



Validering af email adresse via JavaScript

Funktion og beskrivelse til simpel validering af emailadresse, indtastet via formular på en hjemmeside.

Skrevet den **05. Feb 2009** af **limemedia** | kategorien **Programmering / JavaScript** | ★★★★★

Som udvikler/designer af hjemmesider, står man ofte med en formular indeholdende feltet email adresse, dette kunne være i forbindelse med brugerprofiler, indkøbskurve, nyhedsbreve med mere. Hvad man ikke ønsker, er tilfældige indtastninger fra brugere der enten ved en fejl indtaster forkert, eller ikke ønsker at indtaste feltet.

Hvad man kan gøre, er at tilføje en lille JavaScript funktion der validerer gyldigheden af email adressens formatering (ikke om denne reelt eksisterer). Klassikeren er, at kigge om der eksisterer et @ i linien, hvor følgende regulære ekspression tager skridtet videre og undersøger adressen sammenlignet med en bredere formatering, for at undgå de allernemmeste snyde emails.

Funktionen man kan bruge ser således ud

```
<script language="JavaScript1.2" type="text/javascript">
<!--
function verifyemail_check(str) {
    if(!str.match(/^[\\w]{1}[\\w\\.\\-]*@[\\w]{1}[\\w\\.\\-]*\\.([\\w]{2,6})$/i)) {
        return false;
    } else {
        return true;
    }
}
//-->
</script>
```

Hvis vi ønsker at koble denne på en formular, kan følgende eksempel anvendes - via onSubmit på formelementet, kan vi ved submit af formularen køre funktioner der validerer brugerens input.

Formular eksempel, der inkluderer funktionskaldet på onSubmit - jeg anvender her en anden funktion end ovenstående, da jeg samtidig gerne vil lave fejlhåndtering for brugeren

```
<form action="do_something.ext" name="demoform" onSubmit="return validate_form();">
Navn : <input type="text" name="name"><br>
E-mail : <input type="text" name="email"><br>
<input type="submit" name="submit" value="udfør">
</form>
```

Min udvidede valideringsfunktion kunne så se således ud

```
<script language="JavaScript1.2" type="text/javascript">
```

```

<!--
function verifyemail_check(str) {
    if(!str.match(/^[\\w]{1}[\\w\\.\\-]*@[\\w]{1}[\\w\\.\\-]*\\.([\\w]{2,6})$/i)) {
        return false;
    } else {
        return true;
    }
}

function validate_form(form) {
    if (!form.name.value.length > 0) {
        alert("Du glemte at indtaste dit navn !");
        form.name.focus();
        return false;
    }

    if (!verifyemail_check(form.email.value)) {
        alert("Den indtastede email er ikke gyldig !");
        form.email.focus();
        return false;
    }

    return true;
}
//-->
</script>

```

Således kunne et simpelt valideringsscript se ud - det er vigtigt at forstå dette ikke 100% sikrer data i systemet, men man undgår de værste fejlindtastninger fra brugeren. Systemet kan relativt nemt omgås og bagvedliggende systemer bør tage højde for, at data herfra ikke er 100% sikre.

Koden er taget fra <http://www.ljweb.com/usefulscripts/> hvor koden vedligeholdes og nyeste version være tilgængelig.

/ LJ

Kommentar af roenving d. 24. Jul 2005 | 1

Fint med en artikel, som ikke gør mere ud af det, end der er behov for ... Dog må jeg gøre opmærksom på, at det regulære udtryk går langt ud over den gældende standard for email-adresser (som sandbox også gør opmærksom på !-), da _ingen_ udover den originale host må beskæftige sig med det, som findes foran @-tegnet (rfc 2821 <http://www.faqs.org/rfcs/rfc2821.html>, stk. 2.3.10 Mailbox and Address !-) Iøvrigt er konstruktionen [\\w]{1} === \\w ...

Kommentar af googolplex d. 15. Jan 2004 | 2

God artikel :)

Kommentar af mayskov d. 14. Jan 2004 | 3

Kommentar af sandbox d. 15. Jan 2004 | 4

God artikel, men en lidt for restriktiv regex. Min emailadresse til Eksperten-post ville ikke blive godkendt.

Kommentar af talriny's d. 24. Oct 2004 | 5

Jeg er normalt lidt påpasselig med email validering via javascript, da jeg tit får nogle helt fine email adresser nægtet, men det er sikkert godt for nogen, og artiklen forklarer det godt.