

```
const express = require("express");
```

```
const app = express();
```

```
app.use(express.json());
```

```
const PRECIOS = {
```

```
  A: 300,
```

```
  B: 490,
```

```
  C: 670,
```

```
  D: 899
```

```
};
```

```
app.post("/calcular-precio", (req, res) => {
```

```
  const { seccion, dia, cantidad } = req.body;
```

```
  if (!seccion || !dia || !cantidad) {
```

```
    return res.status(400).json({ error: "Faltan algunos datos necesarios." });
```

```
  }
```

```
  const seccionMayuscula = seccion.toUpperCase();
```

```
  const diaMinuscula = dia.toLowerCase();
```

```
  if (!PRECIOS[seccionMayuscula]) {
```

```
    return res.status(400).json({ error: "Sección no válida. Debe ser A, B, C o D." });
```

```
  }
```

```
  if (cantidad < 1) {
```

```
    return res.status(400).json({ error: "La cantidad de boletos debe ser mayor a cero." });
```

```

    }

    let precioPorBoleto = PRECIOS[seccionMayuscula];

    if (diaMinuscula === "domingo") {
        precioPorBoleto *= 0.84; // 16% de descuento
    }

    let precioTotal = precioPorBoleto * cantidad;

    if (cantidad > 1) {
        precioTotal *= 0.95; // 5% de descuento por comprar más de uno
    }

    res.json({
        seccion: seccionMayuscula,
        dia: diaMinuscula,
        cantidad: cantidad,
        precioPorBoleto: precioPorBoleto.toFixed(2),
        precioTotal: precioTotal.toFixed(2)
    });
});

const PUERTO = 3000;
app.listen(PUERTO, () => {
    console.log(`Servidor funcionando en http://localhost:${PUERTO}`);
});

```

 Explicación del funcionamiento

Dependencias y configuración inicial

Se importa express y se inicializa una aplicación (app).

Se utiliza `express.json()` para poder leer datos en formato JSON enviados por el cliente.

Definición de precios

Se usa un objeto llamado PRECIOS que define cuánto cuesta un boleto según la sección:

A: \$300

B: \$490

C: \$670

D: \$899

Ruta POST /calcular-precio

Esta ruta recibe 3 datos por JSON:

seccion: la sección seleccionada (A, B, C o D).

dia: el día del evento (viernes, sábado o domingo).

cantidad: cuántos boletos se desean comprar.

Validación de datos

Se revisa que todos los datos estén presentes.

Se normaliza el texto (seccion en mayúscula, dia en minúscula).

Se verifica que la sección sea válida (A-D).

Se asegura que la cantidad sea mayor a cero.

Cálculo del precio por boleto

Se obtiene el precio base según la sección.

Si el día es domingo, se aplica un 16% de descuento, multiplicando por 0.84.

Cálculo del total

Se multiplica el precio por boleto por la cantidad.

Si la cantidad es mayor a 1, se aplica un 5% de descuento al total.

Respuesta JSON

Se devuelve un objeto con:

La sección seleccionada.

El día.

La cantidad de boletos.

El precio por boleto (ya con descuento si aplica).

El precio total (descuento incluido si aplica).

Inicio del servidor

El servidor escucha en el puerto 3000 y muestra un mensaje en consola al estar activo.