

<b>Nombre:</b>	<b>Fecha:</b>	<b>Nota</b>
<b>Apellidos:</b>	<b>Curso:</b>	

1) Crea un formulario con los siguientes campos:

- Identificador (formado por dos letras en mayúscula, un símbolo y cuatro dígitos (ej. DD#8908))
- Nombre (campo de texto 50 caracteres)
- Correo electrónico: en formato varios caracteres (letras, números, puntos o guiones altos o bajos, 20 máximo) seguido de arroba, letras (20 máximo), punto (1), y 2 o 3 letras.
- Edad (campo numérico y mayor a 18)
- Teléfono (campo de texto, con formato + y dos o tres cifras seguido de guion y 9 cifras)

Añade la siguiente lógica de validación en JavaScript:

- Valida que todos los campos sean requeridos, no estén vacíos y cumplan el formato requerido.
- Muestra un mensaje de error si algún campo no cumple con las condiciones de validación y otro distinto en caso de que esté vacío a la derecha del campo que es incorrecto.
- Si todos los campos son válidos, envía los datos al servidor (simula este proceso con una alerta que muestre los datos).

Eventos:

- Asocia el evento onfocus al identificador para que al poner el foco se vacíe dicho elemento.
- Asocia el evento onblur al identificador para que al perder el foco compruebe que cumple con la expresión regular y aparezca a la derecha un tick verde o rojo mostrando si es correcto o no.

**(4 puntos)**

2) Crea una aplicación que consuma la API de Rick and Morty.

Documentación: <https://rickandmortyapi.com/documentation>

**Endpoints de la API de Rick and Morty:**

1. Personajes (Characters):
  - Obtener todos los personajes: <https://rickandmortyapi.com/api/character/>
  - Obtener un personaje específico: <https://rickandmortyapi.com/api/character/{id}/>
2. Episodios (Episodes):
  - Obtener todos los episodios: <https://rickandmortyapi.com/api/episode/>
  - Obtener un episodio específico: <https://rickandmortyapi.com/api/episode/{id}/>
3. Ubicaciones (Locations):
  - Obtener todas las ubicaciones: <https://rickandmortyapi.com/api/location/>
  - Obtener una ubicación específica: <https://rickandmortyapi.com/api/location/{id}/>

**Instrucciones:**

1. La aplicación constará de un formulario que incluirá dos campos:
  - a) Tipo: Indicará uno de los tipos de datos que puede proporcionar la API (character, episode, location), realiza los checks necesarios para que el usuario no introduzca información incorrecta y que coincida con los endpoints de la API. Obligatorio.
  - b) ID: Id a buscar del tipo introducido anteriormente. Estará compuesto de un número de una o tres cifras. Debes mostrar a la derecha del campo un error en caso de que esté vacío o que no coincida con la expresión regular indicada. No obligatorio.
2. Al enviar el formulario, realizarás dos tipos de llamadas a la API. En caso de que no exista ID, muestra la información del tipo de endpoint introducido con el número de elementos proporcionados por defecto por la API. En caso de que exista el ID, muestra la información del tipo de endpoint e id introducido por el usuario
3. Utiliza la Fetch API para realizar la llamada GET a la API de Rick and Morty (<https://rickandmortyapi.com/api/{tipo}/>), donde {tipo} es el tipo de endpoint que quiere consultar el usuario y (<https://rickandmortyapi.com/api/{tipo}/{id}/>), {id} es el ID del tipo ingresado por el usuario.
4. Muestra la siguiente información para cada tipo de endpoint:
  - a) Character: Image, id, name, species, origin.name.
  - b) Episode: Id, name, air\_date.
  - c) Location: Id, name, type.
5. En caso de que el usuario no introduzca el ID, muestra todos los tipos de datos existentes para el endpoint elegido, no sólo los recibidos por defecto.
6. Añade el efecto hover que quieras a todos los elementos mostrados.
7. Maneja posibles errores, como respuestas no exitosas de la API o problemas de conexión.
8. Diseña la interfaz de manera limpia y legible, utilizando CSS para mejorar la presentación en caso de que tengas tiempo.
9. Añade comentarios en tu código para explicar tu enfoque y decisiones.

**(6 puntos)****Criterios de corrección:****Ejercicio 1:**

- 1 punto – Crea y comprueba que cada campo cumpla con su expresión regular para validarlo de forma correcta.
- 1 punto - Muestra un mensaje a la derecha del campo que está vacío o que no cumple con la expresión regular.
- 0,75 puntos – Realiza el evento onfocus vaciando el campo del identificador.
- 0,75 puntos – Realiza el evento onblur comprobando que cumple con la expresión regular del identificador.

0.5 puntos – Envía el formulario comprobando de forma correcta que todos los campos están validados.

**Ejercicio 2:**

1 punto – Realiza dos tipos de llamada a la api, en función del endpoint elegido por el usuario y si introduce o no el ID.

1 punto – Conecta con la api de manera correcta recibiendo el número de parámetros por defecto.

1,5 puntos – Muestra la información con todas las propiedades definidas para cada tipo de endpoint de manera adecuada en la web.

1 punto – Muestra todos los tipos de datos existentes para el endpoint elegido, no sólo los que recibe por defecto.

1 punto – Añadir efecto hover a los elementos mostrados.

0.5 puntos – Maneja todos los errores y comenta el código adecuadamente.