

Programming - Lektion 5

...

PC 230922

Datatypen Sträng

I JavaScript finns det en datatyp som heter string. På svenska säger vi sträng.

Det är en följd av bokstäver eller snarare tecken.

Strängar omges av antingen dubbla eller enkla citationstecken.

I övningarna från förra veckan fanns några uppgifter där ni skulle manipulera strängar.

Låt användaren mata in en textsträng. Skriv program som matar ut :

- Första bokstaven
- sista bokstaven
- namnet skrivet baklänges

Exempel på strängar

```
namn = "Peter";  
color = 'BlanchedAlmond';  
bgColor = "DarkSlateBlue";  
fill(bgColor);  
greeting = "Hej " + namn;  
text(10,100, 50, greeting, color);
```

Strängar omges av dubbla eller enkla fnuttar.

Om man vill slå ihop två strängar använder man + tecken. Att slå ihop strängar kallas för *konkatenering*.

Sträng som ett objekt

```
namn = "Peter";  
greeting = "Hej " + namn;  
text(10, 100, 50, greeting.length,  
color);
```

Strängen är ett objekt med flera egenskaper och värden knuten till sig. Vi kan ta reda på strängens längd och individuella bokstäver (egentligen tecken) i strängen.

Första bokstaven

	P	e	t	e	r
index	0	1	2	3	4

```
namn = "Peter";  
b = namn.charAt(0);  
text(10, 100, 50, b, color);
```

För att hämta en enskild bokstav så använder vi `charAt(index)`. Strängar indexeras från noll, så om jag vill ha första bokstaven så behöver jag titta i index noll.

Sista bokstaven

	P	e	t	e	r
index	0	1	2	3	4

```
namn = "Peter";  
b = namn.charAt(name.length-1);  
text(10,100,50, b, color);
```

För att hämta en enskild bokstav så använder vi `charAt(index)`. Strängar indexeras från noll så jag måste titta i index

`length - 1`

Från övningar förra veckan

```
let namn = "Musse Pigg";  
  
// Längden av en sträng  
  
let längd = namn.length;  
  
// Skriv ut längden med en förklarande text  
  
// Konkatenera (slår ihop) flera strängar  
  
let output = namn + " har " + längd + " bokstäver";  
  
text(25, 25, 20, output, "black");
```

Övningar förra veckan...

```
// Om en sträng är tom så är längden noll
```

```
let tom = ""
```

```
if (tom.length == 0){
```

```
  text (22, 50, 20, "Strängen är tom", "black");
```

```
}
```



```
//Förvandla strängen till versaler  
stor = namn.toUpperCase();  
text (25, 75, 20, stor, "black");  
  
// Lägg till strängen tre gånger med en for loop  
  
let longName = "";  
for (i = 1; i <=3; i++) {  
    longName += namn;  
}
```

Vända på en sträng

	Ursprunglig sträng				
	P	e	t	e	r
index	0	1	2	3	4
	Baklänges sträng				
	r	e	t	e	P
index	0	1	2	3	4

Läs strängen bokstav för bokstav baklänges

Bokstaven i index 4 ska läggas in i index 0 i den nya strängen.

Tips det går att göra for-satser som räknar nedåt.

Vända på en sträng

	Ursprunglig sträng				
	P	e	t	e	r
index	0	1	2	3	4
	Baklänges sträng				
	r	e	t	e	P
index	0	1	2	3	4

```
namn = "Peter";

back = "" // Tom sträng

//Vänd på en sträng med hjälp av en
//for-sats som räknar nedåt

for (var i =namn.length-1 ; i>=0; i--){
    back = back + namn.charAt(i);
}

text(10,100, 50, namn, "black");
text(10,170, 50, back, "red");
```

Funktioner

- En funktion är ett block av kod
- Vi kan tänka på det som ett miniprogram inne i programmet
- En funktion utförs när den anropas (vilket kan ske på olika sätt)
- Vi har redan sett funktioner i simple.js, t ex `function update()`

```
function name(parameter1, parameter2, parameter3) {  
    // code to be executed  
}
```

Ex på en funktion

```
function myFunction(p1, p2) {  
    return p1 * p2;  
}
```

// Anrop

```
let p3 = myFunction (3, 5);
```

Funktionen

- definieras med nyckelordet `function`.
- kan ta *parametrar*
- koden omges av “måsvingar”.
- kan returnera ett resultat
- anropas via sitt namn
- anropet måste ha rätt antal *argument*

Vi bygger en funktion

Vi ska göra en funktion som omvandlar Fahrenheit till grader Celcius.

Formeln för detta är:

$$\text{Celcius} = 5/9 (\text{Fahrenheit} - 32)$$

```
function toCelsius(f) {  
  return (5/9) * (f-32);  
}  
  
let value = toCelsius(77);  
  
text(10, 50 , 20, value ,  
  "black");
```

En funktion som vänder på en sträng

Om jag vill vända på en sträng flera gånger kan jag skapa en funktion.

```
namn ="PeterC";

function bakOchFram(inString){
  var back = ""
  for (var i =namn.length ; i>=0; i--){
    back = back + namn.charAt(i);
  }
  return back;
}

bak = bakOchFram(namn);
alert(bak);
```

En funktion som ritar en tärning..

Vi ska göra en funktion som kan rita en tärning.

Funktionen ska ha tre argument, position i x-led, position i y-led och värdet som ska visas.

```
ritaTarning(20,20,5);
```

