

Plataformas de Desarrollo

Ejercicio a resolver

Profesor: [Fernando Gonzalo Gaitán](#)

Comisión: *ACN4AV*

Correo Electrónico: fernando.gaitan@davinci.edu.ar

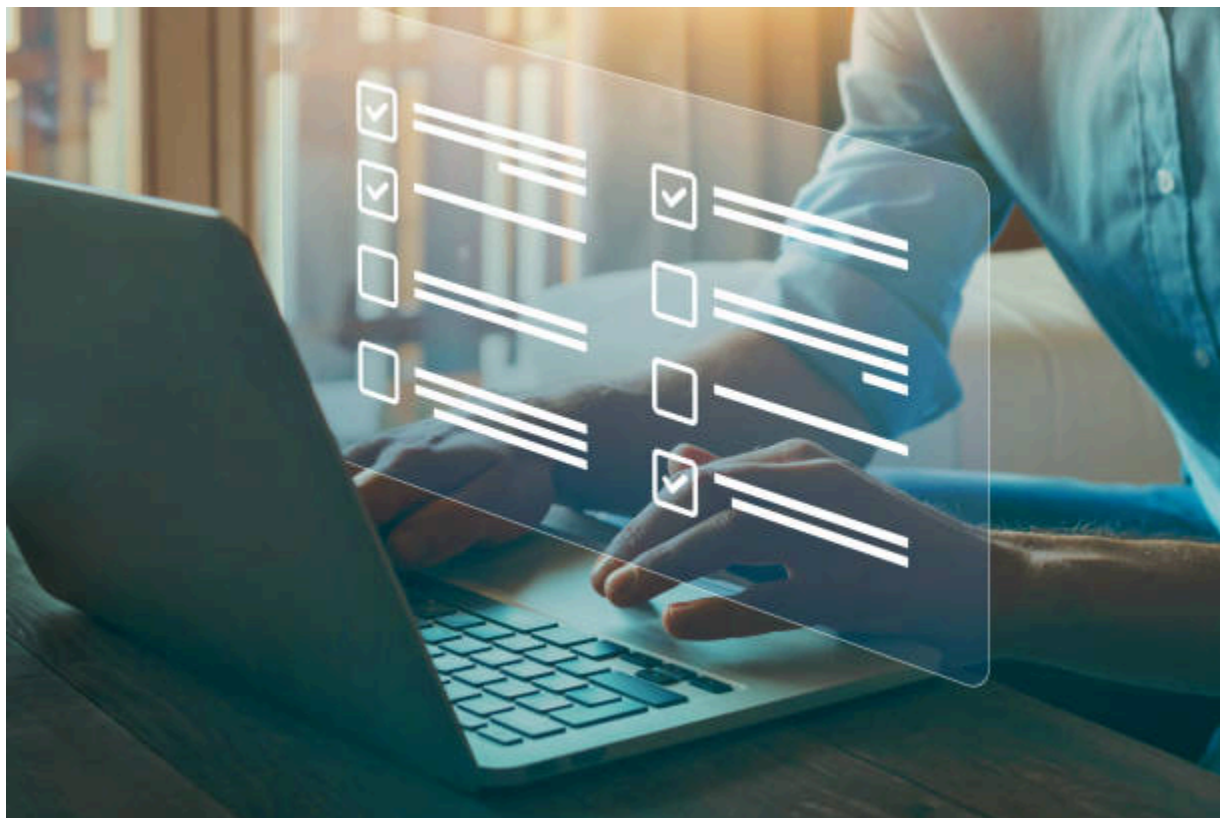


Imagen extraída de la web: <https://www.istockphoto.com/>

Modalidad

El ejercicio es **individual**.

En la fecha de entrega se debe adjuntar al campus el archivo comprimido con el siguiente formato:

PARCIAL1_NOMBRE_ALUMNA_O

Consigna del Parcial

¡Bienvenida/os al TP sobre plataformas de desarrollo! Antes de comenzar, destacamos algunos aspectos importantes:

- **Leé atentamente la consigna:** Es fundamental que leas cada punto cuidadosamente. Prestá atención a los detalles y asegurate de entender lo que se te está preguntando antes de enviar el proyecto.
- **Modalidad del examen:** El examen será de tipo práctico.

Explicación del trabajo

Crear un tablero con la lista de vuelos de un aeropuerto, que renderice la información con el array que se adjunta en esta consigna.


No usar tecnologías que no sean de frontend (HTML, CSS y Javascript) Por ejemplo PHP, Python, .NET, etc.

Además el ejercicio debe resolverse sólo con código Vanilla js, tampoco está permitido el uso de bibliotecas o frameworks como React, Vue, Angular, TypeScript, etc. Excepto casos que ayuden a desarrollar la interfaz gráfica como por ejemplo Bootstrap o Tailwind.

Rúbricas

Mostrar vuelos (4 puntos)

Crear una función llamada **renderizarVuelos(vuelos)** que reciba como argumento un array similar al del archivo adjunto **vuelos.json**, y genere



código HTML como una tabla o cualquier componente similar, que muestre uno por uno los vuelos con la siguiente información:

- Ciudad / País
- Hora de salida
- Estado

El diseño debe ser acorde al trabajo planteado y usable.

Destino	Hora de salida	Estado
 Asunción / Paraguay	05:44	En horario
 Quito / Ecuador	10:35	Retrasado
 La Paz / Bolivia	11:58	Cancelado
 Bogotá / Colombia	17:27	Embarcando
 Mendoza / Argentina	20:49	Cancelado
 Brasilia / Brasil	18:19	Embarcando
 Montevideo / Uruguay	07:22	Embarcando
 Santiago / Chile	09:47	Retrasado
 Caracas / Venezuela	14:05	En horario
 Lima / Perú	22:51	En horario

(Éste es el único punto obligatorio para aprobar el trabajo)

Obtener vuelos de forma asíncrona (3 puntos)

Crear una función llamada **obtenerVuelos()** que recupere la información de **vuelos.json** con el método **fetch()**


- Si los recupera correctamente, debe llamar a la función creada en el ejercicio anterior **renderizarVuelos(vuelos)** pasándole como argumento la respuesta.
- Si por algún motivo la respuesta falla, entonces mostrar por pantalla un error indicando que no se pudieron obtener los vuelos. Probar esto último por ejemplo escribiendo la URL de forma incorrecta.

Esta función debe ejecutarse en un intervalo de un minuto para renderizar posibles actualizaciones en los vuelos.

Filtro de búsqueda (3 puntos)

Crear una función llamada **manejadorBusquedaCodigo()** que gestione las acciones de un formulario de búsqueda, y que permita al usuario buscar el código de un vuelo (de forma local)

- Si el código pertenece a algún vuelo, entonces que vuelva a llamar a la función **renderizarVuelos(vuelos)** con el único registro encontrado.

LA950		Buscar
Destino	Hora de salida	Estado
 Buenos Aires / Argentina	09:01	Embarcando

- Si el código no le pertenece a algún vuelo, debe mostrar por pantalla un error indicando que no se pudo obtener el vuelo.



Error

El vuelo no existe. Por favor verifique e ingrese nuevamente el código.

OK

Detalle de la entrega

En la carpeta de entrega incluir:

- Un documento (por ejemplo Word) con el nombre de la/el alumna/o
- Una carpeta con el código fuente de la aplicación. Esto puede reemplazarse por la URL de [github](https://github.com).

Fecha y hora de entrega

Revisar la tarea en donde se cargó este documento.

Entrega con retraso

En caso de entregar con retraso, automáticamente se descontarán dos puntos del TP.


Por ejemplo si la nota merece un 10 (diez), esto provocará que la misma termine siendo 8 (ocho); o si es 4 (cuatro) la/el alumna/o reprobará con un 2 (dos)

Además aquellos TP que no sólo se entreguen con retraso, sino que también se entreguen una vez empezada la clase, no se aceptarán y la/el alumna/o deberá ir a recuperatorio.

Importante

Como se mencionó antes, el trabajo es individual, en caso de encontrar dos o más trabajos copiados, se anularán, y estos tendrán como calificación: 1 (uno)

Todos los puntos pueden resolverse con los temas vistos en clase hasta el momento de subir esta consigna. Sin embargo la/el alumna/o puede utilizar cualquier herramienta de ayuda como IA, pero el código no estará



garantizado como correcto, cualquier error o mala práctica que se encuentre en el código será corregido y se bajará la nota si se considera necesario.

