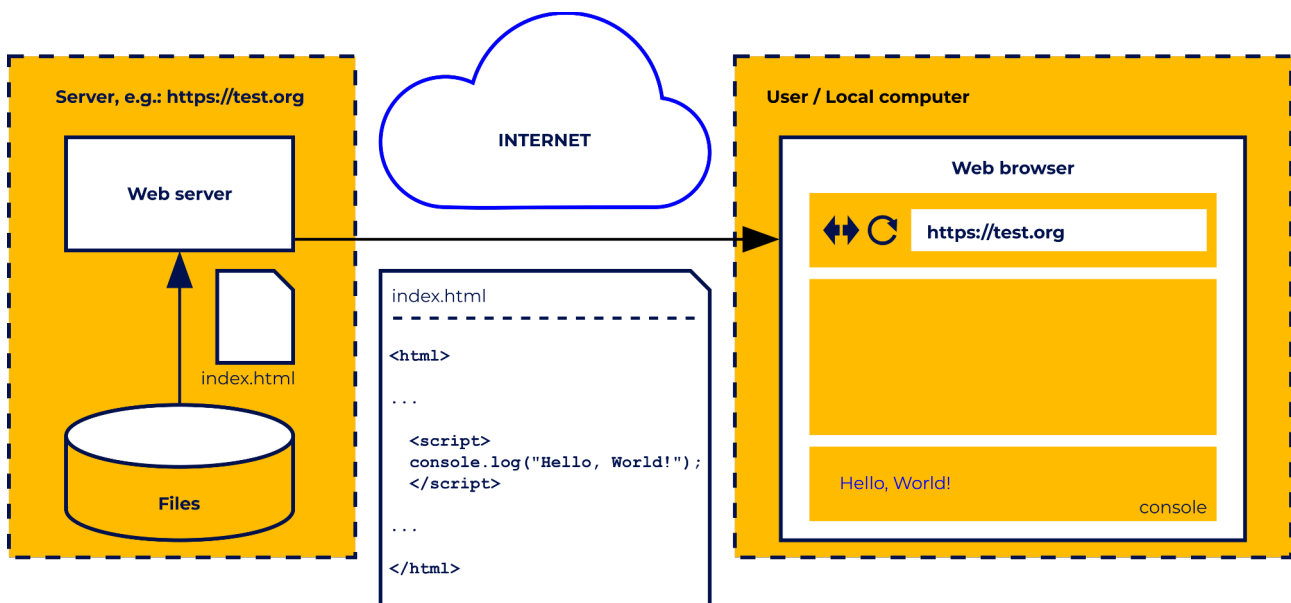




## ¿Cómo podemos ejecutar nuestro código JavaScript?

Comencemos con un ejemplo simple, donde el navegador obtiene una página simple (tal vez incluso vacía) de <https://test.org>. (La dirección está creada para este ejemplo, así que no intentes ingresarla. Mire la figura a continuación).



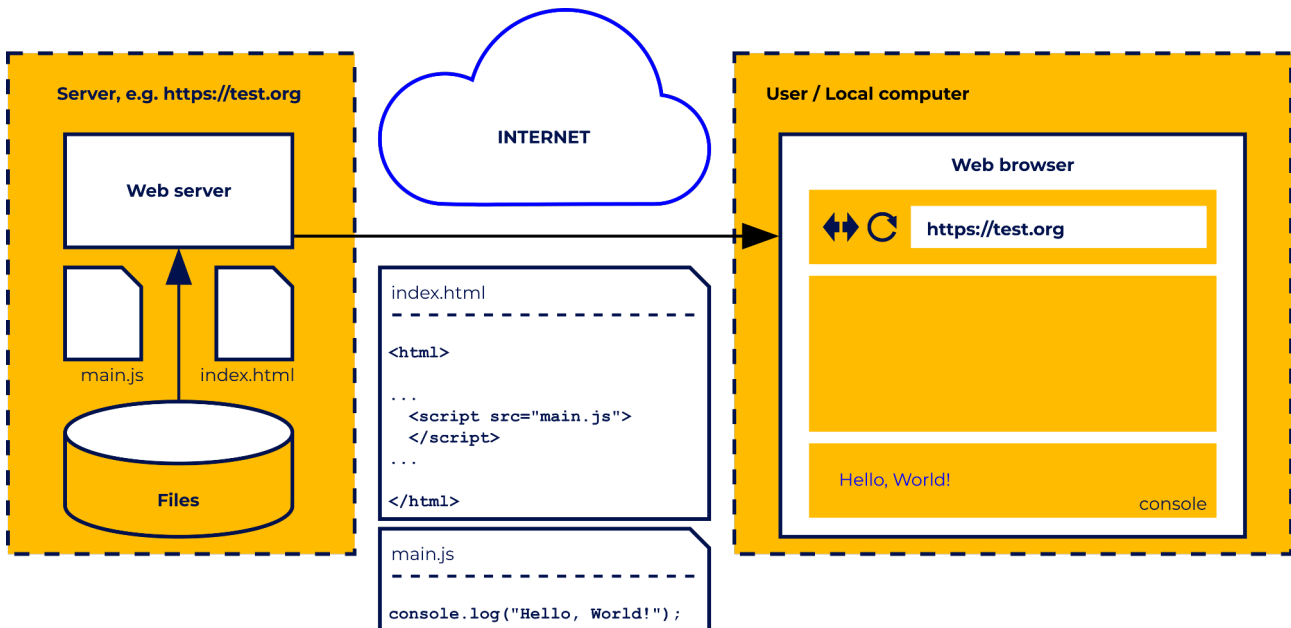
Empecemos por el lado derecho de la figura. El **usuario ejecuta un navegador web en su computadora (por ejemplo, Chrome)**. Usando el atajo de tecla apropiado, **activan las herramientas de desarrollador para poder usar la consola**. Recuerde que estas herramientas no son necesarias para el uso normal del navegador y, por lo tanto, están ocultas de forma predeterminada. Luego, el usuario escribe <https://test.org> (la URL de nuestro sitio falso) en la barra de direcciones.

En el servidor remoto (lado izquierdo del dibujo), asociado a la dirección <https://test.org>, se lanza un servidor web que, tras recibir una solicitud de nuestro usuario, le preparará una respuesta. En el caso más sencillo, la respuesta sólo contendrá un archivo html, que podrá almacenarse en el mismo servidor. **El archivo html (en este ejemplo, index.html) se devuelve al usuario y el navegador lo procesa. Si se define algún contenido (por ejemplo, un párrafo con texto), se mostrará en la ventana del navegador.**

Sin embargo, estamos más interesados en el hecho de que el archivo `index.html` contiene el `<script>` y `</script>` etiquetas, con un fragmento de código JavaScript entre



ellas. ¿Lo reconoces? Obviamente, este es un intento de mostrar nuestro "¡Hola mundo!" en la consola. Al cargar la página, el código colocado dentro del `<script>` Las etiquetas deben ejecutarse y, si las herramientas de desarrollador están habilitadas y el panel de la consola está visible, la consola mostrará "¡Hola Mundo!".



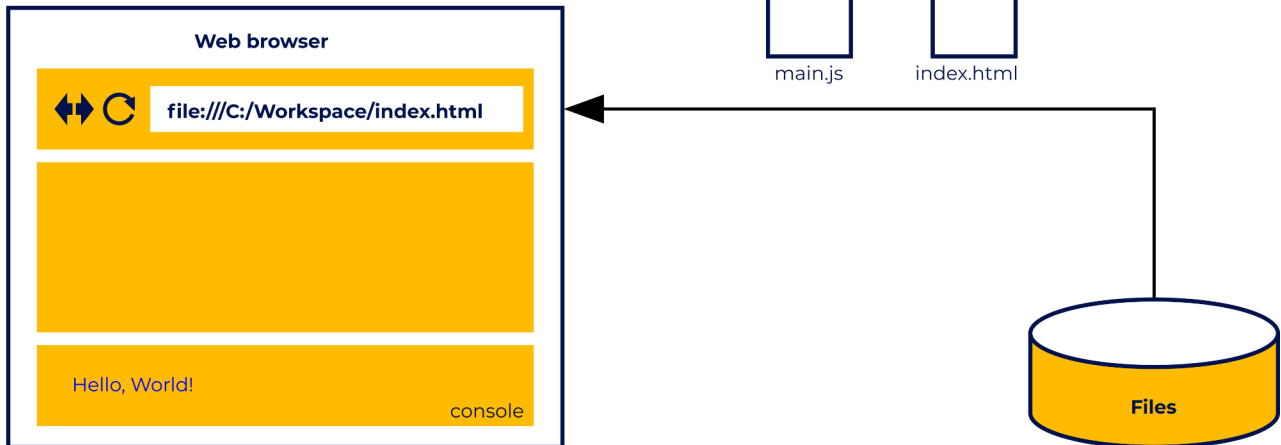
Todo funcionará exactamente igual que en el escenario anterior, excepto que el servidor web proporcionará el archivo main.js además de index.html. El usuario no notará ninguna diferencia. Por supuesto, colocar nuestro código en un servidor remoto sólo para probarlo sería un poco engorroso.

Tenemos otra posibilidad, que es que podemos cargar un archivo html local (es decir, uno que esté en nuestra computadora) en el navegador. Si este código contiene un `<script>` etiqueta que indica algún archivo JavaScript, este archivo también se cargará desde los recursos locales.

La imagen muestra un escenario simple en el que el usuario carga un archivo index.html local en el navegador, en el que hay una referencia a main.js (por lo que este archivo también se cargará automáticamente).



User / Local computer



Puede cargar un archivo html local escribiendo su ruta local después de file:/// en la barra de direcciones o simplemente abriéndolo en su navegador usando el comando Abrir del menú. Dado que el menú de los navegadores suele estar oculto, una forma más sencilla puede ser utilizar un acceso directo para abrir documentos existentes en las aplicaciones. El atajo es universal, no sólo para navegadores y probablemente ya lo hayas visto:



o en el caso de macOS:

