

Ejercicios:

Ejercicio 1 - PHP (Algoritmo de lógica)

Problema:

Escribe una función en PHP llamada `buscarNumeroFaltante($numeros)`, que reciba un array con números consecutivos del 1 al 100, pero con un número faltante. La función debe encontrar el número que falta.

Ejemplo:

php

```
$numeros = range(1, 100);

unset($numeros[array_rand($numeros)]); // Se elimina un número al
azar

echo buscarNumeroFaltante($numeros);
```

Ejercicio 2 - Laravel (CRUD Básico con Eloquent)

Problema:

Crea un CRUD en Laravel para la entidad `Tareas`. Los campos deben ser `id`, `titulo`, `descripcion` y `estado` (pendiente/completado). Implementa las siguientes rutas:

- `POST /tareas` → Crea una nueva tarea.
- `GET /tareas` → Lista todas las tareas.

- `PUT /tareas/{id}` → Actualiza una tarea.
- `DELETE /tareas/{id}` → Elimina una tarea.

Ejercicio 3 - CodeIgniter (Controlador y Modelo)

Problema:

Crea un controlador en CodeIgniter llamado `Usuarios` con una función `listarUsuarios()` que obtenga todos los usuarios desde la base de datos y los devuelva en formato JSON.

Modelo:

Debe haber un modelo `UsuarioModel` con una función `getUsuarios()` para obtener la lista de usuarios desde la base de datos.

Ruta esperada:

`GET /usuarios/listar` → Devuelve todos los usuarios en formato JSON.

Ejercicio 4 - SQL (Consulta Compleja)

Problema:

Dadas las siguientes tablas:

clientes

id	nombre
----	--------

- 1 Juan
- 2 Maria

productos

- | id | nombre |
|----|---------|
| 1 | Laptop |
| 2 | Celular |

ventas

id	cliente_id	producto_id
1	1	2
2	1	1
3	1	2
4	2	2
5	2	2

Escribe una consulta SQL que devuelva el **producto más comprado por cada cliente**.

Ejercicio 5 - JavaScript (Manejo de Arrays y Promesas)

Problema:

Dado el siguiente array de objetos:

js

```
const usuarios = [
```

```
{ nombre: "Carlos", edad: 25 },  
{ nombre: "Ana", edad: 18 },  
{ nombre: "Pedro", edad: 30 },  
{ nombre: "Sofía", edad: 22 }  
];
```

1. Escribe una función en JavaScript que filtre los usuarios con edad mayor o igual a 21.
2. Usa `setTimeout` para simular una consulta que devuelva el array filtrado después de 2 segundos.

Ejercicio 7 - SQL (Consulta Compleja con JOIN y Subconsultas)

Problema:

Dadas las siguientes tablas:

empleados

id	nombre	departamento_id
----	--------	-----------------

1	Luis	1
2	Marta	2
3	Juan	1

departamentos

id	nombre
----	--------

1	Desarrollo
2	Ventas

salarios

id	empleado_id	salario
----	-------------	---------

1	1	2500
2	2	3000
3	3	2700

Consulta requerida:

Escribe una consulta SQL para obtener el **nombre del empleado, su departamento y la diferencia entre su salario y el salario promedio de su departamento.**