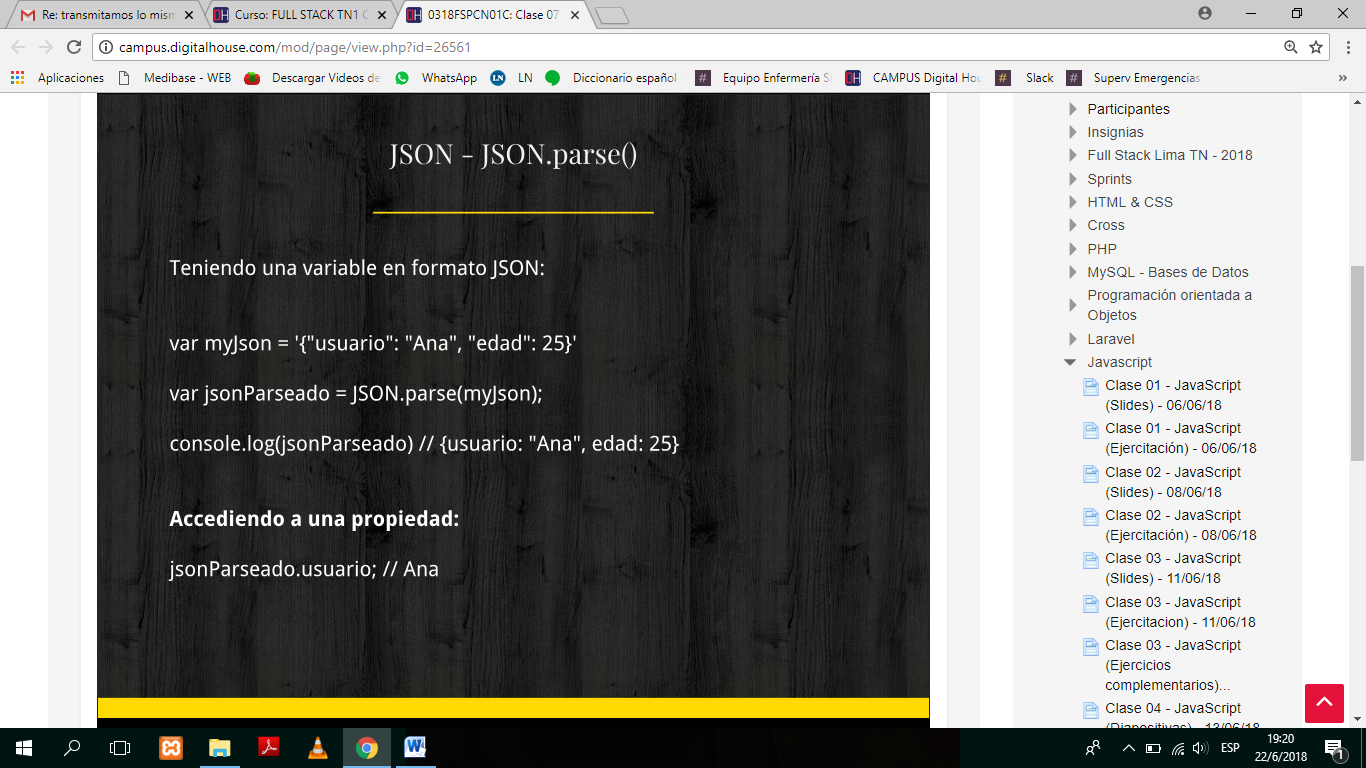
‘{“nombre”:”Patricia”, ”edad”:25, ” mascotas”: [“gato”,”perro”]}’

El OBJETO JSON en JAVA comienza con comilla simple y las propiedades van con comilla doble.

El OBJETO JSON es un String y no se puede hacer nada con él.

En JAVA tiene dos métodos. 1.Parse 2. Stringify

1. Parse: te convierte un objeto JSON de JAVA en un objeto literal. Es lo que más se usa para trabajar.



Luego de ese objeto parseado, le puedo pedir sus propiedades, en este caso en el ejemplo me devuelve “Ana”. ¿cómo le pido sus propiedades?

Jsonparseado.usuario;

Jsonparseado.edad;

1. Stringify

Ajax

Pedidos que yo lanzo y que no sé cuando los obtengo.

Hay 3 formas de trabajarlo, pero veremos sólo 2.

Base de datos no relacional, es aquella que no tiene columnas, ni foreign key, es simplemente un JSON, un string

En response.text: obtengo el JSON!!!!!!

Tercer paso:

Con el método open.

A la variable le agrego un punto y luego el método:

Por ej:

Cuarto paso: SEND

Hay que ejecutar a la instancia del objeto el método .send

Se recomienda bajar JSON FORMATTE.

Web: rest countries. Si pasás la url que te da te trae lo que elegiste

Con esa info que trajo la API puedo incorporarla a un formulario para hacer un SELECT.

Recordar que no puedo usar el forEach para iterar el objeto.

Lo ideal es for in.

Es un elemento iterador de objetos. Si escribo esto:

For( var país in data)

Me trae el INDICE!!! ☹

Lo ideal es hacerlo así:

Console.log (data[país]);

Si quiero una propiedad, agrego un punto y el nombre de la propiedad:

Console.log (data[país].name);

Hay una api de superhéroes, de MARVEY

FETCH api

Es una manera sencilla de traer promesas.

Si no le paso ningún parámetro, por defecto es GET.