

REGULÆRE UTTRYKK I PYTHON

Hvordan få regulære uttrykk inn i python

For å gjøre et søk etter noe i en tekst, så må vi bruke de følgende to linjene.

```
# Hent den koden vi trenger fra  
# python sin egen kodebase.  
import re
```

```
# Et funksjonskall for  
# sok etter det vi vil ha i en tekst ,  
# og leg resultatet i en variabel.  
resultat = re.search(det_vi_vil_ha , tekst)
```

Du kan søke etter hvilken som helst tekststreng.

```
import re
```

```
# en tekst variabel som forteller
```

```
# oss hva vi leter etter.
```

```
det_vi_leter_etter = "hund"
```

```
# En tekst vi vil lete i.
```

```
tekst = "det_var_en_gang_en_hund_som_het_Arne."
```

```
# Fin alle muligheter.
```

```
resultat = re.findall(det_vi_leter_etter, tekst)
```

Og vi kan finne ut om vi har funnet noe ved å si:

```
# Hvis vi har funnet noe  
if resultat:  
    # Sier vi at vi har funnet noe  
    print det_vi_leter_etter, " _finnes _i _teksten."  
  
    # Hvis vi ikke har funnet noe  
else:  
    # Da sier vi at vi ikke har funnet noe  
    print det_vi_leter_etter, " _finnes _ikke _i _teksten."
```

- ▶ Bortsett fra å kunne søke etter noe i en tekst, hva er spesielt med regulære uttrykk?
- ▶ Spesielle symboler i teksten vil gjøre søket mer avansert.

Symbolet . (\ \ .)

. kan være hva som helst. Hvis du ville ha en dot (for eksempel søke etter filer som "hei.txt"), må du bruke \ \ .

```
import re
```

```
det_vi_leter_etter = "h..d" # Her er det vi leter etter
```

```
tekst = "det_var_en_gang_en_hund_som_het_Arne."
```

```
resultat = re.findall(det_vi_leter_etter, tekst)
```

```
if resultat:
```

```
    print det_vi_leter_etter, "_finnes_i_teksten."
```

```
else:
```

```
    print det_vi_leter_etter, "_finnes_ikke_i_teksten."
```

Symbolet *

* vil finne alle symbol som er rett foran den 0 eller flere ganger.

Eksempel: “hu*nd” vil matche

- ▶ hnd
- ▶ hund
- ▶ huund
- ▶ huuund
- ▶ etc.

Paranteser ()

Med () kan du lage større grupper av symboler og få ut hva du har funnet. Eksempel: “h(..)d” vil matche hund og i resultatet fra søket kan du finne ut at .. var “un”

Med dette kan vi begynne å crawle Webben for info. For eksempel, hvordan vil du prøve å finne bildet i dette dokument?

```
<html>
  <head>
    <title>Digge studier ved UiO</title>
  </head>

  <body>
    <h1>Studentliv ved UiO</h1>
    <p>Det er mye digge studier ved UiO</p>
    
  </body>
</html>
```

Vi vet at bilder må starte med "< img" og slutte med "> ". Hva som er inni der er uvisst/ Så om vi ikke vet hva som er der, så kan vi bruke ".", som representerer hvilken som helst bokstav. Men vi trenger den mer enn 1 gang. Hvordan løser vi det problemet? "*" vil ta et symbol mer enn 1 gang. ".*" vil da ta hva som helst. Så da prøver vi '< img.*src=".*".* >' Da vil du kunne finne ut om et dokument har et bilde, men for å hente det ut, må vi bruke ().

```
'< img src="(.)'" >'
```

Problem

Et problem med denne koden

```
import re

tekst = '<html><head><title>Digge_studier_ved_UiO</title>'

det_vi_leter_etter = r'<img.*src="(.)".*>'

resultat = re.findall(det_vi_leter_etter, tekst)
print resultat
```

Symbolet ?

? kan bety to ting

- ▶ sammen med * betyr det at du vil prøve å finne så lite som mulig med denne stjernen.
- ▶ sammen med alt annet, betyr at det rett forran kan forekomme 0 eller 1 gang

Det er veldig greit om du skal unngå å få feil ting tilbake.

Så hvis vi prøver igjen, der vi pakker ? på alle stjernene våre, slik at vi leter etter så lite som mulig hver omgang. Da får vi
'< img.*?src="(.*?)" .*?> ' la oss prøve

Prøv på en live webside ved bruk av urllib og urlopen.