

# PHP - Strängar

En [sträng](#) (på eng. [string](#)) är en sträng av tecken (A,B,C,D...1,2,3,4...) - alltså text. Alla tecken i alfabetet, alla siffror, mellanslag och en mängd andra symboler är godtagbara tecken för strängar. En del tecken har dock speciella funktioner.

## Syntax

En sträng skall alltid skrivas innanför citattecken ("), eller apostrofer ('). Också kallat dubbelfnuttar och enkelfnuttar.

```
"Det här är en sträng"  
'Det här är också en sträng'
```

## Exempel på strängar

Vanligtvis används strängar för utskrift av text. Man kan då skriva ut strängen direkt, eller först spara den i en variabel (om man t.ex. vill skriva ut strängen i flera olika omgångar).

**Exempel** - Skriv ut strängen direkt:

```
echo "Nu skrivs den här strängen ut.";
```

**Exempel** - Strängen sparas först i en variabel och sedan skrivs innehållet ut i variabeln (märk att vi ej använder fnuttar till variabeln):

```
$mystring = "Den här strängen sparas i en variabel och skrivs sedan ut.";  
echo $mystring;
```

## Slå ihop strängar

Ibland behöver man slå ihop strängar. I PHP gör man detta med hjälp av strängoperatören ".".

**Exempel** - Skriv ut två strängar, som slås ihop med ".".

```
echo "Första strängen, " . "...andra strängen.";
```

**Exempel** - En av strängarna sparas i en variabel, sedan slås strängen ihop med variabeln, när allt skrivs ut.

```
$mystring = "Första strängen, ";  
echo $mystring . "...andra strängen.";
```

**Exempel** - man kan också "addera" en sträng till vad som finns sparad i en variabel på följande sätt.

```
$mystring = "Första strängen, ";  
$mystring .= "...andra strängen.";  
echo $mystring;
```

## Tolkningen av strängar (PHP->HTML->Resultat)

Eftersom *PHP* är en förprocesserare till *HTML* betyder det att det sker två tolkningar efter varandra innan resultatet visas upp i din webbläsare.

- Först tolkas *PHP*-koden hos servern, som producerar en textfil och den skickas till din webbläsare.
- I webbläsaren tolkas texten som *HTML*-kod, vilket betyder att saker som radbrytning och mellanslag inte fungerar som i vanlig text.

Om man vill se vad som händer däremellan, d.v.s. när *PHP*-koden har tolkats, men ännu inte tolkats som *HTML*, kan man i sin webbläsare högerklicka i sidan och välja "visa källkod".

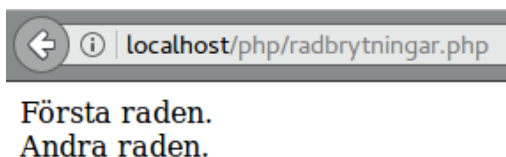
### Radbrytningar i HTML

Radbrytningar i *HTML* görs med *HTML*-kod. Man väljer helt enkelt rätt taggar för rätt ändamål. Om man vill göra en enkel radbrytning i *HTML* kan man t.ex. använda taggen `<br>`.

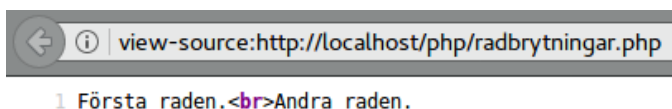
**Exempel** - radbrytning i HTML - vi använder taggen `<br>`.

```
echo "Första raden.";  
echo "Andra raden.";
```

Resultatet i webbläsaren ser ut som väntat:



Men, i källkoden ser det ut såhär:



### Radbrytningar i PHP

När vi gör en radbrytning i *PHP*, så påverkas bara textfilen som syns när vi väljer att se källkoden för en hemsida. Vi kan säga att en radbrytning i *PHP* är en radbrytning för *HTML*-koden, inte i det färdiga resultatet som visas upp i din webbläsare.

**Exempel** - radbrytning, som kanske inte fungerar i Windows (funkar däremot i Linux och Mac)

```
echo "Första raden.\n";  
echo "Andra raden.";
```

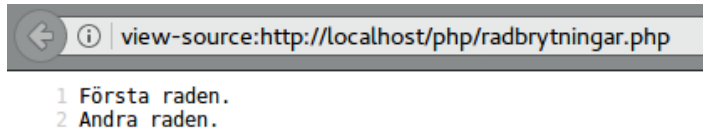
**Exempel** - radbrytning, fungerar kanske i Windows enbart

```
echo "Första raden.\r";  
echo "Andra raden.";
```

**Exempel** - radbrytning med en inbyggd konstant i *PHP* (`PHP_EOL`, som står för "PHP End of Line") som ser till att skapa rätt sorts radbrytning, oavsett system (en konstant är likt en variabel - kommer senare i kursen).

```
echo "Första raden." . PHP_EOL;  
echo "Andra raden.";
```

I alla de ovanstående exempel bör resultatet se ut så här, när man väljer att visa källkoden i webbläsaren:



## Escape-koder

I många programmeringsspråk kan man infoga speciella karaktärer inom en sträng, som får särskild funktionalitet. `\b` och `\r` är exempel på dessa och kallas för [escape-kod](#). Dessa koder fungerar inte innanför enkelfnuttar (apostrofer), utan måste läggas innanför dubbelfnuttar (citattecken).

**Exempel** - `\n` skrivs ut som vanlig text:

```
echo 'Första raden.\n';  
echo 'Andra raden.';
```

**Exempel** - `\n` skapar en radbrytning:

```
echo "Första raden.\n";  
echo "Andra raden.";
```