

# Projektrapport JavaScript

---

## Fetch

```
const pokemonObj = [];  
  
for (let i = 1; i <= 151; i++) {  
  const url = new URL("https://pokeapi.co");  
  url.pathname = `/api/v2/pokemon/${i}`;  
  pokemonObj.push(fetch(url).then((respond) => respond.json()));  
}  
  
Promise.all(pokemonObj).then((results) => {  
  const pokemonInfo = results.map((result) => ({  
    id: result.id,  
    name: result.name,  
    image: result.sprites.other["official-artwork"].front_default,  
    type: result.types.map((type) => type.type.name).join(", "),  
  }));  
  printCards(pokemonInfo);  
});  
}
```

**fetch(url)**: Börjar en förfrågan till att göra en http request från webbrowsern. **fetch()** metoden kräver endast url till den resurs man vill fetcha från. I detta fallet så är det url `https://pokeapi.co` vi försöker att nå.

**.then(respond)**: Fetchen skickar tillbaka ett svar (**Promise**) som blir till ett **response** objekt.

**respond.json()**: **response** objektet omvandlas till ett **json()** objekt. Detta skickar tillbaka ett nytt **Promise**.

**Promise.all(pokemonObj)**: **Promise.all()** använder vi för att köra flera löften parallellt och väntar på att alla löften har blivit uppfyllda innan vi läser av den sista datan.

**.then(result)**: Den sista **.then()** metoden är den som innehåller den data som vi hämtat hem. I detta fallet id, namn, bild och typ av pokemon.

## Lösningar i projektet

För det första så bestämde jag mig för att endast hämta hem 151 stycken pokemon, alltså den första generationen pokemon `for (let i = 1; i <= 151; i++)`.

I `fetchPokemon()` functionen så använder jag mig av `.map()` metoden för att t.ex. hämta hem pokemon typerna. Utan `.map()` metoden hämtades bara första typen hem, med den så hämtades flera typer hem om pokemon hade fler typer än en t.ex **psychic, fairy** istället för bara **psychic**. `.map()` gör att man får en ny array med modifierade element.

Hade från början skrivit en fetch för `fetchModalInfo()` som var väldigt lik den i `fetchPokemon()` men av någon anledning så lyckades inte datan hämtas hem så det blev att skriva en fetch på ett annat sätt för att få det att fungera med att hämta hem datan till modalen.

Inne i funktionen `printCards()` så sätter jag ett id `data-pokemon-id=${pokemon.id}` på 'Read More' knappen för att med en Event Handler sedan kunna koppla ihop rätt pokemon beskrivning med rätt pokemon när modalen poppar upp. Tyvärr har jag i nuläget inte lyckats att få den att fungera som jag vill.

Jeanette Ljungkvist 6/3-2022