

```
import numpy
import pandas as pd
import numpy as np
from numpy import mean
```

```
#Variabili da inserire in input
```

```
Zona="" #riferita a quelle indicate dal GME per il calcolo del PZO, in maiuscolo
Data_i="" #data d'inizio nel formato AAAAMMGG
Data_f="" ##data di fine nel formato AAAAMMGG
TRASe= #tariffa di trasmissione in c€/kWh
BTAu= #tariffa di distribuzione in c€/kWh
TPac= #tariffa premio in €/MWh
Punti_prelievoBT=
Punti_prelievoMT=
S= #superficie utile totale
k= #coefficiente per il calcolo della potenza rinnovabile da realizzare come obbligo
P= #potenza elettrica installata
```

```
pz_file=pd.read_csv("C:\\Users\\studi\\Desktop\\Dropbox (Politecnico Di Torino Studenti)\\Tesi\\Altro\\anno_2020_prezzo_zonale_orario.csv", delimiter=';')
#percorso csv relativo ai prezzi zonali orari in input
cs_file=pd.read_csv("C:\\Users\\studi\\Desktop\\Dropbox (Politecnico Di Torino Studenti)\\Tesi\\Altro\\Consumo_Zone.csv", delimiter=';')
#percorso csv relativo ai prelievi elettrici in input
pr_file=pd.read_csv("C:\\Users\\studi\\Desktop\\Dropbox (Politecnico Di Torino Studenti)\\Tesi\\Altro\\Produzione_Zone.csv", delimiter=';')
#percorso csv relativo alla produzione elettrica in input
```

```
#variabili derivate da dati in input
TRASe=TRASe/100 #in €/kWh
BTAu=BTAu/100 #in €/kWh
CUaf=TRASe+BTAu #corrispettivo unitario di autoconsumo forfetario
cprBT=0.026 #coefficiente delle perdite evitate in BT
cprMT=0.012 #coefficiente delle perdite evitate in MT
TPac=0.1 #in €/kWh
P0=S/k #potenza da realizzare come obbligo da fonti rinnovabili
F=(P-P0)/P #fattore da applicare a lac per considerare che l'incentivo non si applica alla potenza d'obbligo
```

```
#Vettori/variabili da definire
```

```
prezzo_h=[]
E_imm_h_BT=[]
E_imm_h_MT=[]
E_prel_h_BT=[]
E_prel_h_MT=[]
Eac_h_BT=[]
Eac_h_MT=[]
Cac2_h_BT=[]
Cac2_h_MT=[]
Rac_h_BT=[]
Rac_h_MT=[]
```

```
#vettore dei prezzi della zona scelta estratto dal csv
```

```

for k in pz_file.columns:
    if Zona in k:
        prezzo = (pz_file[k].tolist())
prezzo=np.array(prezzo,dtype=float)
k=0

#vettori dati di prelievo e di consumo estratti dai rispettivi csv (BT e MT)
i=1
j=0
consumo_BT=np.zeros(len(prezzo))
for k in cs_file.columns:
    if 'BT' in k:
        cons=(cs_file[k].tolist())
        cons=np.array(cons,dtype=float)
        if(i==1):
            consumo_BT=cons
        else:
            consumo_BT=consumo_BT+cons
        i=i+1
k=0
i=1
consumo_MT=np.zeros(len(prezzo))
for k in cs_file.columns:
    if 'MT' in k:
        cons=(cs_file[k].tolist())
        cons=np.array(cons,dtype=float)
        if(i==1):
            consumo_MT=cons
        else:
            consumo_MT=consumo_MT+cons
        i=i+1
k=0
i=1
produzione_BT=np.zeros(len(prezzo))
for k in pr_file.columns:
    if 'BT' in k:
        prod=(pr_file[k].tolist())
        prod=np.array(prod,dtype=float)
        if(i==1):
            produzione_BT=prod
        else:
            produzione_BT=produzione_BT+prod
        i=i+1
k=0
i=1
produzione_MT=np.zeros(len(prezzo))
for k in pr_file.columns:
    if 'MT' in k:
        prod=(pr_file[k].tolist())
        prod=np.array(prod,dtype=float)
        if(i==1):
            produzione_MT=prod
        else:
            produzione_MT=produzione_MT+prod
        i=i+1

```

```
k=0
i=0
```

```
#sottovettori nell'intervallo temporale scelto (PZO, prelievi e produzione)
```

```
for i in pz_file.columns:
```

```
    if "Data" in i:
```

```
        data=(pz_file[i].tolist())
```

```
while k<len(data):
```

```
    if str(data[k])!=Data_i:
```

```
        k=k+1
```

```
    else:
```

```
        j=k
```

```
        while str(data[j])!=Data_f:
```

```
            prezzo_h.append(prezzo[j]/1000)
```

```
            E_prel_h_BT.append(consumo_BT[j])
```

```
            E_prel_h_MT.append(consumo_MT[j])
```

```
            E_imm_h_BT.append(produzione_BT[j])
```

```
            E_imm_h_MT.append(produzione_MT[j])
```

```
            j=j+1
```

```
        break
```

```
k=0
```

```
while k<len(data):
```

```
    if str(data[k])!=Data_f:
```

```
        k=k+1
```

```
    else:
```

```
        prezzo_h.append(prezzo[k]/1000)
```

```
        E_prel_h_BT.append(consumo_BT[k])
```

```
        E_prel_h_MT.append(consumo_MT[k])
```

```
        E_imm_h_BT.append(produzione_BT[k])
```

```
        E_imm_h_MT.append(produzione_MT[k])
```

```
        k=k+1
```

```
i=0
```

```
j=0
```

```
k=0
```

```
#Calcolo vettore energia condivisa e parti orarie delle componenti del corrispettivo
```

```
while j<len(prezzo_h):
```

```
    Eac_h_BT.append(min(E_imm_h_BT[j],E_prel_h_BT[j])) #energia condivisa in BT
```

```
    Cac2_h_BT.append(prezzo_h[j]*Eac_h_BT[j]*cprBT) #Secondo addendo delle restituzioni componenti  
orario in BT
```

```
    Rac_h_BT.append(prezzo_h[j]*E_imm_h_BT[j]) #ritiro dell'energia orario in BT
```

```
    Eac_h_MT.append(min(E_imm_h_MT[j], E_prel_h_MT[j])) #energia condivisa in MT
```

```
    Cac2_h_MT.append(prezzo_h[j] * Eac_h_MT[j] * cprMT) #Secondo addendo delle restituzioni compone  
nti orario in MT
```

```
    Rac_h_MT.append(prezzo_h[j] * E_imm_h_MT[j]) #ritiro dell'energia orario in MT
```

```
    j=j+1
```

```
#Calcolo componenti del corrispettivo nel periodo temporale scelto
```

```
Eac_m_BT=sum(Eac_h_BT) #energia condivisa in BT
```

```
Eac_m_MT=sum(Eac_h_MT) #energia condivisa in MT
```

```
E_imm_m_BT=sum(E_imm_h_BT) #energia immessa in rete in BT
```

```
E_imm_m_MT=sum(E_imm_h_MT) #energia immessa in rete in MT
```

```
Cac_BT=Eac_m_BT*CUaf+sum(Cac2_h_BT) #Restituzioni componenti tariffarie in BT
```

```
Cac_MT=Eac_m_MT*CUaf+sum(Cac2_h_MT) #Restituzioni componenti tariffarie in MT
```

```
lac_BT=TPac*Eac_m_BT*F #Incentivazione dell'energia condivisa in BT
```

```
lac_MT=TPac*Eac_m_MT*F #Incentivazione dell'energia condivisa in MT
Rac_m_BT=sum(Rac_h_BT) #Ritiro dell'energia in BT
Rac_m_MT=sum(Rac_h_MT) #Ritiro dell'energia in MT
```

```
#Calcolo corrispettivo totale nel periodo temporale scelto
Corrispettivo=Cac_BT+Cac_MT+lac_BT+lac_MT+Rac_m_BT+Rac_m_MT
print(Corrispettivo)
```

```
#Salva csv energia condivisa nel percorso scelto
numpy.savetxt("C:\\Users\\studi\\Desktop\\Dropbox (Politecnico Di Torino Studenti)\\Tesi\\Altro\\energia_c
ondivisa.csv",Eac_h_BT,delimiter=';')
```