

Internetprogrammering

DD1386

Föreläsning 10

Innehåll

- Polling
- Ajax
- Promise

Polling med Ajax

- Med hjälp av Ajax kan man uppdatera en del av en sida och alltså endast hämta den del av sidan som ska uppdateras.
- Om webbläsaren är Edge (Explorer) så kan detta skilja sig något, därför använder man av `window.XMLHttpRequest` för att identifiera webbläsaren.

```
if (window.XMLHttpRequest)
    req = new XMLHttpRequest();
else if (window.ActiveXObject)
    req = new ActiveXObject("Microsoft.XMLHTTP");
```

Polling med Ajax

- `open`, `readyState` och `onreadystatechange`
- Med funktionen `req.open(method, url)` skickar man request till servern för att hämta en resurs: `req.open("GET", "resursens url");`
- Med `req.onreadystatechange` anger man en callback funktion som anropas beroende på värdet av `req.readyState`
- `readyState` kan ha värden, 0, 1, 2, 3 och 4, (se nästa slide)

readyState

Tillståndets nummer	Tillståndsnamn	Förklaring
0	Unset	Klient skapats men open har inte anropats
1	opened	open har anropats
2	Headers_recieved	send anropats och headers är tillgänglig
3	Data håller på att laddas	Payload laddas ner
4	Data har laddats klart	Klart!

Promise

- Ett objekt som representerar resultat av en asynkron operation.
- T.ex man vill hämta data om/från en resurs (t.ex databas, filsystemet, ...) man vill inte blockera resten av sin applikation tills man får svar från resursen och då väljer man utföra hämtningen asynkront.
- Ett Promise objekt kan ha följande tillstånd:
 - Pending
 - Fullfilled
 - Rejected
- Se exemplet

Promise example

```
new Promise((resolve, reject) => {
  console.log('Initial');
  readFromKeyboard(resolve, reject);
}).then((resolv_answ) => {
  console.log('resolved',resolv_answ);
  console.log('Vad roligt att du kommer!!!');
}).catch((reject_answ) => {
  console.log('rejected',reject_answ);
  console.log('Tråkigt');
}).then(() => {
  console.log('Vilken film kan vi bestämma sedan!');
});
console.log('\njag gör något annat här medan du svarar...');
```

```
function readFromKeyboard(resolve, reject){
  const readline = require('readline');
  const { stdin: input, stdout: output } = require('process');
  const rl = readline.createInterface({ input, output });
  rl.question('Ska vi på bio på torsdag?', (answer) => {
    rl.close();
    if( answer=='nej')
      reject(answer);
    else
      resolve(answer);
  });
}
```