

Práctica 1. Selección de arquitecturas y herramientas de programación

Antonio Jesús Pérez Ruiz



Índice

1. Diferencia, con tus propias palabras, entre páginas web, aplicación de escritorio y aplicaciones web.
2. Instala el software XAMPP. Redacta todo el proceso de instalación e incluye capturas de pantalla.
3. Configura XAMPP para su correcto funcionamiento. Redacta todo el proceso e incluye capturas de pantalla. Muestra que funciona accediendo a <http://localhost/phpmyadmin>
4. Crea tu primera página con PHP. Copia el código en la práctica y muestra capturas de pantalla de su localización en el directorio y del resultado al acceder al archivo por el navegador.
5. Instala Node.js. Redacta todo el proceso de instalación e incluye capturas de pantalla. Comprueba su correcto funcionamiento.
6. Instala Typescript. Redacta todo el proceso de instalación e incluye capturas de pantalla. Comprueba su correcto funcionamiento.
7. Crea un servidor web con Node.js. Redacta todo el proceso e incluye capturas de pantalla. Muestra que funciona ejecutando el archivo de creación del servidor y accediendo por navegador al servidor web.
8. Bibliografía

Práctica 1. Selección de arquitecturas y herramientas de programación

1. Diferencia, con tus propias palabras, entre páginas web, aplicación de escritorio y aplicaciones web.

Páginas web: Son sitios en línea accesibles a través de un navegador web. Muestran información y servicios en línea, como blogs o tiendas, y no requieren instalación en tu dispositivo.

Aplicaciones de escritorio: Son programas que se instalan en tu computadora y funcionan sin conexión a Internet. Ejemplos incluyen Microsoft Word y Adobe Photoshop.

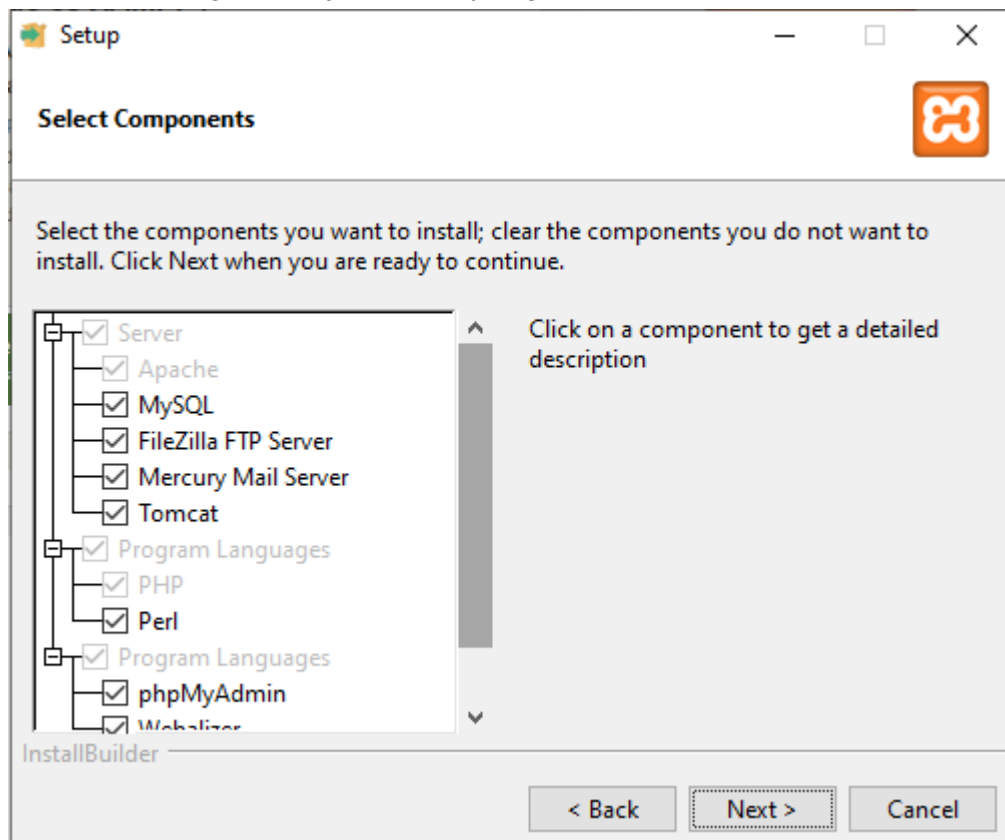
Aplicaciones web: Son programas en línea que se ejecutan en servidores web y se acceden mediante un navegador. Ejemplos son Gmail y Google Drive. Dependes de la conexión a Internet para usarlas.

2. Instala el software XAMPP. Redacta todo el proceso de instalación e incluye capturas de pantalla.

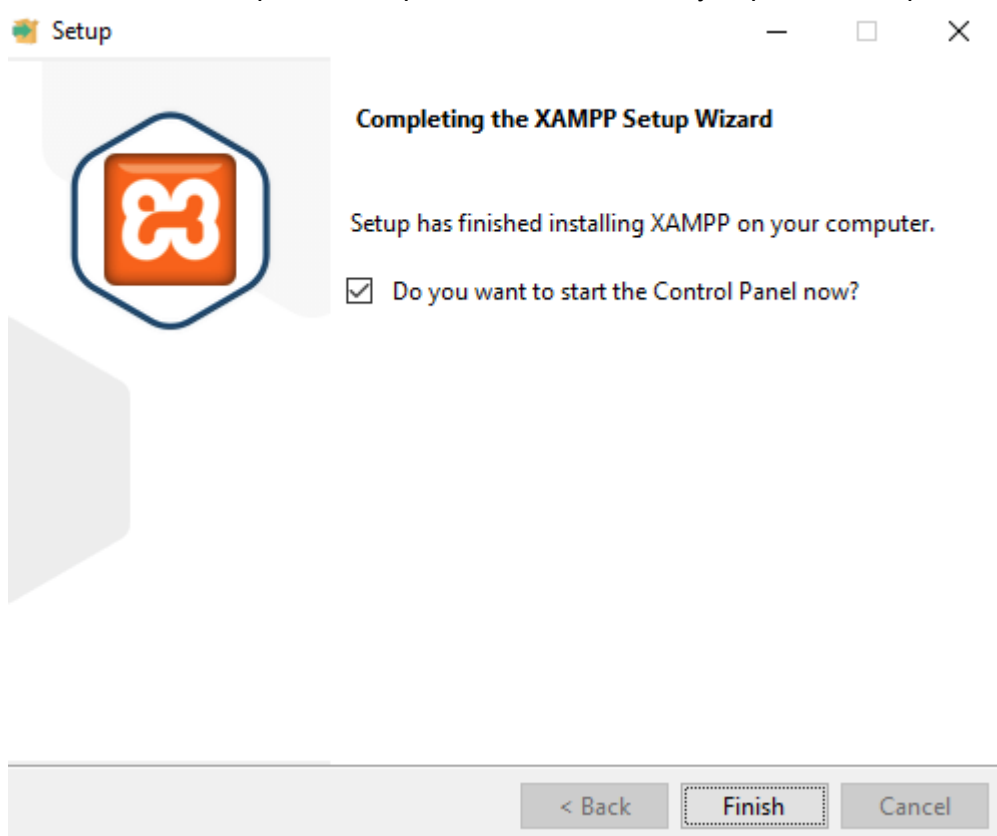
Comenzamos entrando a la página oficial de XAMPP y descargamos el ejecutable.



Una vez descargado lo ejecutamos y seguimos los pasos de instalación.

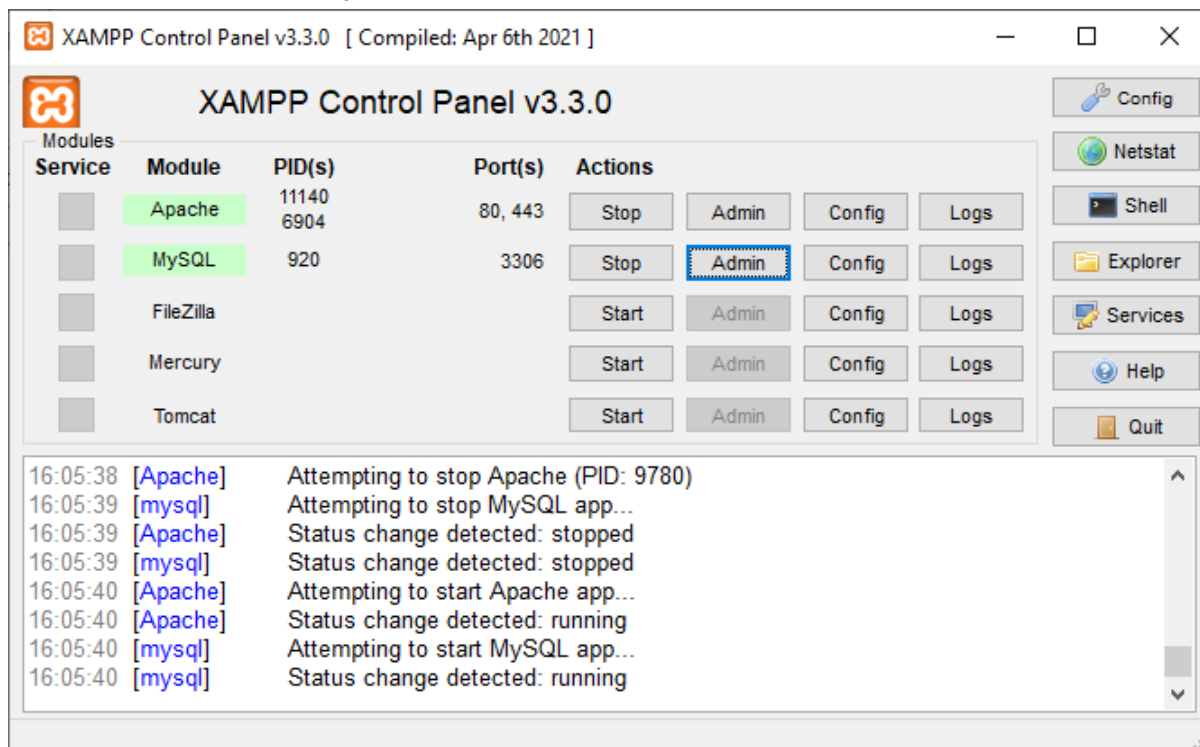


Seleccionamos los productos que deseemos instalar y esperamos a que se instale.

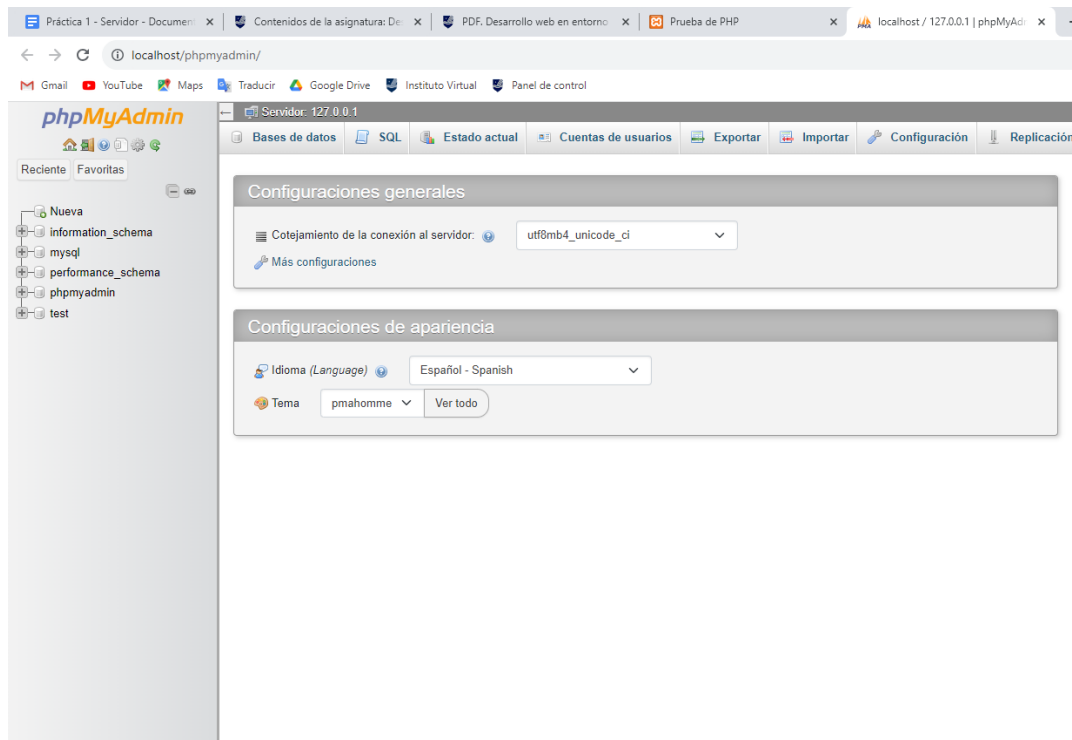


3. Configura XAMPP para su correcto funcionamiento. Redacta todo el proceso e incluye capturas de pantalla. Muestra que funciona accediendo a <http://localhost/phpmyadmin>

Abrimos XAMPP y le damos a Start en Apache y MySQL, y una vez se muestren en verde accedemos a Admin en MySQL.



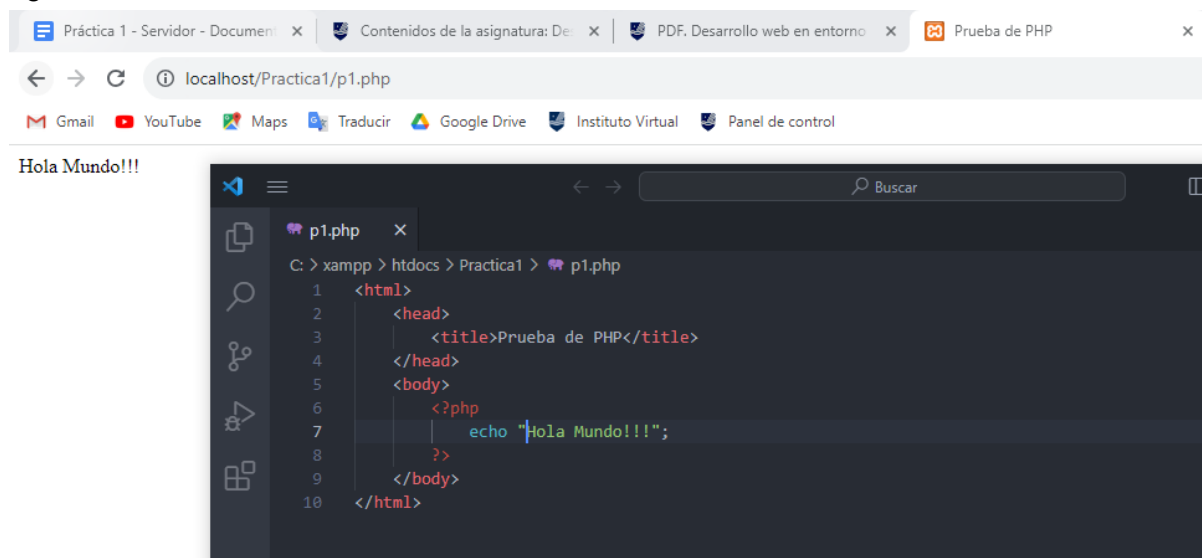
Se nos abrirá <http://localhost/phpmyadmin>



4. Crea tu primera página con PHP. Copia el código en la práctica y muestra capturas de pantalla de su localización en el directorio y del resultado al acceder al archivo por el navegador.

He creado el archivo en xampp > htdocs > Practica1, llamado "p1.php"

En el navegador escribiremos en mi caso (<http://localhost/Practica1/p1.php>) y mostrará lo siguiente:



5. Instala Node.js. Redacta todo el proceso de instalación e incluye capturas de pantalla. Comprueba su correcto funcionamiento.

Comenzamos entrando a la página oficial de Node.js y descargamos el ejecutable, yo he descargado la versión actual para Windows.

The screenshot shows the Node.js download page. The 'Actual' version (20.7.0) is highlighted. Below the version information, there are links to download the installer for Windows, macOS, and Linux. The 'Actual' version is recommended for the majority of users.

Descargas

Versión actual: 20.7.0 (includes npm 10.1.0)

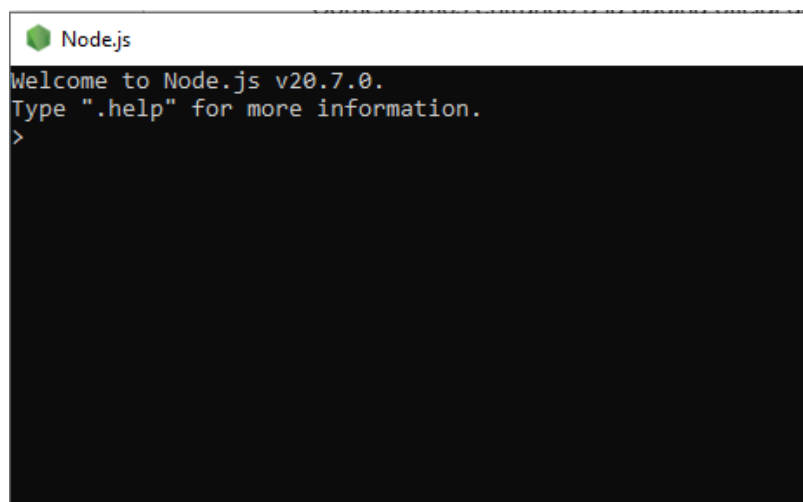
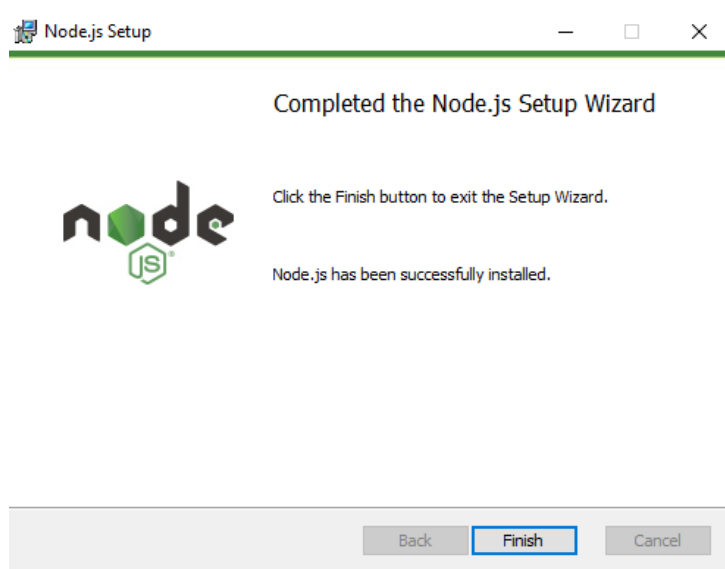
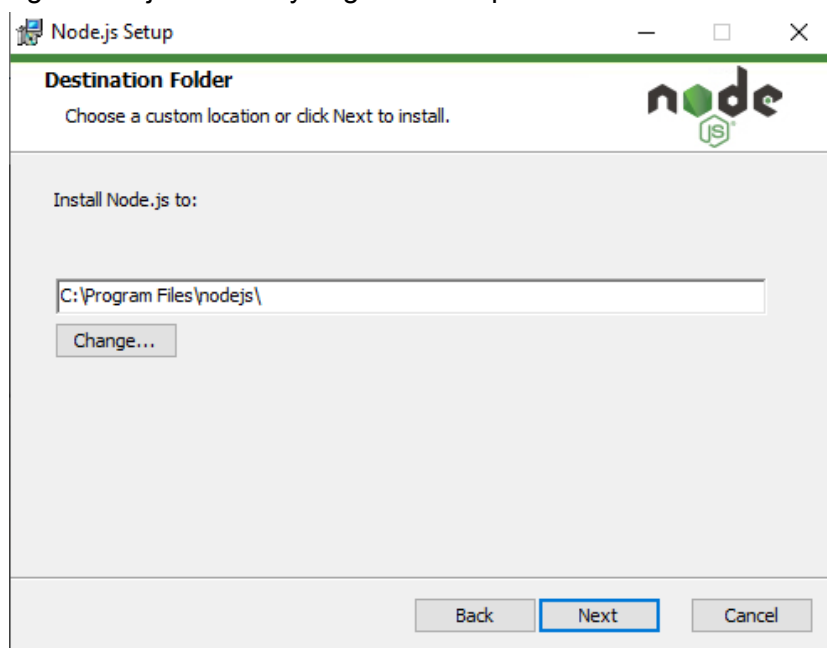
Descargue el código fuente de Node.js o un instalador pre-compilado para su plataforma, y comience a desarrollar hoy.

LTS	Actual
Recomendado para la mayoría	Últimas características
Instalador Windows	Instalador macOS
node-v20.7.0-x64.msi	node-v20.7.0.pkg
Código Fuente	node-v20.7.0.tar.gz

Instalador Windows (.msi)	Binario Windows (.zip)	Instalador macOS (.pkg)	Binario macOS (.tar.gz)	Binario Linux (x64)	Binario Linux (ARM)	Código Fuente
32-bit	64-bit	ARM64	ARM64	ARM64	ARM64	node-v20.7.0.tar.gz
64-bit	ARM64	ARM64	ARM64	ARM64	ARM64	node-v20.7.0.tar.gz
ARMv7	ARMv8	ARMv8	ARMv8	ARMv8	ARMv8	node-v20.7.0.tar.gz

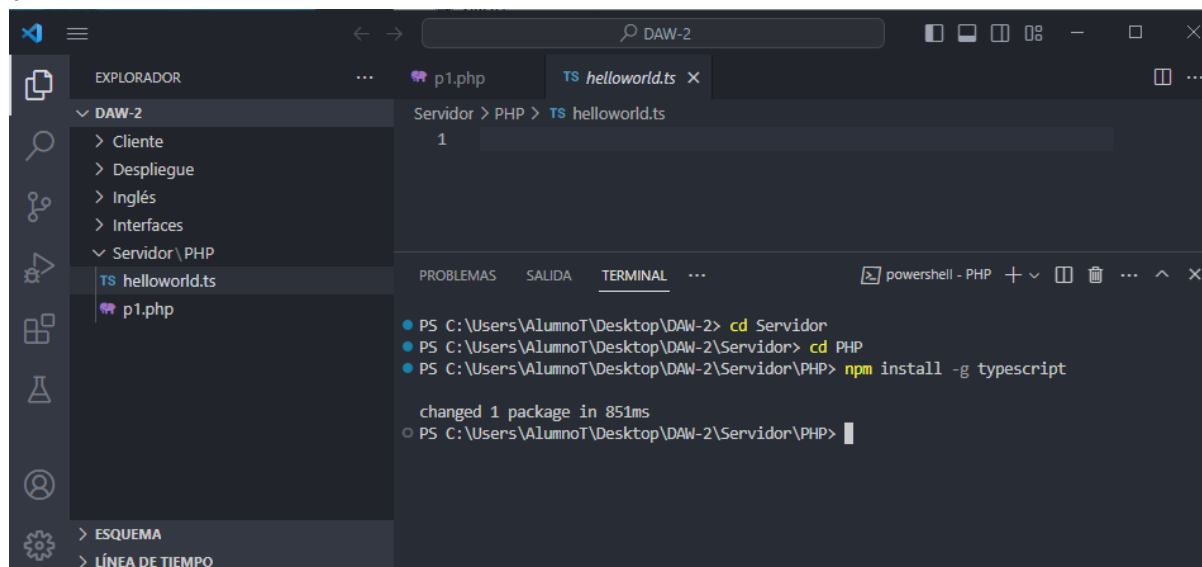


Una vez descargado lo ejecutamos y seguimos los pasos de instalación.



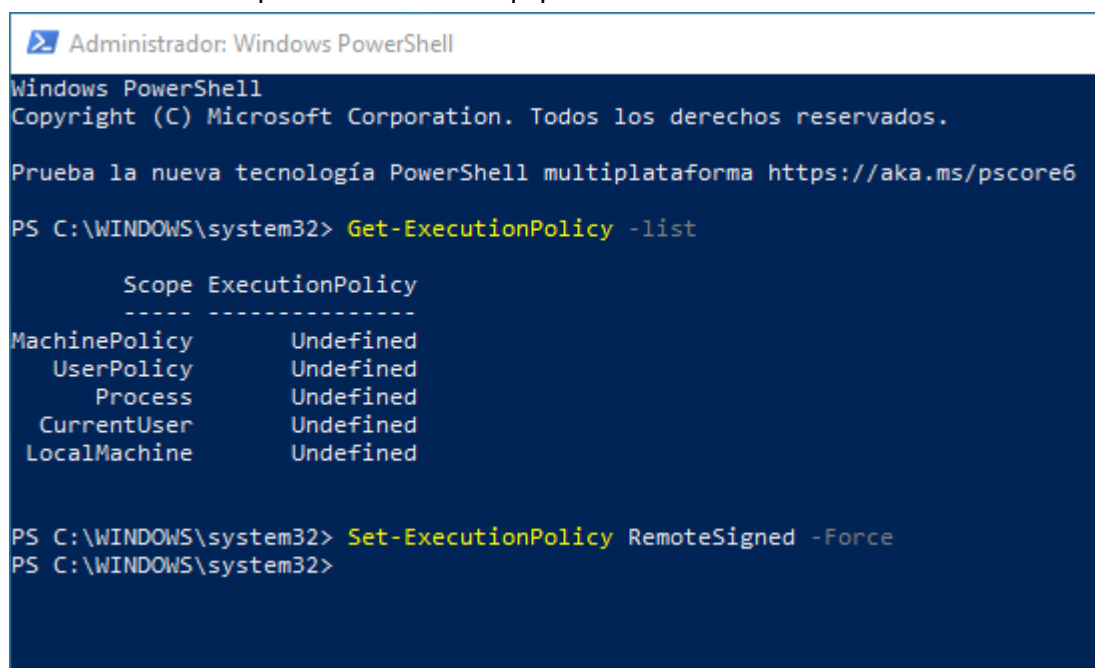
6. Instala Typescript. Redacta todo el proceso de instalación e incluye capturas de pantalla. Comprueba su correcto funcionamiento.

Comenzamos dirigiéndonos al terminal de visual e introducimos el comando `npm install -g typescript`



Nos dirigimos a una carpeta y creamos un archivo llamado “helloworld.ts” en mi caso. Volvemos al terminal y ponemos el comando “`tsc helloworld.ts`” y este nos debería crear un archivo llamado helloworld.js.

En caso de que nos de error por estar deshabilitada la ejecución de scripts vamos a Windows Powershell y usamos los comandos `Get-ExecutionPolicy -list` y `Set-ExecutionPolicy RemoteSigned -Force` para habilitar la ejecución de scripts que suele venir deshabilitada por defecto en el equipo.





Comprobamos que haya creado el archivo javascript

```
DAW-2
```

```
Server > PHP > TS helloworld.ts
```

```
1
```

```
PROBLEMAS SALIDA TERMINAL
```

```
PS C:\Users\AlumnoT\Desktop\DAW-2> cd Servidor
```

```
PS C:\Users\AlumnoT\Desktop\DAW-2\Servidor> cd PHP
```

```
PS C:\Users\AlumnoT\Desktop\DAW-2\Servidor\PHP> npm install -g typescript
```

```
changed 1 package in 851ms
```

```
PS C:\Users\AlumnoT\Desktop\DAW-2\Servidor\PHP> tsc helloworld.ts
```

```
PS C:\Users\AlumnoT\Desktop\DAW-2\Servidor\PHP>
```

7. Crea un servidor web con Node.js. Redacta todo el proceso e incluye capturas de pantalla. Muestra que funciona ejecutando el archivo de creación del servidor y accediendo por navegador al servidor web.

Con Node.js instalado creamos un archivo JavaScript y con el siguiente código escrito nos vamos al terminal y escribimos el comando `node .\nombre del archivo\js`

```
Practica1
```

```
JS joseillo.js
```

```
JS joseillo.js > ...
```

```
1 var http = require("http");
```

```
2 http.createServer(function(req, res) {
```

```
3   res.writeHead(200, {'Content-Type': 'text/html'});
```

```
4   res.write("<h2>Hola Mundo</h2>");
```

```
5   res.end();}).listen(8080);
```

```
PROBLEMAS SALIDA TERMINAL
```

```
PS C:\xampp\htdocs\Practica1> node .\joseillo.js
```

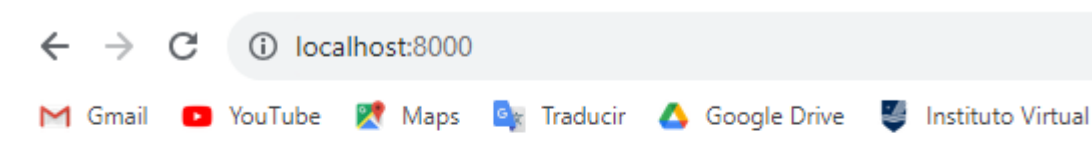
```
PS C:\xampp\htdocs\Practica1> node joseillo.js
```

```
PS C:\xampp\htdocs\Practica1> node .\joseillo.js
```

```
El servidor web implementado se encuentra en la dirección
```



Al entrar en el navegador deberíamos poder visualizar el contenido de la página entrando al puerto 8000 en mi caso.



Hola Mundo

Bibliografía

- *Node.js*. (s. f.). Node.js. <https://nodejs.org/es>
- MEDAC - *Instituto virtual* - Inicio de sesión. (s. f.).
https://medac.instructure.com/courses/42005/files/6538307?module_item_id=7218182
- *XAMPP installers and downloads for Apache Friends*. (s. f.).
<https://www.apachefriends.org/es/index.html>