# Table

* Webben fylld med text och bilder men även tabulär data.
* Allt som kan läggas i en spreadsheet kan även ligga i en table
* Väldigt många användningsområden därav viktigt att lära sig
* Lite mer involverande än t.ex. paragrafelementet då det går att göra mycket med en table
* Bygger kräver att vi kodar element som representerar:  
  Individuella kolumner  
  Rader som innehåller cellerna  
  Tabellen i sig  
  Header   
  Footer
* Vanligt att en table fylls från en databas.

## Table header cell element (<th>

* Till för att märka en cell som en rubric för andra celler
* Markeras som inte bara data utan även rubriker
* Har ingen synlig skillnad men har en semantisk skillnad
* Man måste bestämma om th-elementet sätter rubrik på raden eller kolumnen. För detta använder vi scope-attributet

## Table head och body (<thead>, <tbody>)

* Används för att lägga till mer struktur liknande htmldokumentets head och body
* Viktigt att skilja på th och thead.
* Bra för search engines men även för styling

## Table foot

* För att ge mer struktur. Finns inga superstrikta regler men brukar generellt innehålla summering av data alternativt information om vart datan kommer ifrån.

## Caption

* Caption är en titel för tabellen. Bör ge en överblick av vad tabellen innehåller

# Form

* Webben är har tvåvägskommunikation. Vi kan både visa data för användarna och ta emot data från användarna.
* Finns många element för output, men även de för input. Formuläret och dess kontroller!
* Skickas vanligtvis mot ett backend som tar hand om inputen.

## Formelement

* <form>-elementet wrappar om alla element som ingår i formuläret.
* Formulär ska inte vara nästlade i varandra
* Kan skicka med två http-metoder, post och get

## Input

* Vanligaste elementet i ett formulär
* Är den input användaren ger
* Kan komma i olika former so som tal eller strängar men även flerval och enkelval mm.

## Textarea

* När en rad inte är tillräckligt
* Perfekt för kommentarsfält och liknande

## Select

* Ibland vill vi inte att usern ska skriva text
* Vi har istället färdiga värden man kan välja
* Selecten gör inget av sig själv, behöver option-element
* Skapar en lista med val, kan kategorisera valen elementet optgroup och labelattributet

## Radiobutton

* När vi har få val, t.ex. ja / nej
* Alla som delar samma name attribut går under samma ”prick”. Endast ett val kan vara markerat i samma grupp.
* Ett val kan inte klickas ur

## Checkbox

* Fler boxar som användaren kan klicka i
* Kan klickas ur
* Används ofta för nyhetsbrev och accepterande av regler

## Button

* Inget formulär är komplett utan en knapp
* Kan vara i form av ett input-element eller ett button-element.

## Label

* Sätter ett ”namn” på en input. Kopplas genom id
* Skickar användaren till rätt input när namnet klickas på

## Fieldset och Legend

* Fieldset grupperar formkontroller
* Legend ge en förklaring till grupperingen.