



El futuro digital  
es de todos

MinTIC



# Protocolos HTTP

OPERADO POR:



Mision  
TIC 2022

ruta de aprendizaje 1

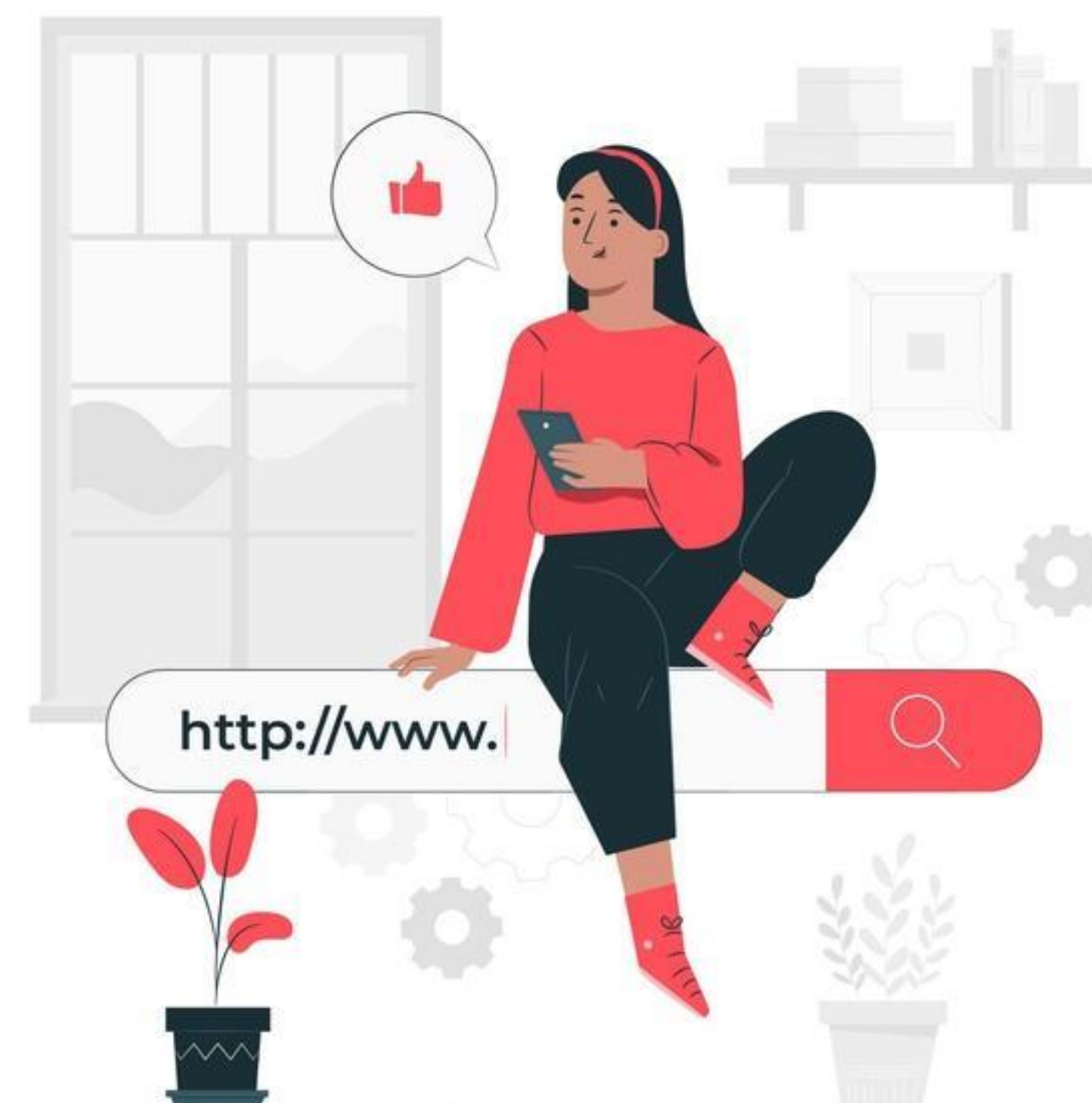




# Protocolo HTTP

HTTP, de sus siglas en inglés: "Hypertext Transfer Protocol", es el nombre de un protocolo el cual nos permite realizar una petición de datos y recursos, como pueden ser documentos HTML.

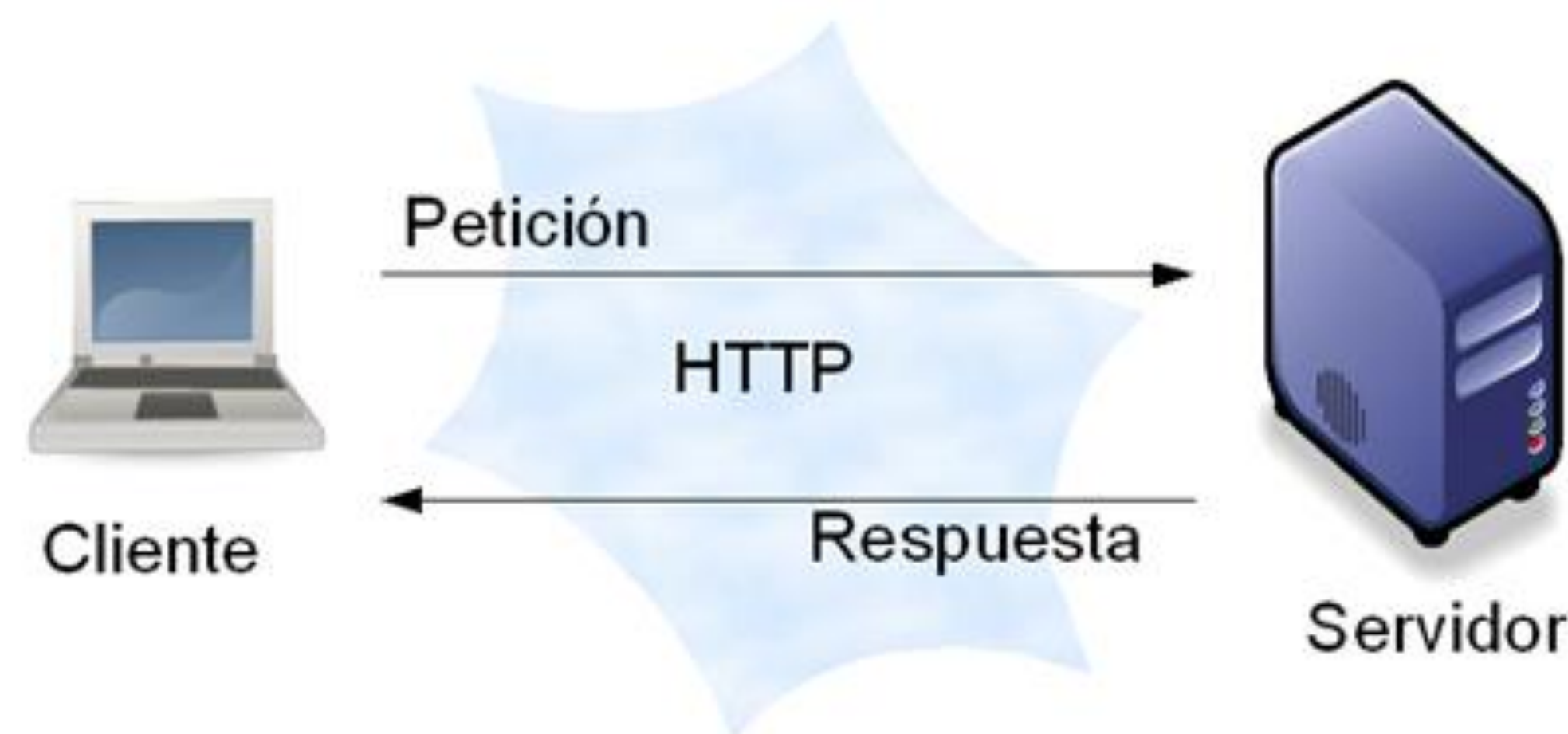
Es la base de cualquier **intercambio de datos** en la Web, y un protocolo de estructura cliente-servidor.





En este protocolo existen una serie de conceptos tales como:

- **Conexión:** es el circuito virtual establecido entre 2 programas en una red de comunicación
- **Mensaje:** es la unidad básica de un protocolo HTTP y consiste en una secuencia estructurada que se transmite entre los programas
- **Cliente:** es el programa que hace la llamada al servidor y es el que atiende en la transmisión la trama de los mensajes
- **Servidor:** es el programa que presta el servicio en la red
- **Proxy:** se trata de un programa intermedio cuya función puede ser variada, puede desde controlar el acceso desde ciertas direcciones, como filtrar contenido proviniendo de algunas direcciones







Así, pues, el protocolo **HTTP** se basa en la conexión entre cliente y servidor. Una transacción **HTTP** consiste básicamente en:

- **Conexión:** establecimiento de una conexión del cliente con el servidor.
- **Solicitud:** envío por parte del cliente de un mensaje de solicitud al servidor.
- **Respuesta:** envío por parte del servidor de una respuesta al cliente.
- **Cierre:** fin de la conexión por parte del cliente y el servidor.





## Servidores DNS

### Un servidor DNS:

Para que el cliente y servidor pueda establecer conexión e intercambiar hipertexto mediante el protocolo **HTTP** es necesario la participación de un servidor **DNS** este servidor.

Es un software que se encarga de traducir la **URL** una dirección **IP** es decir cuando escribimos <https://www.youtube.com/> el servidor **DNS** lo traduce a una dirección como la siguiente <https://208.65.153.238/>.

Esto es lo que permite que los enrutadores de internet puedan encaminar el paquete por la internet y este encuentre el destinatario.

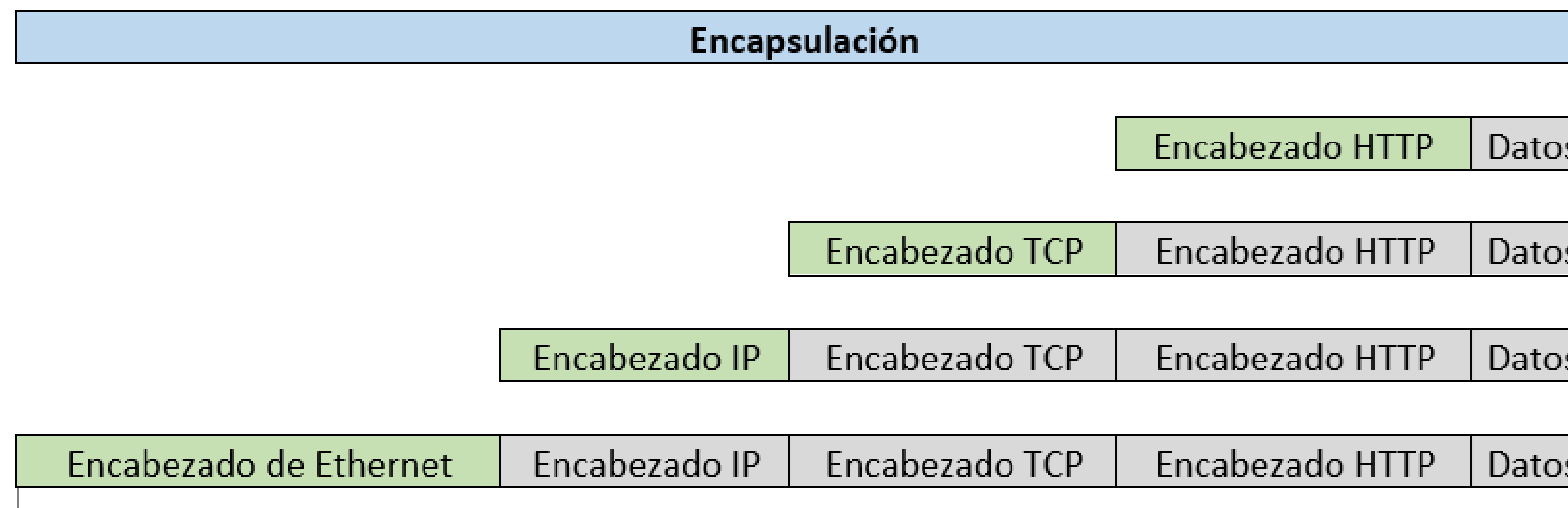






Para aclarar algunas dudas que se puedan presentar, los datos que viajan por una red se le denomina **paquete**, ya que los datos están **encapsulados** (empaquetados) de cierta manera, la manera de empaquetar estos datos es **determinada por los protocolos**.

El protocolo **HTTP** empaqueta los datos (Hipertexto) para que sean enviados correctamente y el protocolo **TCP/IP** le agrega otra capa al paquete con la información para que este llegue al destinatario.





vamos a hacer una analogía con un **problema de la vida real** para entender mejor los conceptos.

Imaginemos que queremos enviar una **docena de huevos** a un familiar en otro departamento, pero el envío lo realizamos por una empresa de **encomienda**.

Para que los huevos **no se quiebren**, debemos **empaquetarlos** en un cartón de huevos, sin embargo, no podemos simplemente entregar los huevos en el cartón, ya que los trabajadores de la empresa de encomienda no sabrán dónde llevarlo.





Para poder llevarlos debemos **empaquetarlos** en otro **contenedor** y en este **contenedor escribir los datos** a donde queremos que entregue nuestro paquete.

El proceso para enviar peticiones al servidor es similar, los **huevos** son los **datos** que enviaremos, el **encapsulado HTTP** es el **cartón** donde van los huevos para que estos no se rompan y el **empaquetado con la dirección** del destinatario es el **encapsulado TCP/IP**

Por eso en la imagen de la derecha, vemos que los datos se van encapsulando hasta llegar al encabezado de Ethernet

### Encapsulación

Encabezado HTTP	Datos
-----------------	-------

Encabezado TCP	Encabezado HTTP	Datos
----------------	-----------------	-------

Encabezado IP	Encabezado TCP	Encabezado HTTP	Datos
---------------	----------------	-----------------	-------

Encabezado de Ethernet	Encabezado IP	Encabezado TCP	Encabezado HTTP	Datos
------------------------	---------------	----------------	-----------------	-------





En resumen, explicaremos de forma básica como se realiza una comunicación entre un cliente y un servidor para acceder a una página web:

- Al colocar una URL en nuestro navegador (el cliente) realiza una **petición** a un servidor **DNS** para que le traduzca el nombre del **dominio** en una dirección **IP**
- Una vez se tenga la dirección **IP** del servidor donde se aloja la página que se desea visitar, el navegador envía un mensaje **(una petición)** mediante el protocolo **HTTP**. En este caso la petición es que le envíe los datos de la página **WEB** que desea visitar



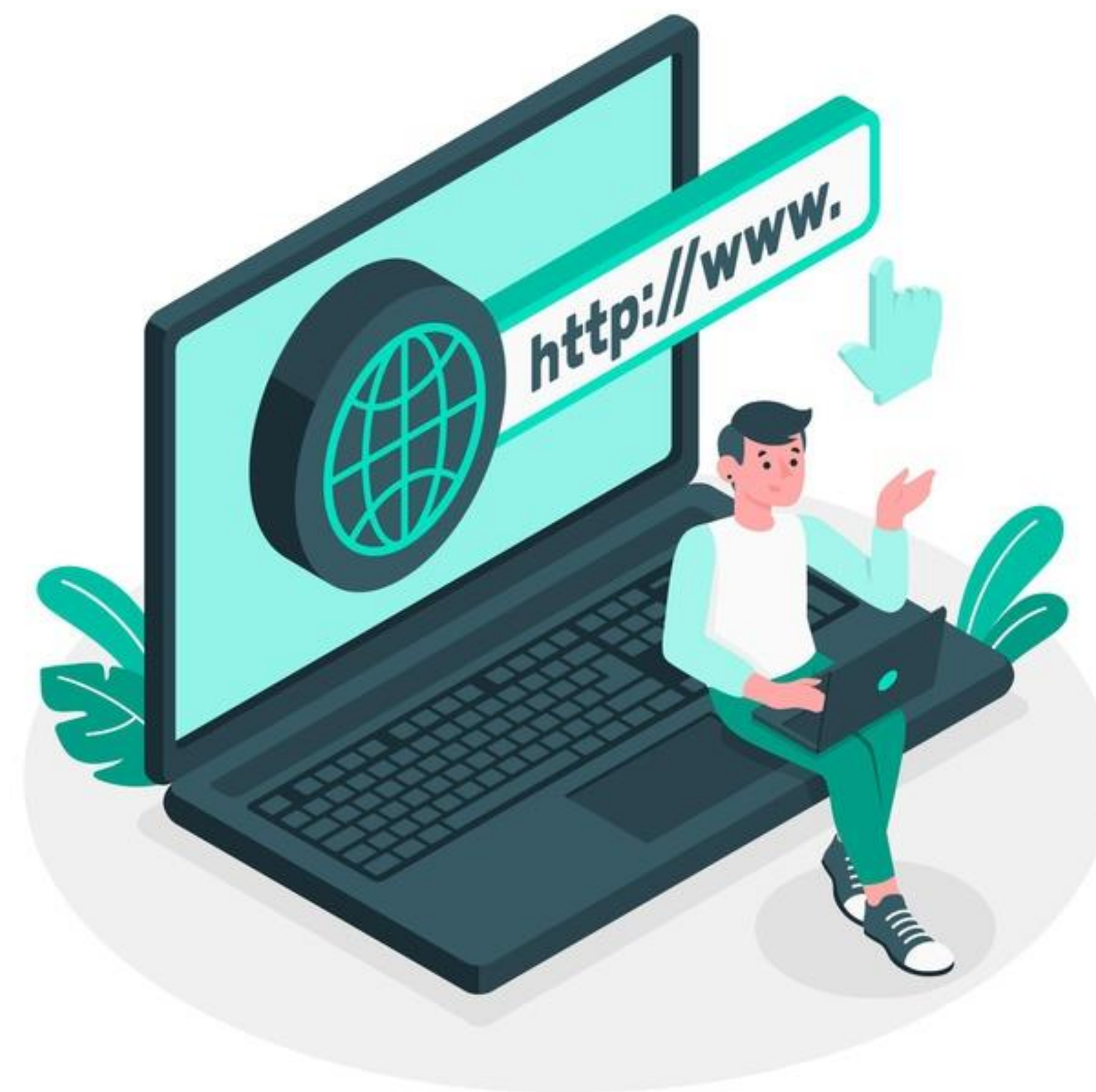




el servidor procesa esa petición y enviar los datos de la página web que el cliente solicitó, una vez más por mediante el protocolo **HTTP**

al llegar la información del servidor nuestro navegador(**el cliente**) toma esa información y realiza la representación de la página web.

En otras palabras, el servidor lo que hace es enviar las instrucciones para que nuestro navegador cree una copia de cómo se debería ver el sitio web que estamos visitando.







## Que es una URL



Para comprender mejor los protocolos HTTP, tenemos que revisar que es una URL.

La **URL (Uniform Resource Locator)** o **Localizador Uniforme de Recursos**, es la dirección específica que se le asigna a un servicio (servidor) disponible en la WEB para que este sea **localizado**, lo que quiere decir que hay una **URL** para cada uno de los recursos (páginas, sitios, documentos, archivos, carpetas) que hay en la World Wide Web.





Una **URL** como <https://www.youtube.com/> está compuesta por varias partes:

- '**https://**' que sería el protocolo de acceso para las páginas de internet. Otro ejemplo es el '**ftp://**' que es el protocolo para descarga de ficheros.
- '**www.**' que es la red donde se encuentra.
- '**youtube**' sería el nombre del dominio
- '**.com**' es el tipo de dominio. Estos pueden ser genéricos como .net, .org, .mobi o territoriales como .mx, .ar, .cl. Hoy en día el tipo de dominio se está diversificando pudiendo contener frases completas como .google o .maps.

**http://www.ejemplo.com**

Esquema                      Autoridad





Por lo tanto la sección de la **'www.example.com'** de la URL conforma el **dominio** que es un nombre que representa una dirección **IP** en la **WEB** y la parte **'https://'** representa el **método o protocolo** que se emplea para la transferencia de los datos.

**http://www.ejemplo.com**

**Protocolo**

**Dominio**



Como ya sabemos las generalidades de una página web y la manera como se conecta con internet, vamos a revisar cómo podemos desarrollar o implementar una página web.

Existen dos lenguajes principales que nos ayudan a hacer estos procesos:

- **HTML**
- **CSS**

El lenguaje HTML nos permite generar la base de la página web. Donde se encuentran los títulos, listas o menús para seleccionar una opción específica, y la estructura general de la página.

En otras palabras el código HTML es la estructura base o el esqueleto de nuestra página web.







Un editor que puede ser muy útil en el proceso, es **Visual studio**. Es gratuito y fácil de manejar. Sin embargo es posible programar en **HTML** en cualquier tipo de editor, no debe ser especial.

Para acceder al editor, debemos buscar en Google: **visual studio code**. Y hacemos click en el primer enlace que nos aparece y con eso ingresamos a la página inicial.



visual studio code



Todos

Imágenes

Videos

Noticias

Libros

Más

Herramientas

Cerca de 470,000,000 resultados (0.64 segundos)

<https://code.visualstudio.com> Traducir esta página

**Visual Studio Code - Code Editing. Redefined**

**Visual Studio Code** is a code editor redefined and optimized for building and debugging modern web and cloud applications. **Visual Studio Code** is free and ...

Download

Visual Studio Code is free and available on your favorite ...

Updates

April 2021 - March 2021 - January 2021 - February 2021 - ...





Ahora podemos **descargar** el programa desde la página inicial, haciendo **click** en cualquiera de las dos opciones que nos aparecen disponibles.



The screenshot shows the Visual Studio Code website. At the top, there's a navigation bar with links: Visual Studio Code, Docs, Updates, Blog, API, Extensions, FAQ, Learn, and a Search Docs button. A 'Download' button with a download icon is highlighted with a green border. Below the navigation bar, a message states 'Version 1.57 is now available! Read about the new features and fixes from May.' The main content area features the text 'Code editing. Redefined.' and 'Free. Built on open source. Runs everywhere.' A large blue button labeled 'Download for Windows' with 'Stable Build' underneath is highlighted with a green border. Below this button, there's a link for 'Other platforms and Insiders Edition'. At the bottom, a disclaimer states 'By using VS Code, you agree to its license and privacy statement.' On the right side of the main content, there's a preview of the Visual Studio Code interface. The 'EXTENSIONS: MARKETPLACE' sidebar is open, showing a list of extensions with their icons, names, versions, and download counts. The main editor area shows a JavaScript file named 'serviceWorker.js' with code for registering a service worker. A dropdown menu is open over the 'navigator.serviceWorker' property, showing a list of available methods and properties like 'product', 'productSub', 'removeSiteSpecificTrackingException', etc.



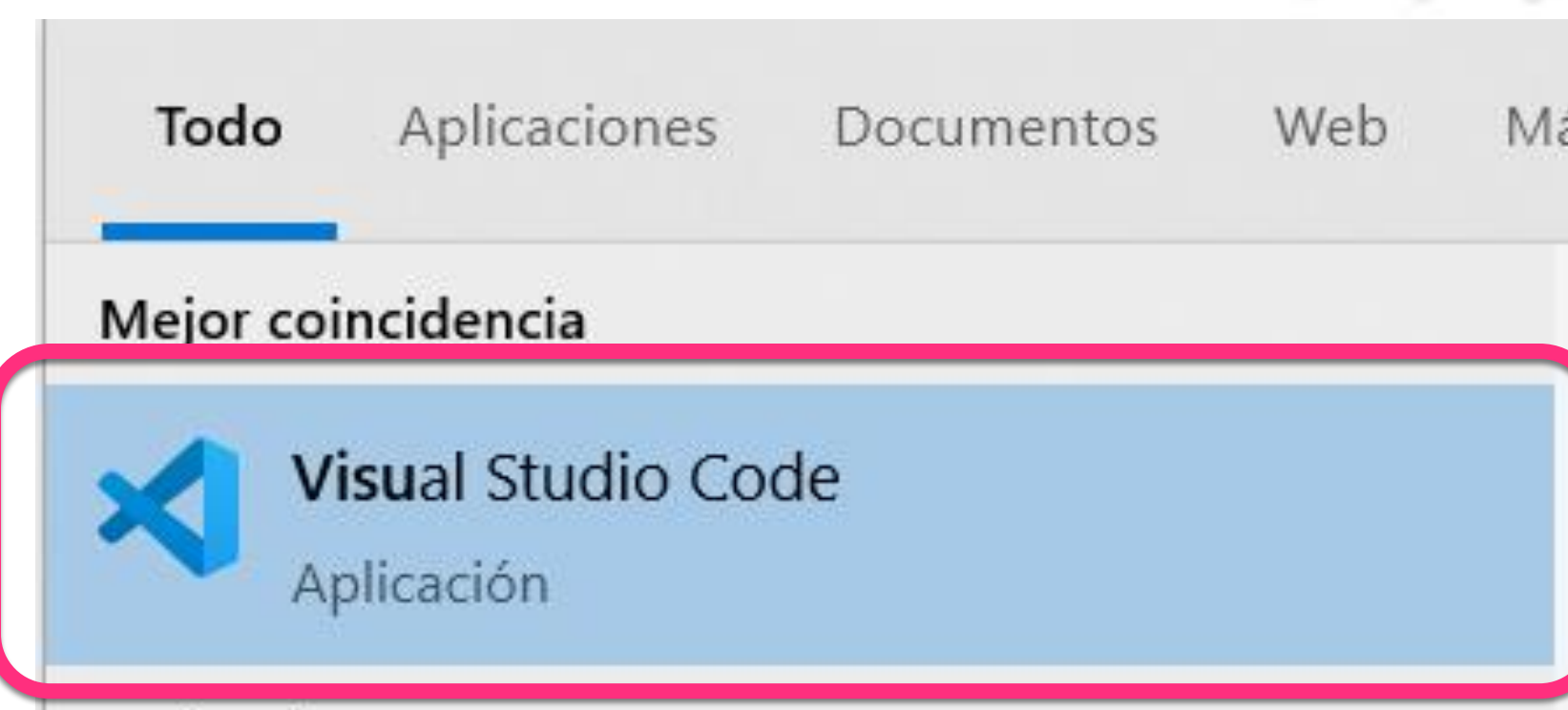


Al hacer **click** iniciará inmediatamente la descarga y tendremos que seguir las instrucciones para instalarlo. Solo debemos dar siguiente y seleccionar una opción para permitir abrir los archivos con **visual studio**.



VSCoDeUserSetup....exe

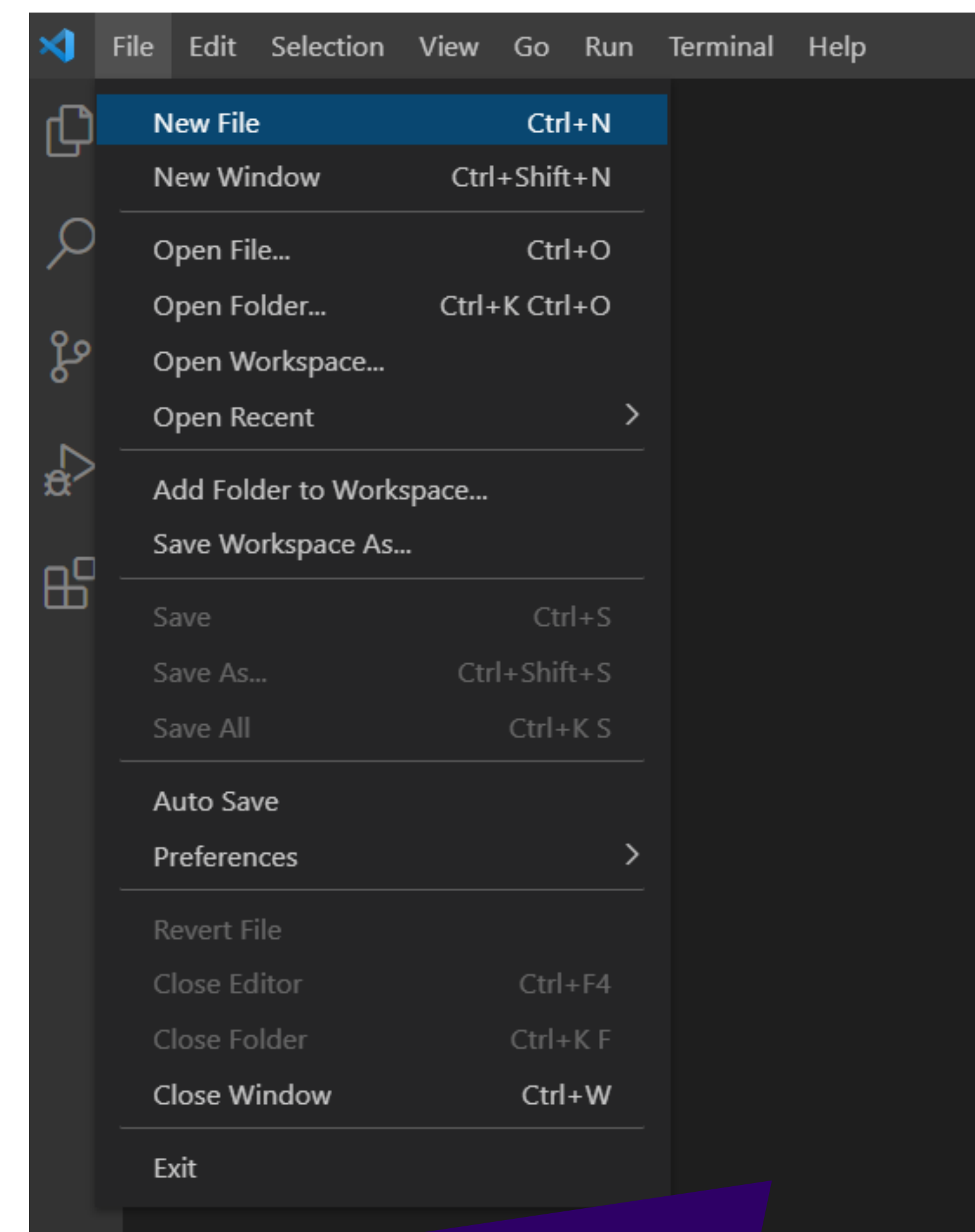
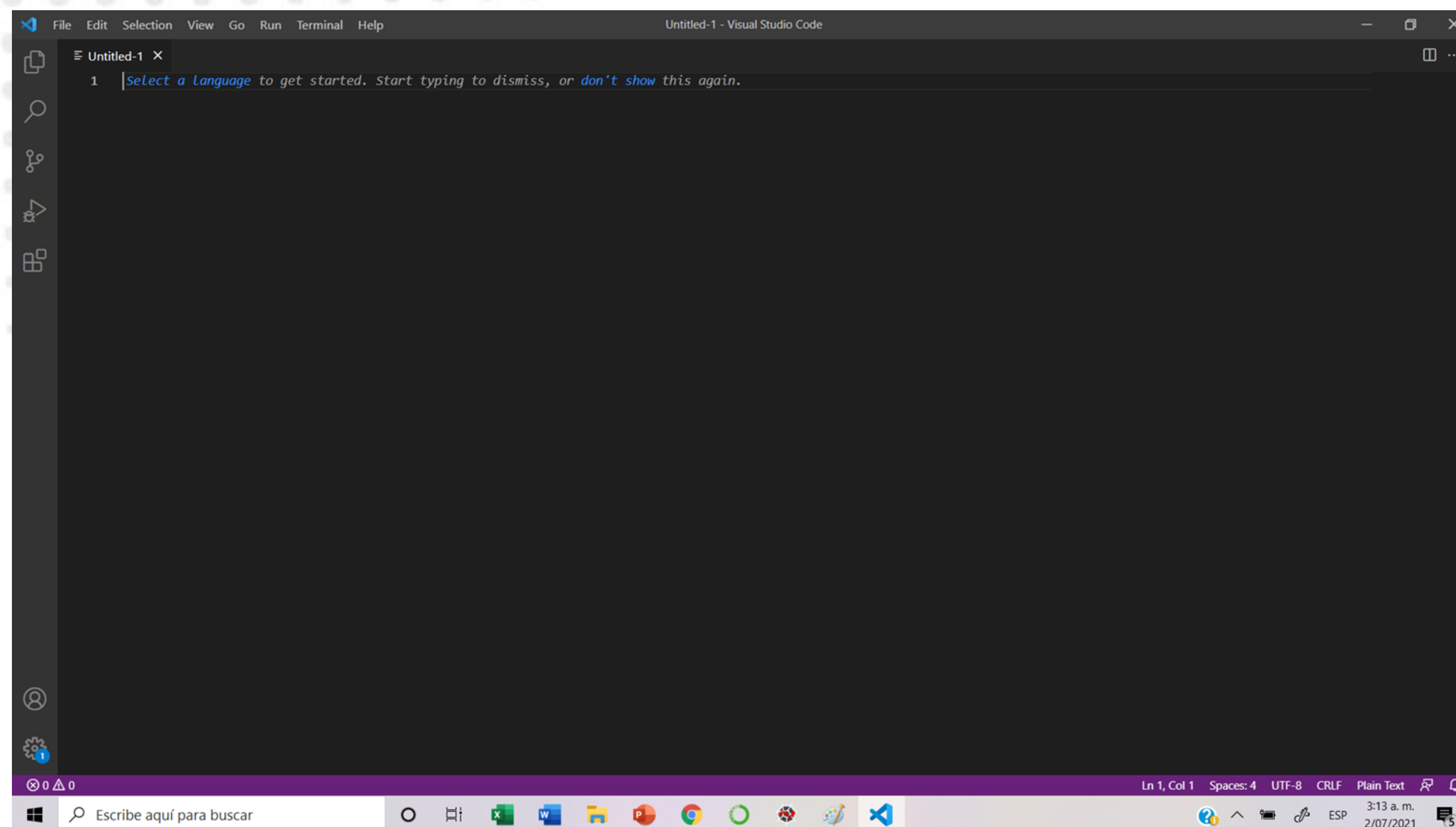
7,2/76,2 MB, Quedan 28 s



Una vez instalado solo lo buscamos en el buscador de nuestro computador y hacemos click sobre la aplicación para ejecutarla.

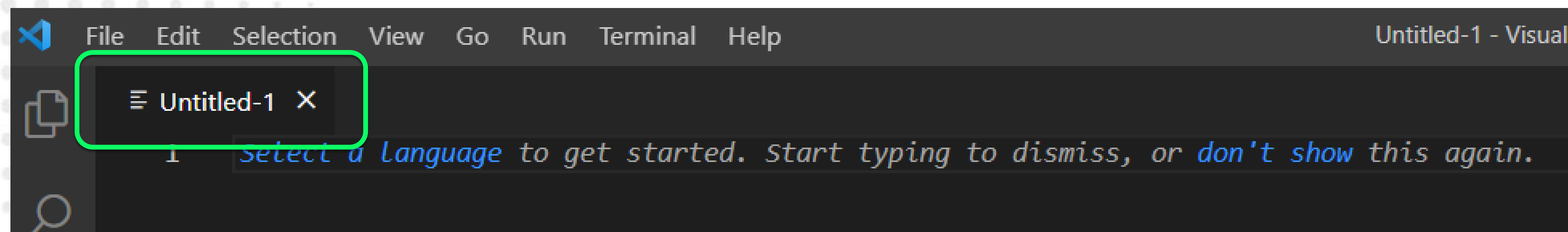


Cuando abrimos el **visual studio code**, nos aparece una ventana principal donde podemos comenzar a escribir nuestro código.



Ahora hacemos **click** en **File** (carpeta) y seleccionamos la primera opción para crear nuestro primer archivo..





Ahora tenemos un archivo nuevo donde podemos comenzar a crear nuestra estructura de página web. Aquí también podemos crear el código de **CSS**.

En la próxima clase vamos a revisar cómo comenzar el proceso de construcción de una página web.





El futuro digital  
es de todos

MinTIC

**GRACIAS**

**OPERADO POR:**

