Sending emails using python

Topics: import packages, SMTP lib, sending a request to the server

In [ ]:
```
import smtplib
from email.message import EmailMessage
email = EmailMessage() ## Creating a object for EmailMessage
email['from'] = 'xyz name'   ## Person who is sending
email['to'] = 'xyz id'       ## Whom we are sending
email['subject'] = 'xyz subject'  ## Subject of email
email.set_content("Xyz content of email") ## content of email
with smtlib.SMTP(host='smtp.gmail.com',port=587)as smtp:
## sending request to server

    smtp.ehlo()          ## server object
smtp.starttls()     ## used to send data between server and client
smtop.login("email_id","Password") ## login id and password of gmail
smtp.send_message(email)   ## Sending email
print("email send")    ## Printing success message
```

Medium Article Reader

Topics: web scraping, text to speech

In [ ]:
```
import pyttsx3
import requests
from bs4 import BeautifulSoup
engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices')
engine.setProperty('voice', voices[0].id)
def speak(audio):
  engine.say(audio)
  engine.runAndWait()
text = str(input("Paste article\n"))
res = requests.get(text)
soup = BeautifulSoup(res.text,'html.parser')
articles = []
for i in range(len(soup.select('.p'))):
    article = soup.select('.p')[i].getText().strip()
    articles.append(article)
text = " ".join(articles)
speak(text)
# engine.save_to_file(text, 'test.mp3') ## If you want to save the speech as a audio file
engine.runAndWait()
```

Alarm clock

Topics: DateTime module, play sound library

In [ ]:
```
from datetime import datetime
from playsound import playsound
alarm_time = input("Enter the time of alarm to be set:HH:MM:SS\n")
alarm_hour=alarm_time[0:2]
alarm_minute=alarm_time[3:5]
alarm_second=alarm_time[6:8]
alarm_period = alarm_time[9:11].upper()
print("Setting up alarm..")
while True:
    now = datetime.now()
    current_hour = now.strftime("%I")
    current_minute = now.strftime("%M")
    current_seconds = now.strftime("%S")
    current_period = now.strftime("%p")
    if(alarm_period==current_period):
        if(alarm_hour==current_hour):
            if(alarm_minute==current_minute):
                if(alarm_seconds==current_seconds):
                    print("Wake Up!")
                    playsound('audio.mp3')
                    break
```

URL shortener

Create a python script to a short URL using an API.

In [ ]:
```
from __future__ import with_statement
import contextlib
try:
        from urllib.parse import urlencode
except ImportError:
        from urllib import urlencode
try:
        from urllib.request import urlopen
except ImportError:
        from urllib2 import urlopen
import sys

def make_tiny(url):
        request_url = ('http://tinyurl.com/api-create.php?' +
        urlencode({'url':url}))
        with contextlib.closing(urlopen(request_url)) as response:
                return response.read().decode('utf-8')

def main():
        for tinyurl in map(make_tiny, sys.argv[1:]):
                print(tinyurl)

if __name__ == '__main__':
        main()
```

Weather app

Topics: Web scraping, scraping google

In [ ]:
```
from bs4 import BeautifulSoup
import requests
headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/58.0.3029.110 Safari/537.3'}

def weather(city):
    city=city.replace(" ","+")
    res = requests.get(f'https://www.google.com/search?q={city}&oq={city}&aqs=chrome.0.35i39l2j0l4j46j69i60.6128j1j7&sourceid=chrome&ie=UTF-8',headers=headers)
    print("Searching in google.....\n")
    soup = BeautifulSoup(res.text,'html.parser')
    location = soup.select('#wob_loc')[0].getText().strip()
    time = soup.select('#wob_dts')[0].getText().strip()
    info = soup.select('#wob_dc')[0].getText().strip()
    weather = soup.select('#wob_tm')[0].getText().strip()
    print(location)
    print(time)
    print(info)
    print(weather+"°C")

print("enter the city name")
city=input()
city=city+" weather"
weather(city)
```

Key logger

Topics: pynput library, interaction with files

In [ ]:
```
from pynput.keyboard import Key, Controller,Listener
import time
keyboard = Controller()


keys=[]
def on_press(key):
    global keys
    #keys.append(str(key).replace("'",""))
    string = str(key).replace("'","")
    keys.append(string)
    main_string = "".join(keys)
    print(main_string)
    if len(main_string)>15:
      with open('keys.txt', 'a') as f:
          f.write(main_string)
          keys= []
def on_release(key):
    if key == Key.esc:
        return False

with listener(on_press=on_press,on_release=on_release) as listener:
    listener.join()
```

Dice roll simulator

Topics: random module, looping, and if-else

In [ ]:
```
import random
while True:
     print(''' 1. roll the dice            2. exit     ''')
     user = int(input("what you want to do\n"))
     if user==1:
        number = random.randint(1,6)
        print(number)
     else:
        break
```

Guess the number game

Topics: random module, for loop, f strings

In [ ]:
```
import random
number = random.randint(1,10)
for i in range(0,3):
    user = int(input("guess the number"))
    if user == number:
        print("Hurray!!")
        print(f"you guessed the number right it's {number}")
        break
if user != number:
    print(f"Your guess is incorrect the number is {number}")
```

Random password generator

Topics: random module, joining strings, taking input

In [ ]:
```
import random
passlen = int(input("enter the length of password"))
s="abcdefghijklmnopqrstuvwxyz01234567890ABCDEFGHIJKLMNOPQRSTUVWXYZ!@#$%^&*()?"
p =  "".join(random.sample(s,passlen ))
print (p)
```