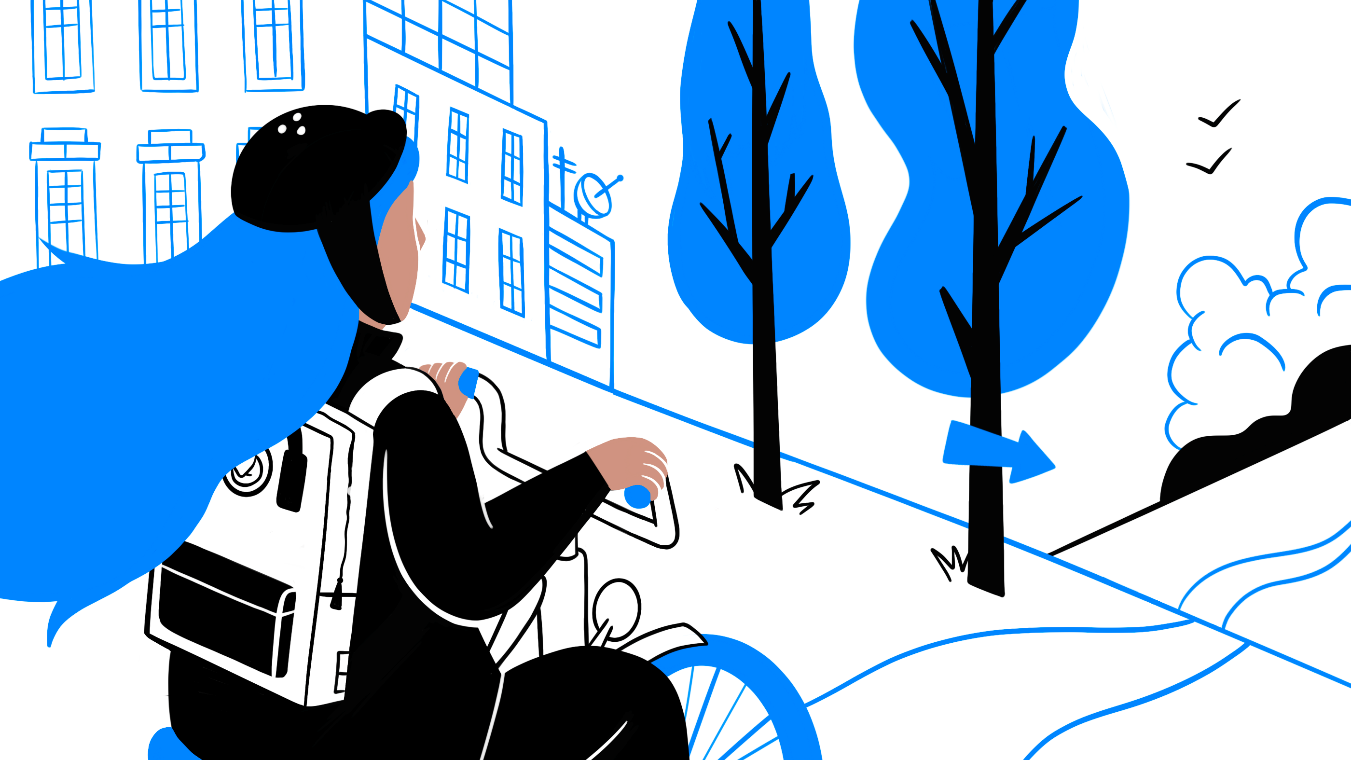
**Prueba final 3**

Es hora de comprobar lo que has aprendido en el Sprint 3. Puedes volver a este cuestionario si te atascas en el Proyecto 3, ya que hemos agregado algunas sugerencias útiles para resolver algunos de los problemas comunes del proyecto.

Este cuestionario contiene 8 preguntas y te tomará alrededor de 20 minutos completarla.

Vamos a dar una vuelta a la manzana. Estarás en tu elemento.

**Git, Línea de comandos y Estructura de archivos (5 preguntas)**

Supón que hay un nuevo repositorio en GitHub que deseas clonar, la URL del repositorio es git@github.com:fake-user/fake-repo.git.

Pregunta

1. Escribe un comando para clonar este repositorio en tu computadora usando su URL

git clone git@github.com:fake-user/fake-repo.git

¡Correcto!

¡Bien hecho!

Pregunta

2. Bien, ahora tienes el repositorio clonado en tu computadora y te gustaría comenzar a realizar cambios en él. Primero, vamos a crear un nuevo directorio llamado **new-dir** dentro del repositorio local. ¿Qué comando(s) funcionará(n)?

Elige tantas como quieras

mkdir new-dir

mkdir fake-repo/new-dir

¡Correcto!

cd fake-repo && mkdir new-dir

¡Correcto!

¡Perfecto!

Pregunta

3. Ahora estás dentro del repositorio local y has creado el directorio llamado **new-dir**.

fake-repo/ <- estás aquí

new-dir/

¿Cuál de los siguientes comandos creará un archivo llamado **README.md** en la raíz del repositorio? Elige todas las respuestas correctas.

Elige tantas como quieras

touch README.md

Correcto. Esto creará un archivo vacío llamado README.md en el directorio de trabajo actual. El ./ antes del nombre del archivo está implícito.

touch ./README.md

Correcto. Esto creará un archivo vacío llamado README.md en el directorio de trabajo actual. El ./ antes del nombre del archivo se refiere al directorio de trabajo actual y es opcional.

touch /README.md

touch ~/README.md

¡Lo has entendido bien!

Pregunta

4. Todavía estás en la raíz del repositorio local **fake-repo**.

fake-repo/ <- todavía estás aquí

new-dir/

README.md

Escribe un comando para crear un archivo llamado **new-file** dentro de **new-dir**. Sólo utiliza el comando **touch**, no cambies los directorios.

touch ./new-dir/new-file

¡Correcto!

¡Buen trabajo!

Pregunta

5. Imagina que tienes un repositorio remoto [https://github.com/me/repo](https://github.com/username/test-repo) en GitHub. y quieres descargarlo en tu computadora, trabajar en el código y luego subir los cambios a GitHub.

Coloca los comandos Git en el orden correcto.

1

git clone https://github.com/me/repo.git

¡Tienes razón! Este comando descargará el código del repositorio a tu computadora.

2

git add -A

¡Tienes razón! Este comando pondrá en escena todos los archivos actualizados para un commit.

3

git commit -m "add semantic tags to index.html"

¡Tienes razón! Este comando confirmará el cambio localmente.

4

git push -u origin

¡Tienes razón! Este comando subirá el código actualizado a la rama main de tu repositorio de GitHub.

¡Perfecto!

Posicionamiento (3 preguntas)

Pregunta

6. Relaciona las siguientes propiedades de **position** con su correspondiente definición.

El elemento está posicionado según el flujo normal del documento, pero se puede ajustar su posición con top, right, bottom y left.

relative

Por defecto, un elemento posicionado relativamente se comporta igual que un elemento con position: static, pero puede ajustarse su posición.

Se elimina el elemento del flujo normal del documento y se posiciona con relación a su ancestro más cercano con una position. El elemento puede volver a posicionarse con top, right, bottom y left.

absolute

Si el padre del elemento que deseas posicionar absolutamente no está posicionado de otra manera, siempre puedes aplicarle position: relative. Esto permitirá que sus hijos estén posicionados absolutamente sin efectos secundarios no deseados.

El elemento está posicionado según el flujo normal del documento, y no puede reposicionarse con top, right, bottom y left.

static

Este es el valor predeterminado de position.

El elemento es eliminado del flujo normal del documento y se posiciona con relación a la ventana del navegador. El elemento puede volver a posicionarse con top, right, bottom y left.

fixed

Esto resulta útil para elementos que siempre deseas que aparezcan en la misma posición en la página. Entre los ejemplos están las ventanas emergentes y las barras de navegación.

¡Perfecto!

Pregunta

7. Selecciona las afirmaciones verdaderas sobre el posicionamiento absoluto.

Elige tantas como quieras

Este elemento se posiciona de acuerdo con el padre más cercano con posicionamiento relativo.

¡Esta es una elección correcta! Un elemento con posicionamiento absoluto se coloca según el antecesor más cercano con cualquier valor de posicionamiento, excepto static.

El elemento se posiciona según el padre más cercano con posicionamiento estático.

Su uso excesivo puede limitar la flexibilidad de tu página.

¡Esta es la correcta! Los elementos con posicionamiento absoluto funcionan, en su mayoría, de forma independiente a los demás elementos de tu página web, por lo que tendrás que estar atento a ello cuando realices cambios en el diseño.

Esto hace que los elementos permanezcan en la misma posición (incluso mientras se desplaza la página) en la ventana del navegador.

¡Lo has entendido bien!

Pregunta

8. Qué tipo de elementos pueden posicionarse con la propiedad **z-index**?

Todos los elementos.

Sólo los elementos con una propiedad position predeterminada (es decir, position: static).

Sólo los elementos con una position no predeterminada (es decir, nada a excepción de position: static).

Para afectar un elemento con z-index, los elementos deben tener una propiedad position no predeterminada.

Sólo los elementos con position: relative.

¡Buen trabajo!

**¡Gran trabajo! ¡Ahora ya puedes continuar con el proyecto!**

No te preocupes si no lo hiciste todo bien. Puedes repasar tus conocimientos consultando el resumen del Sprint.