

4.4 Tärningskast i tabell

```
1 <!-- DiceTable.html
2 Placerar 20 tärningskast i en (4x5) HTML-tabell som
3 kodas med en JavaScript-funktion utan returvärde
4 Samspel mellan HTML och JavaScript -->
5 <title>Tärningskast i tabell</title>
6 <script>
7
8 function myRandInt(a, b) // Funktion med returvärde
9 { // Slumpar heltal mellan a och b
10     return parseInt(a + Math.random() * (b - a + 1))
11 }
12
13 function TableMaker() // Funktion utan returvärde
14 { // JavaScript skriver med HTML till webbsidan:
15     document.writeln('<table border = "1" width = "50%">') // Starttagg
16     document.writeln('<caption><h3>Integer random numbers</h3></caption><tr>')
17
18     for (i = 1; i <= 20; i++)
19     {
20         document.writeln('<td><h3>' + myRandInt(1, 6) + '</h3></td>')
21         if (i % 5 == 0)
22             document.writeln('</tr><tr>') // Ny tabellrad var 5:e utskrift
23     }
24
25     document.writeln('</tr></table>') // Sluttagg
26 }
27
28 TableMaker() // Anrop av TableMaker
29
30 </script>
31 <br>Ladda om sidan (Ctrl-R) för att köra om scriptet.
```

Funktionen TableMaker()

Denna funktion som gör huvudjobbet i scriptet **DiceTable** (ovan), returnerar inget värde. Den konstruerar en tabell och skriver i den 20 tärningskast. Tabellen har 4 rader och 5 kolumner och skrivs ut med en enkel for-sats. Loopen är inte nästlad, för att åstadkomma 2D-tabellstrukturen. Istället använder den sig av en if-sats och modulooperatorn % för att byta rad var 5:e utskrift. Men i själva verket ”byter” den inte rad på ett traditionellt sätt med \n, utan skapar nya rader var 5:e utskrift medr **table**-elementets **tr**-tagg (*table row*). Det är *HTML-kod* som skickas till *JavaScript*-funktionen `document.wri-`

Integer random numbers

3	5	6	2	2
2	1	3	6	5
6	6	2	6	2
6	2	6	2	5

Ladda om sidan (Ctrl-R) för att köra om scriptet.

`println()`, se rad **22**, vilket är anmärkningsvärt. Därför vill vi titta närmare på denna konstruktion:

Samspel mellan JavaScript och HTML

I scriptet `DiceTable` på rad **15** öppnas HTML-elementet `table` i `document.writeln()`:s parameterlista och går över flera rader. Det stängs på rad **25**. `table`-elementet finns fördelat över funktionen `TableMaker()` i de olika satser som anropar `document.writeln()`, inkl. i `for`-loopen. Som vanligt används `tr` (*table row*) i `table`-elementet, för att skapa rader och `td` (*table data*) för att skapa dataceller.

På rad **22** i `for`-loopen stängs i varje varv förra raden och öppnas en ny rad. Och genom att lägga den i `if`-satsen görs detta var 5:e utskrift. Man skulle kunna extrahera all HTML-kod från `document.writeln()`:s parameterlistor och skriva den i ett separat script, för att se hur en cell i tabellen uppstår, se övn 4.5 b).

Scriptet `DiceTable`:s överordnade struktur

Scriptet består av de två funktionerna `myRandInt()` i och `TableMaker()` samt huvudprogrammet som endast består av anropet av funktionen `TableMaker()` på rad **28**. Anropet av funktionen `myRandInt()` däremot finns inte i huvudprogrammet. Denna funktion anropas i funktionen `TableMaker()`, närmare bestämt i `for`-satsen på rad **20**.