



At lave en breadcrumb ("Du er her: ")...

Denne artikel gennemgår, hvordan du laver en simpel breadcrumb.

Bemærk: Denne artikel indeholder kun eksempler på breadcrumb's fra sites med hierakiske sider (eg: /artikler/20040113.php).

Skrevet den **16. Feb 2009** af **jensgram** | kategorien **Programmering / PHP** | ★★★★★

[Hvad er en breadcrumb?]

En breadcrumb er en logisk navigation, der viser, hvor man befinder sig på et site. Altså et lille stykke navigation, der kan hjælpe folk til at få overblik over, hvor de befinder sig på et site.

Eksempelvis kan man støde på breadcrumb'en:

"Forside / Web / Objects", hvis man går ind på <http://www.jensgram.dk/web/objects/>. Sammenhængen er her tydelig og overblikket er der hurtigt.

Denne funktion findes på mange sites (fordi den er praktisk og giver gennemskelighed), men folk opretter til stadighed spørgsmål i PHP-kategorien. Det var så meningen, at denne artikel skulle kaste lidt lys over sagen.

[Inden vi kommer for godt i gang!]

Det havde været fedt, hvis Eksperten ville "highlighte" PHP, men det må vi tænke os til ;) Desuden er koden pænere med indrykninger, men de forsvinder også.

[Hvad skal jeg bruge?]

Det eneste krav er, at din struktur skal være som i ovenstående eksempel. Altså:

"www.ditdomæne.dk/mappe/måske_endnu_en_mappe/dinfil.php"

Funktionen kan selvfølgelig også laves ved strukturer som

"www.ditdomæne.dk/?page=dinside", men denne kan jo variere meget, så det må vente...

[Så lad os da begynde!]

Ja, hvorfor ikke? Jeg går ud fra at du er så meget inde i PHP, at du kender til termer som "funktioner", "argumenter", "variabler" og "array's". Hvis ikke så kig på [http://dk.php.net/!](http://dk.php.net/)

Vi starter med at lave vores funktion:

```
<?php
function breadcrumb()
{
    $root = 'http://www.ditdomæne.dk/';
}
?>
```

Så langt, så godt. Det er så tanken, at jeg vil benytte variabelen `$_SERVER['PHP_SELF']` i selve funktionen til at hente placeringen af selve filen.

Jeg kunne også have valgt at bruge `$_SERVER['HTTP_HOST']` i stedet for `$root`, men lad os nu bare holde fast ved den.

Hvis vi antager, at `$_SERVER['PHP_SELF']` indeholder `"/web/objects/index.php"` så er det jo tydeligt, at vi skal have opdelt denne streng ved `"/`. Dette klarer `explode()`:

```
<?php

function breadcrumb()
{
    $root = 'http://www.jensgram.dk/'; // Domænet
    $dirs = explode('/', $_SERVER['PHP_SELF']); // Split ved slashes
    $size = sizeof($dirs)-1; // Antal biblioteker
    $outp = '<a href="' . $root . '" >Forside</a >';
    $div = ' / '; // Streng mellem biblioteker i output
    $fnam = substr($dirs[$size], 0, strrpos($dirs[$size], '.')); // Filnavn
    (mellem begyndelse af sidste element i $dirs-array'et og (sidste) punktum)

    return $outp;
}

?>
```

Nu skete der jo en masse, men fortvivl ikke. Der venter forklaring.

`$dirs` vil (med ovenstående eksempel) være et array som følger:

```
Array (
    0 => '',
    1 => 'web',
    2 => 'objects',
    3 => 'index.php'
);
```

Dette skal vi bruge om lidt.

`$size` indeholder størrelsen af ovenstående array. Vi trækker en fra, da det første element er tom (strengen starter med `/'`). Denne variabel skal også bruges snart.

`$outp` indeholder på nuværende tidspunkt:

```
'<a href="http://www.jensgram.dk/" >Forside</a >'
```

Dette er det link, som jeg altid vil have stående (folk skal jo gerne kunne finde tilbage til forsiden).

\$fnam indeholder filnavnet. I dette tilfælde er det:

'index'.

Det ser måske lidt avanceret ud, men det er meget simpelt. Først henter vi positionen af det sidste punktum i det sidste element i \$dirs ('index.php'). Dette gøres med strrpos(). Denne returnerer 5 i dette tilfælde (første bogstav er 0).

Herefter benytter vi substr() til at hente fra tegn nu til tegn 5 ('index'). Så simpelt er det.

[Lav links til bibliotekerne]

Nu har vi faktisk alt hvad vi behøver til at løbe gennem bibliotekerne og lave links dertil. Til dette benytter jeg en for()-løkke:

```
<?php

function breadcrumb()
{
    $root = 'http://www.jensgram.dk/'; // Domænet
    $dirs = explode('/', $_SERVER['PHP_SELF']); // Split ved slashes
    $size = sizeof($dirs)-1; // Antal biblioteker
    $outp = '<a href="' . $root . '" >Forside</a >';
    $div = ' / '; // Adskillelse ved links mellem biblioteker i output
    $fnam = substr($dirs[$size], 0, strrpos($dirs[$size], '.')); // Filnavn
    (mellem begyndelse af sidste element i $dirs-array'et og (sidste) punktum)

    for ($i = 1; $i < $size; $i++) {
        $outp .= $div . '<a href="' . $root . $dirs[$i] . '/' >' . $dirs[$i] .
'</a >'; // Lav adskillelse og link til det aktuelle bibliotek ($dirs[$i])
        $root .= $dirs[$i] . '/'; // Tilføj det aktuelle bibliotek til $root.
    }

    return $outp;
}

?>
```

For()-løkken starter med at sætte en iterations-variabel (\$i) til 1. Dette skyldes, at jeg vil benytte \$dirs[\$i] og jeg nævnte jo tidligere at \$dirs[0] er tom. Desuden benytter jeg \$i < \$size som betingelse for løkken. Dette betyder, at jeg ikke får det sidste element i \$dirs-array'et med (fil-delen), hvilket er meningen nu!

Inden i løkken tilføjes et link til hvert bibliotek til \$outp-variablen (denne returneres senere). \$div angiver, hvad der skal være i mellem link'ene. I dette tilfælde er det strengen ' / ', men man kan benytte al HTML (billeder etc.).

Vi tilføjer desuden det aktuelle bibliotek til \$root, da vi jo skal benytte det til underbibliotekerne osv.

[Udskriv filnavnet - men ikke alle!]

Status:

Vores script kan nu returnere:

```
'<a href="http://www.jensgram.dk/" >Forside</a > / <a href="http://www.jensgram.dk/web/" >web</a >
/ <a href="http://www.jensgram.dk/web/objects/" >objects</a > / '
hvis $_SERVER['PHP_SELF'] indeholder:
'/web/objects/index.php'
```

Så kommer vi til selve filnavnet. Problemet er dog, at vi måske ikke vil have alle filer vist. For eksempel benytter jeg selv 'index.php' som indeks, hvilket vil sige at /mappe/ henviser til /mappe/index.php. Det kan således være rart at kunne undgå nogle filer, da 'mappe' vil være smartere i en breadcrumb end 'mappe' / 'index'.

Vi opretter derfor et array med "forbudte" filnavne. Vi tjekker så, om den aktuelle fil er deri når vi udskriver linket:

```
<?php

function breadcrumb()
{
    $root = 'http://www.jensgram.dk/'; // Domænet
    $dirs = explode('/', $_SERVER['PHP_SELF']); // Split ved slashes
    $size = sizeof($dirs)-1; // Antal biblioteker
    $outp = '<a href="" . $root . "" >Forside</a >';
    $div = ' / '; // Adskillelse ved links mellem biblioteker i output
    $fnam = substr($dirs[$size], 0, strrpos($dirs[$size], '.')); // Filnavn
    (mellem begyndelse af sidste element i $dirs-array'et og (sidste) punktum)
    $hide = array('index', 'default'); // Filnavne, der ikke skal linkes til

    for ($i = 1; $i < $size; $i++) {
        $outp .= $div . '<a href="" . $root . $dirs[$i] . "/" >' . $dirs[$i] .
'</a >'; // Lav adskillelse og link til det aktuelle bibliotek ($dirs[$i])
        $root .= $dirs[$i] . '/'; // Tilføj det aktuelle bibliotek til $root.
    }

    if (in_array($fnam, $hide) == false) {
        $outp .= $div . '<a href="" . $root . $dirs[$size] . "" >' . $fnam .
'</a >'; // Skriv link til filnavn, hvis det ikke skal skjules
    }

    return $outp;
}

?>
```

Først og fremmest har jeg tilføjet array'et \$hide. Her kan du selv skrive, hvilke filer der ikke skal vises i din breadcrumb.

Jeg har også tilføjet et if()-statement. Betingelsen er, at filnavnet (\$fnam) ikke skal være i array'et \$hide. Til dette benyttes in_array().

Er filnavnet tilladt skrives det til \$outp-strengen som derefter returneres.

Nu har vi faktisk en funktion, der vil returnere:

"Forside / web / objects", hvis man går ind på <http://www.jensgram.dk/web/objects/index.php>

Vi ser her, hvordan index.php sorteres fra, da navnet 'index' er i \$hide-array'et.

[En sidste finpudsning]

Ovenstående er næsten hvad vi ønskede. Det ville dog være smart, hvis de forskellige elementer havde store begyndelsesbogstaver og underscores (_) blev lavet til mellemrum i link-teksterne (selvfølgelig ikke i selve link'ene!)

Dette klares med ucwords(), der laver det første tegn i hvert ord stort (uppercase) og strrtr() til at ændre underscores til mellemrum. Begge dele sker kun i link-teksterne:

```
<?php

function breadcrumb()
{
    $root = 'http://www.jensgram.dk/'; // Domænet
    $dirs = explode('/', $_SERVER['PHP_SELF']); // Split ved slashes
    $size = sizeof($dirs)-1; // Antal biblioteker
    $outp = '<a href="' . $root . '" >Forside</a >';
    $div = ' / '; // Adskillelse ved links mellem biblioteker i output
    $fnam = substr($dirs[$size], 0, strrpos($dirs[$size], '.')); // Filnavn
    (mellem begyndelse af sidste element i $dirs-array'et og (sidste) punktum)
    $hide = array('index', 'default'); // Filnavne, der ikke skal linkes til

    for ($i = 1; $i < $size; $i++) {
        $outp .= $div . '<a href="' . $root . $dirs[$i] . '/' . " >' .
ucwords(strtr($dirs[$i], '_', ' ')) . '</a >'; // Lav adskillelse og link til
det aktuelle bibliotek ($dirs[$i])
        $root .= $dirs[$i] . '/'; // Tilføj det aktuelle bibliotek til $root.
    }

    if (in_array($fnam, $hide) == false) {
        $outp .= $div . '<a href="' . $root . $dirs[$size] . '" >' .
ucwords(strtr($fnam, '_', ' ')) . '</a >'; // Skriv link til filnavn, hvis det
ikke skal skjules
    }

    return $outp;
}

?>
```

[Afslutning]

Nu kan du indsætte en komplet breadcrumb med følgende:

```
<?php  
print breadcrumb();  
?>
```

Nemmere kan det da vist ikke blive?!

Venlig hilsen
- Jens Gram

PS: Jeg ved godt, at det er lidt sent, men vil bare lige gøre opmærksom på, at en formateret version af denne artikel kan findes på <http://www.jensgram.dk/web/e-artikler/?aid=39>

Kommentar af limemedia d. 21. Oct 2004 | 1

Ok artikel, men når man tænker på at antallet af underbiblioteker kraftigt påvirker ens ranking i søgemaskiner, vil en tilgang til "breadcrumb" problematikken være bedre via en database løsning, således man *både* kan have en cool breadcrumb OG søgemaskine venlige url's.

Kommentar af hermandsen d. 12. Mar 2004 | 2

Helt sikkert en læseværdig artikel!! :)

Kommentar af 44fire44 d. 16. Jan 2004 | 3

Takker for en fed tutorial ;)

Kommentar af netassist d. 15. Apr 2004 | 4

Kommentar af googolplex d. 30. May 2004 | 5

Fin artikel.

Kommentar af sandbox d. 26. Jan 2004 | 6

Simpelt og elegant. Fin artikel

Kommentar af bootup d. 03. Jul 2007 | 7

Kunne ikke rigtig bruge den da jeg ønskede database styret breadcrumb navigation. Men fremgår også af beskrivelsen af artiklen at den ikke er til dette :)

Kommentar af e-block d. 10. Mar 2011 | 8

jeg har i mine filnavne 'ae', 'oe' og 'aa'. hvordan ændrer man dem til 'æ', 'ø' og 'å'?