

**Onsdag 28 Februari 2018**



**Back End-utveckling i PHP**

**50 YHP**



# Horoskop



# och Objekt



- Skapa en sida, **index.php**, där man med en form kan skicka sitt förnamn, efternamn och personnummer
- Dessa ska sparas i en cookie så att man bara behöver fylla i sina uppgifter 1 gång i veckan, om uppgifterna redan finns ska man direkt skickas till **myhoroscope.php** där man kan se sitt stjärntecken.
- **myhoroscope.php** ska vid varje besök skapa ett objekt från en klass (Person) där man skickar in förnamn, efternamn och personnummer i konstruktorn. Objektet ska i konstruktorn själv räkna ut sitt horoskop

# Back to Javascript



# Requests

- Allt man kallar nättrafik, som sker mellan server och klient, består utav anrop (requests) från webbläsaren och svar från servern.
- Detta är varför instruktioner till webbläsaren måste skickas innan någon output (d.v.s session\_start, header, setcookie)
- Requests kan vara av typen POST eller GET  
...även *PUT, PATCH & DELETE*

Detta kallas för förfrågans "method"

# Requests

- I alla requests följer normalt information om webbläsaren + användarens kaka med.
- Alla requests innebär i *klassisk mening* att sidan laddas om.
- Det blir vanligare och vanligare att man dock gör requests från **Javascript**, det standardiserade sättet är med **Fetch**

# Fetch

- Fetch skickar en request till en valfri URL och returnerar svaret som ett Javascript värde, vanligtvis **JSON**
- Vi kan skriva PHP sidor som inte skriver ut någon HTML utan istället skriver ut JSON
- På så sätt kan vi knyta samman Javascript med PHP ❤️

# Fetch

*Some notes....*

- Fetch skickar inte med kakor som standard, utan måste anges med inställningen "credentials"
- I PHP finns det en funktion som heter **json\_encode()** som kan göra om nästan vad som helst till JSON 🔥 🔥 🔥

# Fetch

```
fetch("/horoscope.php", {  
  method: 'POST',  
  credentials: 'include'  
})  
  .then((response) => response.json())  
  .then((json) => {  
    console.log('Gotcha');  
  })  
  .catch((err) => {  
    console.log(err);  
  });
```

# gemensam övning:

## FETCH JSON

- Skapa en ny mapp med en **products.php** samt en **index.html**
- Skriv först **products.php** och använd `json_encode` för att skriva ut en lista med produkter. Ingen html kod ska skrivas ut från sidan.
- Skapa sen er `index.html`, där ska ni ha en **<button>** som när man klickar på den använder `fetch` för att hämta innehållet i **products.php** och använder den informationen för att lista alla produkter, i JavaScript.



# Fetch mitt horoskop



Vi ska göra en helt ny horoskop-sida från scratch.

Hemsidan ska bestå av följande filer:

- **Disclaimer.php** ska innehålla sidans användarvillkor och en knapp för att godkänna. Godkännandet ska sparas i en webbkaka. Ingen annan sida ska gå att besöka om godkännandet inte finns sparat i kakan.
- **sethoroscope.php** ska bara gå att begära via POST och om villkoren är godkända, ska ta ett födelsedatum från POST och spara det nuvarande horoskopet i SESSION
- **gethoroscope.php** ska bara gå att begära via POST och om villkoren är godkända. Ska ge tillbaka det nuvarande horoskopet i SESSION som ett namn och en kort beskrivning i JSON format.
- **index.php** ska bara gå att besöka om man godkänt villkoren. Från denna sida ska man kunna ange sitt födelsedatum och med JavaScript (fetch) skicka detta till **sethoroscope.php**. Detta ska man alltid kunna göra.  
Sidan ska också använda Fetch för att kolla om ett horoskop redan finns i SESSION och då skriva ut detta på sidan med Javascript.

Tänk på att använda fetch för att uppdatera efter ett nytt datum skrivs in

Kom också ihåg att hantera eventuella fel, vad händer om inget finns i SESSION, vad gör man om cookien "går ut" (expires?)

Tack 🙇