

Shubhangi_Dhikale_35

1. Write a Python program to read an entire text file

```
In [2]: Text="""python is a popular general purpose programming language.
It is used in machine learning,web development,desktop application,and many other
file=open("Myfile.txt","w")
file.write(Text)
file.close()

file=open("Myfile.txt","r")
a=file.read()
print(a)
```

python is a popular general purpose programming language.
It is used in machine learning,web development,desktop application,and many other fields.

2. Write a Python program to read the first n lines of a file

```
In [5]: with open ("Myfile.txt","r")as file:
        a=file.readline()
        print(a)
```

python is a popular general purpose programming language.

3. Write a Python program to append text to a file and display the text

```
In [2]: text="python is a programming language and python is used for machine learning and de
file=open("myfile.txt","a")
file.write(text)
file.close()

file=open("myfile.txt","r")
file.read()
print(text)
```

python is a programming language and python is used for machine learning and deep learning

4. Write a Python program to read the last n lines of a file

```
In [13]: file=open("myfile.txt","r")
lst=file.read()
last_n=10
for i in range(len(lst)):
    if i>last_n:
        print(lst[i],end="")
file.close()
```

programming language and python is used for machine learning and deep learning

5. Write a Python program to read a file line by line and store it into a list

```
In [21]: text="""python is a popular general purpose programming language.
It is used in machine learning.web development,desktop application,and many other
file=open("Myfile.txt","w")
file.write(text)
file.close()

file=open("Myfile.txt","r")
a=file.readlines()
print(a)
```

```
['python is a popular general purpose programming language.\n', 'It is used in
machine learning.web development,desktop application,and many other fields.']
```

6. Write a Python program to count the number of lines in a text file

```
In [22]: file=open("Myfile.txt","r")
a=len(file.readlines())
print(a)
```

2

7. Write a Python program to count the frequency of words in a file

```
In [28]: from collections import Counter
def word_count(test_text):
    with open(test_text) as f:
        return Counter(f.read().split())
print("Number of words in the file:",word_count("Myfile.txt"))
```

```
Number of words in the file: Counter({'is': 2, 'python': 1, 'a': 1, 'popular':
1, 'general': 1, 'purpose': 1, 'programming': 1, 'language.': 1, 'It': 1, 'use
d': 1, 'in': 1, 'machine': 1, 'learning.web': 1, 'development,desktop': 1, 'app
lication,and': 1, 'many': 1, 'other': 1, 'fields.': 1})
```

8. Write a Python program to copy the contents of a file to another

file

```
In [32]: file=open("Myfile.txt","r")
a=file.read()
print(a)

file1=open("myfile.txt","w")
file1.write(a)
file.close()
file1.close()

file1=open("myfile.txt","r")
b=file1.read()
print(b)
```

python is a popular general purpose programming language.
It is used in machine learning.web development,desktop application,and many other fields.
python is a popular general purpose programming language.
It is used in machine learning.web development,desktop application,and many other fields.

9.Write a Python script to display the various Date Time formats - Go to the editor

a) Current date and time b) Current year c) Month of year d) Week number of the year e) Weekday of the week f) Day of year g) Day of the month h) Day of week

```
In [34]: import datetime
current_date=datetime.datetime.today()
print("current_date is:",current_date)
print(current_date.strftime("current year: %Y"))
print(current_date.strftime("Month number of the year:%B"))
print(current_date.strftime("Week number of the year:%U"))
print(current_date.strftime("%w"))
print(current_date.strftime("%j"))
print(current_date.strftime("%A"))
print(current_date.strftime("%d"))
```

```
current_date is: 2022-08-01 17:43:58.208576
current year: 2022
Month number of the year:August
Week number of the year:31
1
213
Monday
01
```

10.Write a Python program to determine whether a given year is a leap

year

```
In [35]: import datetime

num=int(input())
last_date=datetime.datetime(num,12,31)
days=int(last_date.strftime("%j"))

if days==366:
    print(f'{num} is leap year')
else:
    print(f'{num} is not a leap year')
```

2022

2022 is not a leap year

11. Write a Python program to convert a string to datetime. Go to the editor

○ Sample String : Jan 1 2014 2:43PM ○ Expected Output : 2014-07-01 14:43:00

```
In [37]: import datetime
d1="Jan 1 2014 2:43 PM"
print("original date:",d1)
std_d1=datetime.datetime.strptime(d1,"%b %d %Y %I:%M %p").strftime("%Y-%m-%d %H:%M:%S")
print("updated date:",std_d1)
```

original date: Jan 1 2014 2:43 PM

updated date: 2014-01-01 14:43:00

12. Write a Python program to subtract five days from current date

```
In [1]: import datetime
current_date=datetime.datetime.today()
print(current_date)
five_days=current_date-datetime.timedelta(days=5)
print(five_days)

# a=five_days-current_date
# print("subtract five days from current date:",a)
```

2022-08-02 14:45:02.464118

2022-07-28 14:45:02.464118

13. Write a Python program to print yesterday, today, tomorrow

```
In [5]: today=datetime.datetime.today()
print("today:",today.date())
tomorrow=today+datetime.timedelta(days=1)
print("tomorrow:",tomorrow.date())
yesterday=today+datetime.timedelta(days=-1)
print("yesterday:",yesterday.date())
```

```
today: 2022-08-02
tomorrow: 2022-08-03
yesterday: 2022-08-01
```

14. Write a Python program to print next 5 days starting from today

```
In [7]: current_date=datetime.datetime.today()
print("current_date:",current_date)
for i in range(1,6):
    next_five=current_date+datetime.timedelta(days=i)
    print("next_five:",next_five )
```

```
current_date: 2022-08-02 14:55:26.565786
next_five: 2022-08-03 14:55:26.565786
next_five: 2022-08-04 14:55:26.565786
next_five: 2022-08-05 14:55:26.565786
next_five: 2022-08-06 14:55:26.565786
next_five: 2022-08-07 14:55:26.565786
```

15. Write a Python program to drop microseconds from DateTime

```
In [8]: today=datetime.datetime.today()
print("today:",today)
n_today=today.replace(microsecond=0)
print(n_today)
```

```
today: 2022-08-02 14:58:01.018513
2022-08-02 14:58:01
```

16. Write a Python program to find the date of the first Monday of a given week

```
In [15]: import datetime

date=datetime.datetime.strptime("2022-30-01", "%Y-%U-%W")
print('date of the first monday:',date.date())
```

```
date of the first monday: 2022-01-01
```

17. Write a Python program to select all the Sundays of a specified year


```
In [19]: #U=week number in year
#w=day number in week
import datetime
first_sunday=datetime.datetime.strptime("2022-1-0", "%Y-%U-%w")
for i in range(52):
    print((first_sunday+datetime.timedelta(weeks=i)).date())
```

2022-01-02
2022-01-09
2022-01-16
2022-01-23
2022-01-30
2022-02-06
2022-02-13
2022-02-20
2022-02-27
2022-03-06
2022-03-13
2022-03-20
2022-03-27
2022-04-03
2022-04-10
2022-04-17
2022-04-24
2022-05-01
2022-05-08
2022-05-15
2022-05-22
2022-05-29
2022-06-05
2022-06-12
2022-06-19
2022-06-26
2022-07-03
2022-07-10
2022-07-17
2022-07-24
2022-07-31
2022-08-07
2022-08-14
2022-08-21
2022-08-28
2022-09-04
2022-09-11
2022-09-18
2022-09-25
2022-10-02
2022-10-09
2022-10-16
2022-10-23
2022-10-30
2022-11-06
2022-11-13
2022-11-20
2022-11-27
2022-12-04
2022-12-11

2022-12-18
2022-12-25

18. Write a Python program to create a file and write some text and rename the file name

In []:

19. What are different methods available in the OS module for creating a directory?

In []:

20. Explain os.listdir() method

In []:

21. What are different methods for removing directories and files in Python?

In []:

22. What are exceptions in Python

if the error occurs in the programme then it leads to the termination of the programme. Exceptions: These are raised when programme contain some error. when error detected it does not stop the execution of programme. it executes only when error occurred

In [20]:

```
try:
    print("python")
    print(class1)
except:
    print("velocity")
```

python
velocity

In [21]:

```
try:
    print("python")
except:
    print("velocity")
```

python

23. What are Built-in exceptions?

In []: The exceptions which are already predefined in the python.

24.What are User-defined Exceptions?

In []: The exceptions created by programmer

25.When would you not use try-except?

Generally you use try/except when you handle things that are outside of the parameters that you can influence. A try block allows you to handle an expected error. The except block should only catch exceptions you are prepared to handle. If you handle an unexpected error, your code may do the wrong thing and hide bugs. Here, you can not use try/except

26.Can try-except catch the error if a file can't be opened?

Yes, Can try-except catch the error if a file can't be opened

```
In [22]: def FileCheck(fn):  
        try:  
            open(fn, "r")  
            return 1  
        except IOError:  
            print("Error: File does not appear to exist.")  
            return 0  
result=FileCheck("testfile")  
print(result)
```

```
Error: File does not appear to exist.  
0
```