

## INFORMACIÓN DE LA PLACA DE UN AUTO

1. Dentro de su directorio **krakeDev**, crear una carpeta llamada **placas**. Dentro de la carpeta crear los archivos `placas.html`, `placas.js`, `serviciosPlacas.js`, `utilitariosCadenas.js` y `placas.css`. Agregar el archivo `utilitarios.js`.
2. En este ejercicio obtendremos la información que existe en las placas de un vehículo, basados en la información que nos proporciona el siguiente link:  
<https://www.conmicelu.com/tramites/placas-de-ecuador-por-provincias/>
3. En el archivo `utilitariosCadenas.js` crear las siguientes funciones

Función	<b>esMayuscula</b>
Descripción	Verifica si el caracter es una letra Mayúscula. Algoritmo: Obtiene el código Ascii del caracter Valida si el código corresponde a las letras mayúsculas de acuerdo a la tabla Ascii
Parámetros	<b>caracter</b> : el caracter a validarse
Retorno	true si el caracter es una letra mayúscula, caso contrario retorna false

Función	<b>esDigito</b>
Descripción	Verifica si el caracter es un dígito (número del 0 al 9). Algoritmo: Obtiene el código Ascii del caracter Valida si el código corresponde a un dígito de acuerdo a la tabla Ascii
Parámetros	<b>caracter</b> : el caracter a validarse
Retorno	true si el caracter es un dígito, caso contrario retorna false

Función	<b>esGuion</b>
Descripción	Verifica si el caracter es un guión. Algoritmo: Obtiene el código Ascii del caracter Valida si el código corresponde a un guión de acuerdo a la tabla Ascii
Parámetros	<b>caracter</b> : el caracter a validarse
Retorno	true si el caracter es un guión, caso contrario retorna false

4. En el archivo serviciosPlacas.js crear la siguiente función

Función	<b>validarEstructura</b>
Descripción	<p>Valida la estructura básica de una placa, en base a las siguientes reglas:</p> <ul style="list-style-type: none"> <li>• La longitud de la placa puede ser de 7 u 8 caracteres.</li> <li>• El primer caracter es una letra mayúscula.</li> <li>• El segundo caracter es una letra mayúscula.</li> <li>• El tercer caracter es una letra mayúscula.</li> <li>• El cuarto caracter es un guión.</li> <li>• El quinto caracter es un dígito.</li> <li>• El sexto caracter es un dígito.</li> <li>• El séptimo caracter es un dígito.</li> </ul> <p>IMPORTANTE</p> <ul style="list-style-type: none"> <li>• El octavo caracter es un dígito. NO TODAS LAS placas tienen octavo caracter. Antes de intentar acceder al caracter octavo, verificar la longitud de la placa.</li> </ul>
Parámetros	<b>placa:</b> la placa a validarse
Retorno	<p>En caso de que exista algún error con la placa, retorna un String con la descripción del error. Por ejemplo: La placa debe tener 7 u 8 caracteres. Recomendación: Concatenar los errores, de manera que se pueda retornar todos los errores. Por ejemplo: El cuarto caracter debe ser un guión, el sexto caracter debe ser un dígito.</p> <p>Si no existe ningún error, retornar null</p>

5. En el archivo placas.js crear la siguiente función

Función	<b>validarPlaca</b>
Descripción	<ol style="list-style-type: none"> <li>1. Obtener el valor ingresado por el usuario en la caja de texto</li> <li>2. Invocar a validarEstructura y guardar el retorno en una variable erroresEstructura.</li> <li>3. Si erroresEstructura es null, significa que no hubo errores, por lo tanto mostrar un mensaje ESTRUCTURA VALIDA. Caso contrario, mostrar en pantalla un mensaje ESTRUCTURA INCORRECTA y además mostrar los errores que retornó la función validarEstructura.</li> </ol>
Parámetros	No recibe nada
Retorno	No retorna nada

6. Crear la página html, con los componentes necesarios para probar la página al momento.

- Una label Ingrese la placa
- Una caja de texto para ingresar la placa
- Un botón VALIDAR

- Un label para mostrar si la placa es VALIDA o INCORRECTA
- Un label para mostrar los errores de validación

Agregar estilos a la página.

Probar el funcionamiento correcto hasta el momento, con la mayor cantidad de casos de prueba posibles. Realizar las capturas de los casos de prueba y pegar en un archivo de Word. Agregar este archivo al repositorio y subirlo.

7. En serviciosPlacas.js crear la siguiente función

Función	<b>obtenerProvincia</b>
Descripción	Obtiene la primera letra de la placa, determina la provincia a la que pertenece la placa
Parámetros	placa: La placa del vehículo
Retorno	Nombre de la provincia a la que pertenece, si no pertenece a ninguna provincia retorna null

8. En placas.js, dentro de la función validarPlaca, solamente si la estructura es correcta, invocar a obtenerProvincia, guardar el resultado en una variable. Si el valor obtenido no es null, mostrar la provincia en pantalla. Si es null, mostrar un mensaje provincia incorrecta.

9. En la página html, agregar los componentes necesarios para mostrar la información de provincia. Probar la mayor cantidad de casos. Agregar las capturas de los casos de prueba al documento de Word y subir todo al repositorio.


10. En serviciosPlacas.js crear la siguiente función

Función	<b>obtenerTipoVehiculo</b>
Descripción	Obtiene el segundo caracter de la placa y en base a lo indicado en la página web, indica el tipo de vehículo al que corresponde
Parámetros	placa: La placa del vehículo
Retorno	El tipo de vehículo correspondiente, si la letra no corresponde a ningún tipo, retorna null

11. En placas.js, dentro de la función validarPlaca, solamente si la estructura es correcta, invocar a obtenerTipoVehiculo, guardar el resultado en una variable. Si el valor obtenido no es null, mostrar el tipo en pantalla. Si es null, mostrar un mensaje tipo de vehículo incorrecto

12. Agregar los componentes necesarios en la página, probar, capturar los casos de prueba y subir al repositorio

13. En serviciosPlacas.js crear la siguiente función

Función	<b>obtenerDiaPicoYPlaca</b>
Descripción	<p>Obtiene el último caracter de la placa. Para obtener el último caracter, considere que la última posición de cualquier cadena es la longitud de la cadena menos uno.</p> <p>En base al último caracter obtenido y a la siguiente imagen</p>  <p>Determina qué día de la semana tiene pico y placa</p>
Parámetros	placa: La placa del vehículo
Retorno	El día que tiene pico y placa. Ejemplo: Viernes

14. En placas.js, dentro de la función validarPlaca, solamente si la estructura es correcta, invocar a obtenerDiaPicoYPlaca, guardar el resultado en una variable. Mostrar en pantalla el día que tiene pico y placa

15. Agregar los componentes necesarios en la página html, para mostrar la información. Realizar la mayor cantidad de casos de prueba, agregar capturas al documento y subirlo al repositorio.

16. Agregar un botón limpiar y una función limpiar, para limpiar la caja de texto y todos los textos de la página, realizar pruebas, agregar capturas al documento y subirlo.