

```
# -*- coding: utf-8 -*-  
"""
```

Created on Wed Apr 14 17:26:57 2021

```
@author: krist  
"""
```

```
# impoterer nødvendige bibliotek  
# skal kjøres kl 12:00
```

```
from entsoe import EntsoeRawClient  
import pandas as pd  
import datetime as dt  
import numpy as np  
from currency_converter import CurrencyConverter  
# definerer CurrencyConverter() funksjonen som c  
c = CurrencyConverter()
```

```
# API key for å be om prisinformasjon om elektrisitet fra entsoe  
client = EntsoeRawClient(api_key='254758a9-c9f0-45da-b9ed-afa7889d9956')
```

```
# definerer Funksjonen som blir kalt i hovedprogrammet(planer.py)  
# funksjonen henter strømprisen for dagen og regner ut gjennomsnittsprisen
```

```
def get_el_price():  
    # finner den gjellende dagen slik at men får prisene for riktig dag  
    today = str(dt.date.today()).replace('-', '')  
    tomorrow = str(dt.date.today() + dt.timedelta(days=1)).replace('-', '')  
    #definerer tidssone for å få prisene for riktig tid  
    start = pd.Timestamp(today, tz='Europe/Oslo')  
    end = pd.Timestamp(tomorrow, tz='Europe/Oslo')  
    # NO_3 er området man henter priser for NO_3 = Midt-Norge  
    country_code = 'NO_3'  
    # henter prisene fra entsoe  
    d_a_p = client.query_day_ahead_prices(country_code, start, end)  
    date = dt.date.today().strftime('%d-%m-%y')
```

```
# lager en midlertidig csv-fil som kan manipuleres for å få den informasjonen ønskes  
with open('temp_csv/outfile.csv', 'w') as f:  
    f.write(d_a_p)
```

```
# Åpner csv-filen og fjerner alle unødvendige rader  
price_csv = pd.read_csv('temp_csv/outfile.csv')  
index_price = []  
for i in range(29, 122, 4):  
    index_price.append(i)  
    price = price_csv.iloc[index_price]
```

```
# trekker ut prisen (EUR / MWh), og beregner gjennomsnittsprisen for dagen  
# og konverterer til Kr/kWh ved å bruke CurrencyConverter funksjonen  
price_list = []  
for l in range(0, len(price)):  
    price_list.append(float(str(price.iloc[l])[68:73]))
```

```
kr_kwh = round(c.convert(np.mean(price_list), 'EUR', 'NOK')/1000, 4)
```

```
# lagrer prisen i en csv-fil slik at den kan hentes når strømregninga skal regnes ut.  
price_df = pd.DataFrame(data={'Date': [date], 'Kr/kWh': [kr_kwh]})  
price_df.to_csv('temp_csv/avg_price_day.csv', index=False)
```

```
pf = pd.read_csv('temp_csv/avg_price_day.csv')  
print(pf)  
return pf
```