

# Javascript

En grundläggande introduktion

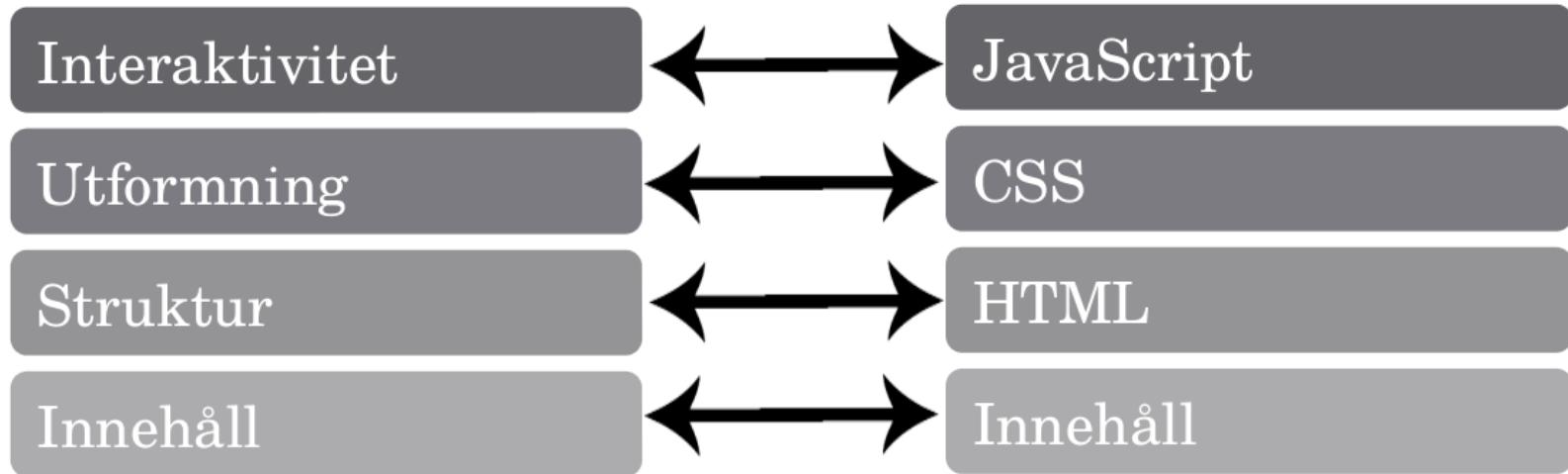
Anton Tibblin 2016-08-30

Johan Holmberg 2023-11-10

# Dagens agenda

- JavaScript?
- JavaScript – syntax?
- JavaScript – att leta upp element i HTML-dokumentet
- JavaScript – att manipulera element i HTML-dokumentet
- JavaScript – exempel

# Var passar JavaScript in?



# **JavaScript lever i din webbläsare**

Allt som behövs för att köra JavaScript är en webbläsare  
- mycket smidigt!

# Har du sett det innan?

```
▼<html>
  ▼<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>Filmtajm</title>
    <link href="css/style.css" rel="stylesheet" type="text/css">
    <link href="plugins/select2/select2.css" rel="stylesheet" type="text/css">
    <link href="bootstrap/css/bootstrap.css" rel="stylesheet" type="text/css">
    <script type="text/javascript" src="jqueryUI/js/jquery-1.8.0.min.js"></script>
    <style type="text/css"></style>
    <script type="text/javascript" src="jqueryUI/js/jquery-ui-1.8.23.custom.min.js"></script>
    <script type="text/javascript" src="plugins/select2/select2.min.js"></script>
    <script type="text/javascript" src="bootstrap/js/bootstrap.min.js"></script>
    <script type="text/javascript" src="plugins/tinySort/src/jquery.tinysort.min.js"></script>
    <script type="text/javascript" src="plugins/tinySort/src/jquery.tinysort.charorder.min.js"></script>
    <script type="text/javascript" src="plugins/tinySort/libs/jquery.opensource.min.js"></script>
    <script type="text/javascript" src="plugins/quicksand/quicksand.js"></script>
    <script type="text/javascript" src="plugins/highCharts/js/highcharts.js"></script>
    <script type="text/javascript" src="js/script.js"></script>
  </head>
  ▼<body>
```

# JavaScript – varför?

- Vi vill ge liv till annars statiska webbsidor, exempelvis:
  - automatiskt modifiera dynamiskt innehåll
    - Facebook
    - Twitter
    - Gmail
  - utföra beräkningar
    - Valutakonverteringar
  - förbättrade GUI:n
    - Animeringar
    - Lägga till/ta bort/modifiera befintliga HTML-element

# JavaScript – exempel

- <https://www.valuta.se/>
- <https://yensdesign.com/tutorials/validateform/>
- <https://zty.pe/>

# Lite historik



1995: Netscape



1996: JScript



1997: ECMAScript



2005: AJAX



2006: JQuery



2008: V8-motorn



2010: Node.JS



2015: ES 6

# JavaScript – Var skrivs det?

- JavaScript skrivs som ren text
- Kan bäddas in i HTML-kod...
- ...eller sparas i separata filer som sedan inkluderas i HTML-dokumentet.

# JavaScript – Var skrivas det?

```
▼<html>
  ▼<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>Filmtajm</title>
    <link href="css/style.css" rel="stylesheet" type="text/css">
    <link href="plugins/select2/select2.css" rel="stylesheet" type="text/css">
    <link href="bootstrap/css/bootstrap.css" rel="stylesheet" type="text/css">
    <script type="text/javascript" src="jqueryUI/js/jquery-1.8.0.min.js"></script>
    <style type="text/css"></style>
    <script type="text/javascript" src="jqueryUI/js/jquery-ui-1.8.23.custom.min.js"></script>
    <script type="text/javascript" src="plugins/select2/select2.min.js"></script>
    <script type="text/javascript" src="bootstrap/js/bootstrap.min.js"></script>
    <script type="text/javascript" src="plugins/tinySort/src/jquery.tinysort.min.js"></script>
    <script type="text/javascript" src="plugins/tinySort/src/jquery.tinysort.charorder.min.js"></script>
    <script type="text/javascript" src="plugins/tinySort/libs/jquery.opensource.min.js"></script>
    <script type="text/javascript" src="plugins/quicksand/quicksand.js"></script>
    <script type="text/javascript" src="plugins/highCharts/js/highcharts.js"></script>
    <script type="text/javascript" src="js/script.js"></script>
  </head>
  ▼<body>

    <script>
      /*
       | This function is called when the document has finished loading.
      */
      function start(){
        /*
         | On newNoteForm submit.
        */
        $("form").first().submit(function(e){
          e.preventDefault();
```

# Programmeringsspråket JavaScript

- Ett högnivåspråk
- Ett otypat språk
- Ett skriptspråk

Ni kan redan programmera, så efter lite träning med syntaxen borde ni komma igång relativt snabbt.

# Exempel på JavaScripts syntax

```
let name = "Anton";
const year = 2016;

if (a == b) {
    doSomething();
} else {
    doSomethingElse();
}

function aFunction(arg1, arg2) {
    // Do something cool with
    // arg1 and arg2
}
```

```
// Count to ten

for (let i = 0; i < 10; i++) {
    // This is a for loop!
}

let index = 0;

while (index < 10) {
    // This is a while loop!
    i++;
}
```

# Att hitta element med JavaScript

- Vi kan använda JavaScript för att hitta HTML-element!
- Vi kan letar upp element genom att:
  - Ange elementtypen (t.ex. alla `<p>`-element)
  - Ange ett klassnamn (t.ex. alla element med klassen “*blue*”)
  - Ange ett elements id
- Jämför detta med hur vi använder CSS för att hitta våra element.

# Exempel på JavaScripts syntax

```
<script>

    // Hitta alla element av typen "p"
    let allPs = document.querySelectorAll("p");

    // Hitta alla element med klassen "blue"
    let blueElements = document.querySelectorAll(".blue");

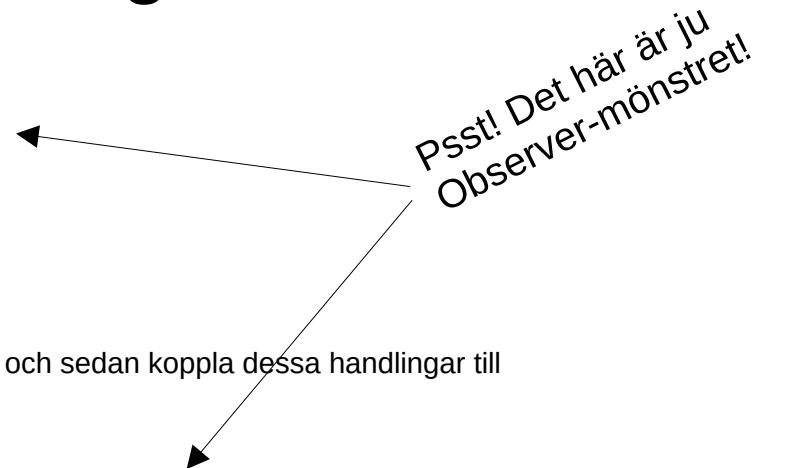
    // Hitta elementet med id:t "start"
    let start = document.querySelector("#start");

</script>
```

# Händelsedrivna användargränssnitt

- Vi "lyssnar" efter användarens handlingar, exempelvis
  - Klick
  - Musrörelser
  - Svepningar mm.
- Vi kan sedan koppla ihop dessa handlingar med element – och sedan koppla dessa handlingar till kod

```
<script>
// Användaren klickar på ett element med id:t "start"
document.querySelector("#start").addEventListener("click", someFunction);
function someFunction() {
    // Körs när användaren klickar på elementet med id:t "start"
}
</script>
```



# Att modifiera element

- Vi kan ändra ett elements attribut
- Vi kan ändra ett elements CSS-egenskaper
- Vi kan ändra innehållet i ett element
  - Bilder, text, etc.
- Vi kan lägga till och ta bort element

# Exempel på att modifiera element

```
<script>  
    // Ändra ett elements attribut  
    element.setAttribute("name", "a new name");  
  
    // Ändra CSS-egenskap för ett element  
    element.style.color = "black";  
  
    // Lägga till ett element  
    let listElement = document.createElement("ul");  
    element.appendChild(listElement);  
  
    // Ta bort ett elment  
    element.removeChild(listElement);  
  
</script>
```

# Att hantera händelser

```
<script>

// Användaren klickar på ett element med id:t "start"
document
    .querySelector("#start")
    .addEventListener("click", aFunction);

function aFunction(event) {
    // Byt färg på texten
    event.target.style.color = "red";
}

</script>
```

# This – att referera till elementet som utlöste händelsen

Nyckelordet “this” är ett smidigt sätt att referera till det element som utlöste en händelse utan att använda event-objekt:

```
// Användaren klickar på ett
// element med id:t "start"

document
    .querySelector("#start")
    .addEventListener("click",
        aFunction);

function aFunction() {
    // Byt färg på texten
    this.style.color = "red";
}

// Hitta alla element av typen "p"
let ps =
document.querySelectorAll("p");

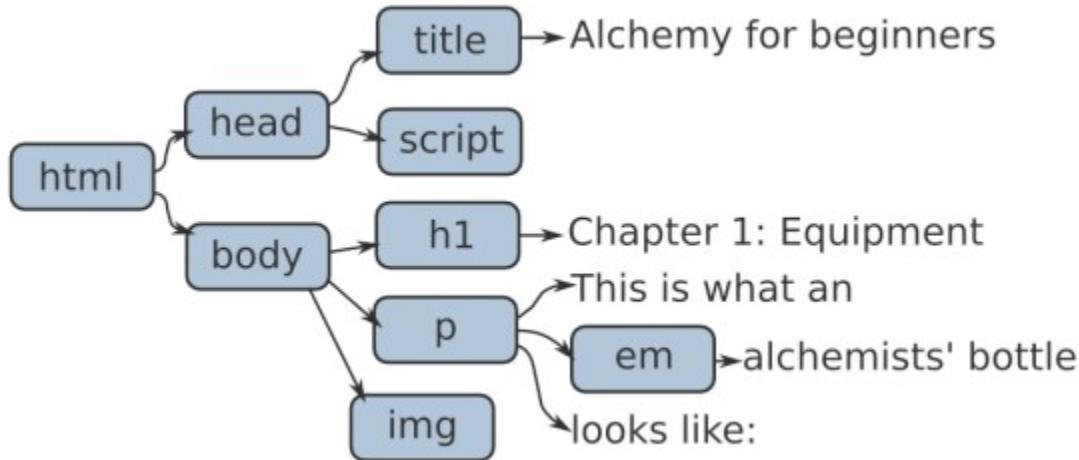
ps.forEach((pElement) => {
    pElement.addEventListener("click",
        colorize);
});

function colorize() {
    // Ändrar färg på den paragraf
    // som vi klickade på
    this.style.color = "red";
}
```

# Exempel med JavaScript

Hitta och manipulera element

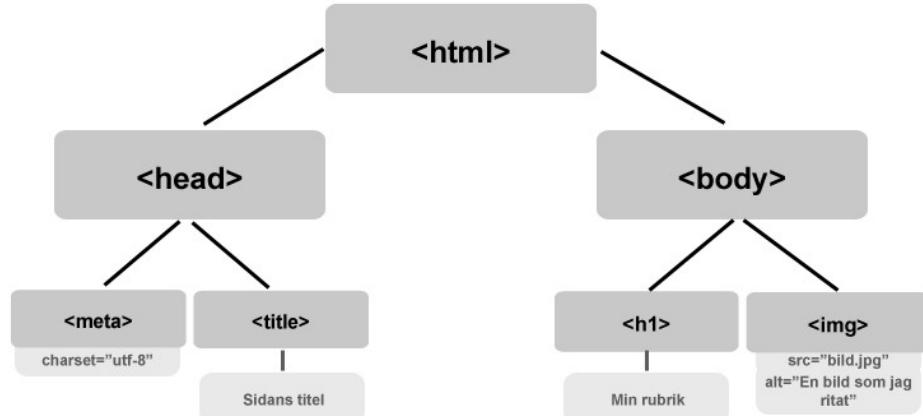
# DOM – Document Object Model



- En trädstruktur som motsvarar taggarnas ordning och nästling i HTML-dokumentet
- Varje element är en “nod” i trädet
- Relationer mellan noderna beskrivs som förälder/barn och syskon

# Källkod ↔ Noder

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Sidans titel</title>
  </head>
  <body>
    <h1>Min rubrik</h1>
    
  </body>
</html>
```



# Att navigera i DOM

Vi antar att “element” är en nod/ett element i DOM-trädet

```
// Navigera mellan noder på samma nivå
element.nextSibling
element.previousSibling

// Nedåt, barnnoder
element.firstChild
element.lastChild

// Uppåt, förälder
element.parentNode

element.childNodes // Alla barn
```

# Att hämta värden från ett formulär

```
<label for="name">Namn:</label>
<input type="text" id="name">
<button id="showName">Visa namn i popup-ruta</button>

<script>
document.querySelector("#showName").addEventListener("click", showName);
function showName() {
    let name = document.querySelector("#name").value;
    alert("Hallå " + name + "!");
}

// Med en anonym funktion
document.querySelector("#showName").addEventListener("click", () => {
    let name = document.querySelector("#name").value;
    alert("Hallå " + name + "!");
});
</script>
```

# Att modifiera element

- Vi kan ändra ett elements attribut
- Vi kan ändra ett elements CSS-egenskaper
- Vi kan ändra innehållet i ett element
  - Bilder, text, etc.
- Vi kan lägga till och ta bort element