

FORMS

Med formulär kan vi hämta in data från användaren

I ren **HTML** kan vi inte göra så mycket med informationen

```
const inputValue = document.getElementById( 'input' ).value
```

Vi kan dock strukturera upp den på rätt sätt:

för backend och javascript

```
<form action="login.php" method="POST">  
  <input type="text" name="username">  
  <input type="password" name="password">  
  <input type="submit" value="Skicka">  
</form>
```

`action` och `method` är för att skicka informationen till backend

Vi fokuserar på att strukturera tillgängliga former

`input` behöver inte ligga i ett formulär men gör ofta det för att vi ska kunna gruppera den data som finns i inputfälten

Viktigt när vi väl ska skicka informationen eller plocka informationen med JavaScript: username & password ska skickas samtidigt

Vi kollar på några vanliga inputtyper och hur vi stylar dem

Ur tillgänglighetsperspektiv behöver vi oftast märka upp inputfält

```
<form>
  <label for="todo">Add Todo</label>
  <input type="text" name="todo" id="todo">
  <input type="submit" value="Skicka">
</form>
```

input måste länkas till deras **labels** via **ID**

name är till för att skicka värdet till backend

```
<input  
  type="text"  
  name="username"  
  id="username"  
  value="zero_cool"  
  placeholder="Input username">
```

value faktiskt värde som plockas via t.ex. JavaScript
(`element.value`)

placeholder Kan inte hämtas ut, bara grafiskt

```
<form>
  <label for="birthdate">Birthdate</label>
  <input type="text"
        name="birthdate"
        id="birthdate"
        placeholder="YYMMDD">
  <input type="submit" value="Send">
</form>
```

placeholder kan vara bra för att ge extra info om fältet till användaren

Använd ibland istället för **label**

```
<label for="human">Human</label>  
<input type="checkbox" name="human" id="human">  
  
<label for="other">Other</label>  
<input type="radio" name="other" id="other">  
<input type="radio" name="other" id="other">
```

Ge radio buttons samma **name** för att välja mellan dem



Typer av text-input

- `url`
- `number`
- `search`
- `tel`
- `email`

Mest synligt på mobila enheter

Andra inputtyper

```
<select>  
  <option value="Cherry"> Cherry </option>  
  <option value="Banana"> Banana </option>  
</select>
```

```
<textarea cols="10" rows="20"></textarea>
```

Textarea kan sättas med både `rows/cols` samt höjd och bredd i CSS,
kan vara bra att köra båda.

cols: *The visible width of the text control, in average character widths. If it is specified, it must be a positive integer. If it is not specified, the default value is 20 (HTML5).*

rows *The number of visible text lines for the control.*

```
<datalist id="fruits">
  <option value="banana">Banana</option>
  <option value="apple">Apple</option>
</datalist>
<input type="text" name="fruits" list="fruits">
```

Sökbar select, som autocomplete

Fungerar inte i Safari

Vissa input-typer är `stylade native`

Innebär att operativsystemet bestämmer vilken styling som appliceras

Oberäkneligt och svårt att styla dessa element

- `checkbox`
- `radio`
- `select/option`

Om ni måste ta bort styling: **appearance: none;**

```
input{  
  -webkit-appearance: none;  
  -moz-appearance: none;  
  appearance: none;  
}
```

Ojämnt stöd och fungerar inte i alla browsers

Tar bort styling helt, du måste ersätta all styling

HTML5 VALIDERING

HTML5 har vissa attribut som sköter valideringen i frontend åt oss
Vi måste alltid validera i backend ändå men här kan vi säkra upp och ge
feedback till användaren enkelt

required

```
<form>
  <label for="username">Username</label>
  <input type="text" name="username" id="username" required>
  <input type="submit" value="Skicka">
</form>
```

Username

Password

Regex pattern

```
<form>
  <label   for="password">Password</label>
  <input   type="password"
           name="password"
           id="password"
           required
           pattern="(?=^.{8,}$)((?=.*\d)|(?=.*\W+))(?![\.\n])(?=.*[A-Z])(?=.*[a-z])"
  <input   type="submit" value="Skicka">
</form>
```

Vanligt använda regex mönster

<http://html5pattern.com/>

Gruppövning (1h): skapa "slutför köp"-formulär

1. **Research**: Undersök existerande sidor, hur många och vilka input-fält används
2. **Diskussion**: vilka element måste ni använda och hur ska det vara strukturerat
3. **Implementering**: skapa ett grundläggande formulär i HTML och CSS som du skulle vilja använda för att slutföra ett köp.

Kolla under dagens lektion på studentportalen för länktips

PSEUDO-KLASSER FÖR INPUT

Pseudo-classer

- `:active` - när elementet är nedtryckt t.ex.
- `:focus` - när elementet är fokuserat, tabbat till
- `:hover` - när man har musen över

Fler tillstånd

- `:enabled` - element som har enabled
- `:disabled` - element som har disabled
- `:checked` - ikryssad checkbox t.ex.
- `:indeterminate` - inte ikryssad
- `:valid` - rätt input
- `:invalid` - fel input

TABLE

Specifika table-egenskaper

```
table {  
  border-collapse: collapse;  
  border-spacing: 1rem;  
}  
  
th, td {  
  border: 2px solid red;  
}
```

id	name
0	zero_cool
2	master_89

Specifika table-attribut

- `colspan` - hur många columns en cell ska ta
- `rowspan` - hur många rows en cell ska ta

```
<td colspan="2"></td>
```

```
<td rowspan="2"></td>
```

```
<td colspan="2" rowspan="2"></td>
```

Semantisk markup

```
<table>
  <thead>
    <th>header</th>
  </thead>
  <tbody>
    <td>Data</td>
  </tbody>
  <tfoot>
    <th>footer</th>
  </tfoot>
</table>
```

Tillgänglighet

Definiera om headingen är för en column eller för en row

```
<th scope="col"></th>  
<th scope="row"></th>
```

Lägg till en titel

```
<caption>Title</caption>
```