

Fredagspython - Hjemmeside

Denne oppgaven handler om å lage ei enkel nettside ved hjelp av Flask, som er ei pakke som hører til Python. Ved hjelp av Flask kan vi lett lage en nettside med Python. HTML og CSS er fint for å lage fine nettsider, men vi trenger en minimal mengde for å få siden vår til å fungere.

Dokumentasjonen til flask finner dere her:

<http://flask.pocoo.org/docs/1.0/>

Vi kan teste sidene våre lokalt på egne maskiner, men vi kan også kjøre koden vår slik at alle får tilgang til siden. Blant annet Pythonanywhere tilbyr denne muligheten.

<https://www.pythonanywhere.com/>

For spørsmål om HTML:

<https://www.w3schools.com/html/>

Komme i gang

Vi skal vise dere et kort eksempel, hvor vi lager en kalkulator hvor brukeren kan skrive inn tall, og få svaret.

Eksempel 1: Hello World

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello, World!'

if __name__ == '__main__':
    app.run()
```

Det etter @ kaller vi en *decorator*. I dette tilfellet sier den at gitt URL-en, skal funksjonen under kalles på.

Vi kan returnere en vanlig streng, men den vil kjøres som HTML-kode. Pass derfor på at formateringen ikke alltid fungerer som i terminalen.

Eksempel 2: Kalkulator-side

Vi kan utvide Hello World-eksempelet vårt ved å legge inn litt html. Vi lager et eget HTML-dokument, som vi kan lese med Flask sin `render_template()`.

På hovedsiden vår lager vi en knapp som det står "kalkulator" på, som fører oss til kalkulatorsiden.

På kalkulatorsiden tar vi inn to tall, og sender dette til en side som skriver svaret til brukeren.

Hovedprogrammet vårt ser da slik ut:

```
#Hjemmeside

from flask import Flask
from flask import render_template
from flask import request

app = Flask(__name__)

@app.route('/')
def helloWorld():
    #return 'Hello, World!'
    return render_template("hjem.html")

@app.route('/kalkulator')
def kalkulator():
    print("ost")
    return render_template("matte.html")

@app.route('/kalkulator/resultat',methods=['POST'])
def resultat():
    print("inni")
    tall1 = int(request.form["tall1"])
    print(tall1)
    tall2 = int(request.form["tall2"])
    print(tall2)
    summen = tall1 + tall2
    return "Resultatet av summering er:<br>" + str(summen)

if __name__ == '__main__':
    app.run()
```

I HTML-dokumentet *matte.html* har vi starten på en *form*:

```
<form action="/kalkulator/resultat" method="post">
```

action betyr at vi sender brukeren til den siden, *method="post"* betyr at vi sender med variabler til den siden vi sender brukeren til. Det er viktig at dekoratoren i python-koden vår også indikerer at vi skal ta imot data fra en side som har en post-metode:

```
@app.route('/kalkulator/resultat', methods=['POST'])
```

Da kan vi gjøre følgende:

```
tall1 = int(request.form["tall1"])
```

Det vi gjør her, er å hente en av verdiene vi har sendt, fra `request.form`. Det er ei ordbok, som vi kan slå opp verdiene vi vet vi sendte. Her konverterer vi til et heltall.

Flere utfordringer:

Utfordring 1:

Her lagde vi en side som legger sammen tall. Utvid siden slik at brukeren må skrive hva slags operasjon vi skal bruke på de to tallene. Kan vi legge til andre typer funksjonalitet?

Utfordring 2:

Bruk oppgaven fra forrige fredagspython (tekstgenerering) til å skrive en tekst til brukeren når en knapp blir trykket på.

Utfordring 3:

Lag ei ordbok på nett. La python lese inn ei tekstfil med ordpar på et hvilket som helst språk, og la programmet ditt slå opp ordet brukeren skriver.

Utfordring 4:

Lag flere sider som du selv vil, og prøv deg fram.

