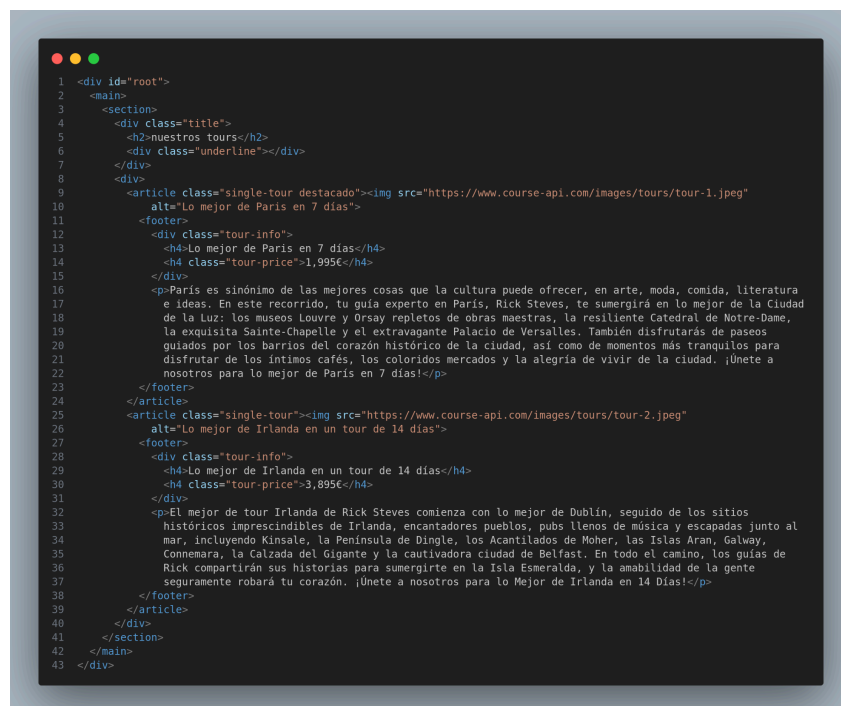
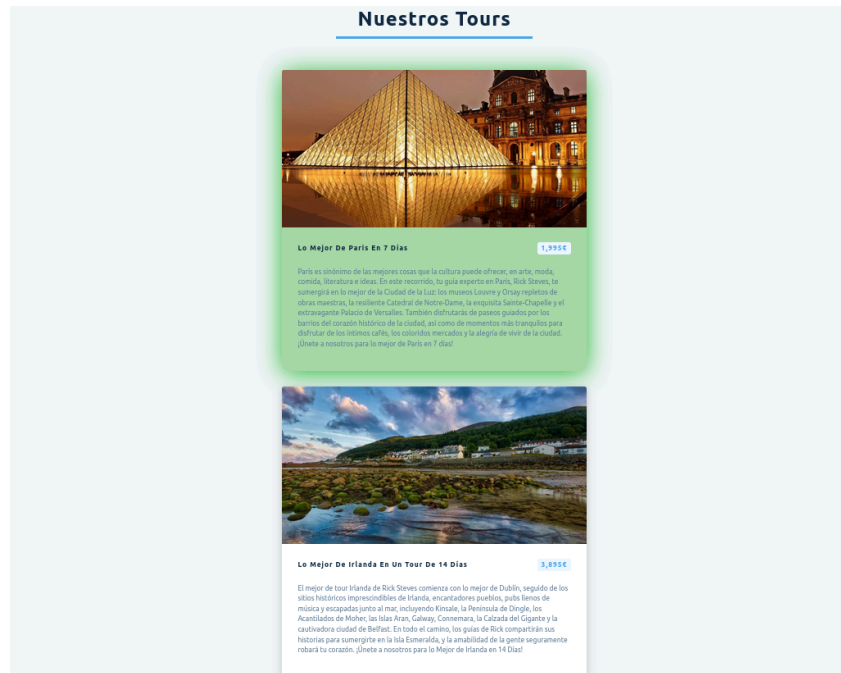


DWEC05 - Tarea 02

ENUNCIADO DE LA TAREA:

En la plantilla que se os facilita aparece una un documento HTML de ejemplo (después deberás borrar o comentar las líneas que aparecen) para que veais que elementos teneis que crear y con qué clase para que quede correctamente estilado.



Se trata de crear 2 componentes personalizados con React y visualizarlos en la página pero vamos a tener una serie de condiciones a la hora de renderizar:

- Primer componente: consistirá en un elemento `<main>` que albergará un elemento `<section>`.
Dentro de éste habrá un elemento `<div>` que contendrá un elemento `<h2>` y un `<div>` con clase `“underline”` para dibujar la línea en el título.
- Segundo componente: consistirá en un `<article>` que tendrá dentro un elemento `` y un `<footer>`. Éste último tendrá un `<div>` para la información de los tours, que irán en un `<h4>` para el nombre del tour, otro `<h4>` para el precio, y a la misma altura del div habrá un `<p>` para la información descriptiva del tour.

Es algo lioso entenderlo así, por eso os he puesto también una foto de como está organizado el HTML (que lo teneis en la plantilla que os adjunto)

La información de los tours la tenéis adjunta en un fichero que se llama **data.js**. En este fichero podréis editar la información (no añadir campos, solo modificar para pruebas) para ver que las condiciones de renderizados se aplican correctamente:

1. Cuando algún tour tenga el campo **destacado** en `true`, se deberá añadir la clase `“destacado”` al **article** que hará que se resalte solo (los estilos están ya definidos).
2. Cuando algún tour no contenga el **src** de la imagen, deberá renderizar una imagen representativa (está en el fichero de datos comentado).
3. Cuando algún tour no contenga el **precio** del mismo, tendrá que aparecer `“A consultar”` en el elemento que muestra el precio en la página web.
4. En el caso de que algún tour no tenga **información** descriptiva del mismo, no renderizaremos ese tour en concreto.

Una vez terminado el ejercicio, cambiando la información del fichero, comprobareis que los tours se renderizan correctamente según las condiciones que le hemos puesto.

ACLARACIONES EL ENUNCIADO:

- Cuando os descarguéis la plantilla, ejecutar `“npm i”` para instalar las dependencias del proyecto.
- Podéis utilizar y guiaros con la estructura de ficheros e *imports* que utiliza un proyecto de vite de prueba.
- Los nombres de los componentes los decidís vosotros mismos.

ACLARACIONES SOBRE LA ENTREGA DE LA TAREA:

- Para la entrega de esta tarea tendréis que subir el proyecto a alguna plataforma (github, gitlab, ...) para que yo pueda descargarla ya que por tema del tamaño del proyecto no podremos subirlo a la plataforma moodle. Nombrar al repositorio con la nomenclatura habitual.

Apellido1_Apellido2_Nombre_DWEC04_Tarea02

- Utilizar los comentarios de código para realizar breves explicaciones. En caso de necesitar/querer explicar más en detalle el ejercicio utilizar la [plantilla](#) e ir adjuntando capturas de pantalla del código y las explicaciones y aclaraciones necesarias .
- Criterios de evaluación relacionados con la tarea:

RA-3	CE	a) Se han identificado los objetos predefinidos del lenguaje.
	CE	c) Se han escrito sentencias que utilicen los objetos predefinidos del lenguaje para cambiar el aspecto del navegador y el documento que contiene.
	CE	d) Se han generado textos y etiquetas como resultado de la ejecución de código en el navegador.
	CE	e) Se han escrito sentencias que utilicen los objetos predefinidos del lenguaje para interactuar con el usuario.
	CE	h) Se ha depurado y documentado el código.
RA-4	CE	a) Se han clasificado y utilizado las funciones predefinidas del lenguaje.
	CE	b) Se han creado y utilizado funciones definidas por el usuario.
	CE	c) Se han reconocido las características del lenguaje relativas a la creación y uso de arrays.
	CE	d) Se han creado y utilizado arrays.
	CE	e) Se han reconocido las características de orientación a objetos del lenguaje.
	CE	f) Se ha creado código para definir la estructura de objetos.
	CE	g) Se han creado métodos y propiedades.
	CE	i) Se ha depurado y documentado el código.