


Universidad Tecnológica Nacional Facultad Regional Avellaneda 	
Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos	
Materia: LABORATORIO III	
Apellido:	Fecha:
Nombre:	Docente ⁽²⁾ : NEINER
División:	Nota ⁽²⁾ :
Legajo:	Firma ⁽²⁾ :
Instancia ⁽¹⁾ :	<div> <div>PP</div> <div>X</div> <div>RPP</div> <div></div> <div>SP</div> <div></div> <div>RSP</div> <div></div> <div>FIN</div> </div>

Crear una aplicación Web que permita realizar un ABM de distintos ciudadanos. El objetivo es lograr tener la siguiente funcionalidad:

- 1- (1pts) Crear la siguiente jerarquía de clases en **Typescript**:
 - a. **Persona**: nombre(string), apellido(string) y edad(entero) como atributos. Un constructor que reciba tres parámetros. Un método, *personaToString* que retorne la representación de la clase en formato cadena.
 - b. **Ciudadano**, hereda de persona, posee como atributos dni(entero), pais(string) y foto(string). Un constructor para inicializar los atributos (el atributo de la foto debe ser opcional). Un método *ciudadanoToJson* que retornará la representación del objeto en formato JSON. Se debe de reutilizar el método *personaToString*.
- 2- Crear la clase **Manejadora** que posea los siguientes métodos y funcionalidades:
- 3- (2pts) **agregarCiudadano**. Tomará los distintos valores desde la página index.html y guardará a ese ciudadano en el archivo **“./BACKEND/ciudadanos.json”**. La foto del ciudadano se guardará en **“./BACKEND/fotos”**. El archivo debe poder almacenar ‘n’ ciudadanos.
- 4- (2pts) **mostrarCiudadanos**. Recuperará a todos los ciudadanos del archivo .json y generará un listado dinámico (tabla) que mostrará toda la información (incluida la foto) de cada uno de los ciudadanos. Agregar columnas que permitan: **Eliminar** y **Modificar** al ciudadano elegido.
- 5- (1pts) **eliminarCiudadano**. Eliminará al ciudadano del archivo. Refrescar el listado para visualizar los cambios.
- 6- (2pts) **modificarCiudadano**. Mostrará todos los datos del ciudadano y permitirá modificar cualquier campo. Luego se guardarán los cambios en el archivo. Refrescar el listado para visualizar los cambios.
- 7- (1pts) El botón filtrar por país, invocará al método **filtrarPorPais** que retornará un listado completo de todos los ciudadanos según el país seleccionado.
- 8- (1pts) Generar, luego de la transpilación, solo un archivo .js (denominado app.js) que contenga todo el contenido necesario para interactuar con index.html.
- 9- (2pts) Agregar un checkbox a cada input para que el listado solo muestre las columnas que están seleccionadas.
- 10- (2pts) Generar la pre visualización de la foto del ciudadano antes de guardarla (tanto para el alta como para la modificación)

IMPORTANTE:

Se pueden bajar templates de internet o traer código hecho, pero en ningún caso se debe incluir código obsoleto o que no cumpla ninguna función dentro del parcial. Utilizar AJAX para cada acción realizada.

Se tiene que sumar 8 puntos para lograr un cuatro (4). Nueve puntos equivalen a un cinco (5), diez puntos equivalen a seis (6), once puntos: siete (7), doce puntos a un ocho (8), trece puntos equivalen a un nueve (9) y si se suman los catorce puntos la nota será de diez (10).