# Övning: Mini Todo-app med API

## 🎯 Övningens story

Du ska bygga en liten webbapp som hämtar **todos** från ett externt API och visar dem på webbsidan.

Syftet med övningen är att du ska träna på:

- att arbeta med **API:er**
- att förstå **asynkron JavaScript** (`async / await`)
- att använda **hämtad data i HTML (DOM-manipulation)**
- att hantera **fel på ett kontrollerat sätt**

Appen behöver inte vara snygg – fokus ligger på **funktion och förståelse**.

---

## STEG 1 – Hämta data (READ)

### Uppgift

Hämta todos från följande API:

```
https://jsonplaceholder.typicode.com/todos
```

Använd **fetch** tillsammans med **async / await** och logga resultatet i konsolen.

### Krav

- Använd en `async function`
- Använd `await fetch(...)`
- Omvandla svaret med `response.json()`
- Logga datan med `console.log`

### 💡 Tips

> Fetch kastar inte automatiskt fel vid t.ex. 404 eller 500.

> Kontrollera därför `response.ok`.

### Förväntad förståelse

- API:t returnerar **en array med objekt**
- Varje todo-objekt innehåller bland annat:
  - `title`
  - `completed`
  - `userId`

---

## STEG 2 – Visa data i HTML (READ → UI)

### Uppgift

Visa **endast 10 todos** på webbsidan.

- Lägg todos i en `<ul>`-lista
- Varje todo ska visa:
  - **titel**
  - **status** (klar / ej klar)

### Exempel

```
Buy milk
Do homework
```

### Regler

- Alla HTML-element ska skapas med **JavaScript**
- Inget innehåll får vara hårdkodat i HTML-filen
- Använd `slice(0, 10)` för att begränsa antalet todos
- Använd `forEach` **eller** `map`

---

# STEG 3 – Felhantering (DEBUG & TÄNK)

## Uppgift

Ändra API-URL:en till en **ogiltig adress**.

Om något går fel ska du visa följande meddelande på webbsidan:

> "Kunde inte hämta todos 😢"

## Krav

- Använd `try / catch`
- Kontrollera `response.ok`
- Visa felet i **DOM:en**, inte bara i `console.log`

---

# ⭐ BONUS

Välj **en** av följande:

## 🔶 A. Filtrering

Visa endast todos där:

```
completed === true
```

---

## 🔶 B. Axios-version

Gör samma API-anrop med **axios** istället för fetch.

Jämför:

- Mindre kod?
- Enklare felhantering?

## Lycka till 🚀