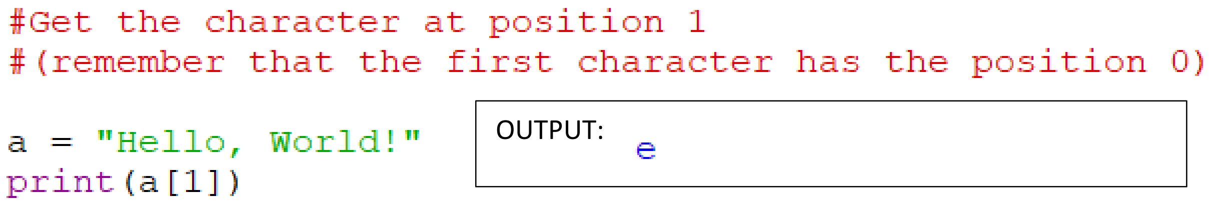
Practical Learning #4
Python Strings, Date and Time

Python Strings

- String literals in python are surrounded by either single quotation marks, or double quotation marks.
- strings in Python are arrays of bytes representing unicode characters. However, Python does not have a character data type, a single character is simply a string with a length of 1.
- Square brackets can be used to access elements of the string.

1. Create a new python file name it as PL4_LastnameFirstname.
2. Try and type the following codeL
3.

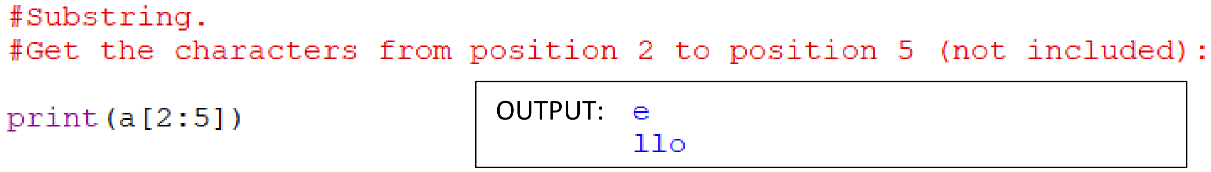


```
#Get the character at position 1
#(remember that the first character has the position 0)

a = "Hello, World!"
print(a[1])
```

OUTPUT:

```
e
```

4.

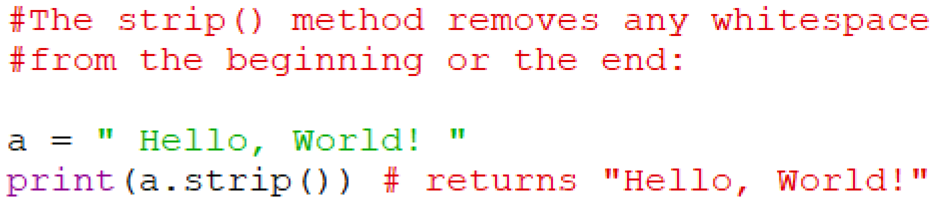


```
#Substring.
#Get the characters from position 2 to position 5 (not included):

print(a[2:5])
```

OUTPUT:

```
e
llo
```

5.



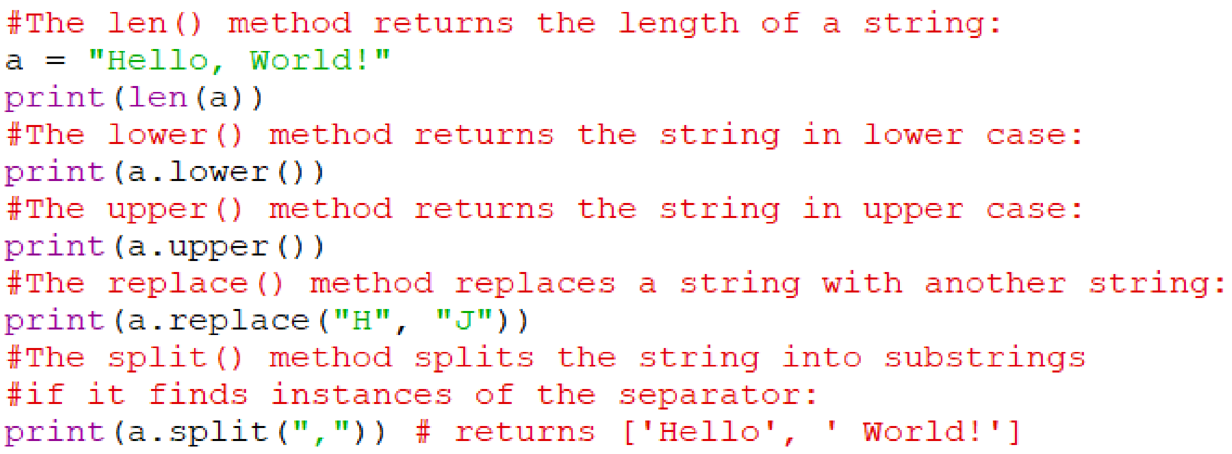
```
#The strip() method removes any whitespace
#from the beginning or the end:

a = " Hello, World! "
print(a.strip()) # returns "Hello, World!"
```

OUTPUT:

```
e
llo
Hello, World!
```

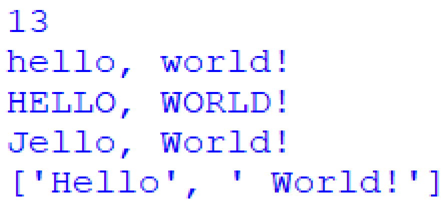
6. Comment codes from #3 to #5.

7.



```
#The len() method returns the length of a string:
a = "Hello, World!"
print(len(a))
#The lower() method returns the string in lower case:
print(a.lower())
#The upper() method returns the string in upper case:
print(a.upper())
#The replace() method replaces a string with another string:
print(a.replace("H", "J"))
#The split() method splits the string into substrings
#if it finds instances of the separator:
print(a.split(",")) # returns ['Hello', ' World!']
```

OUTPUT:

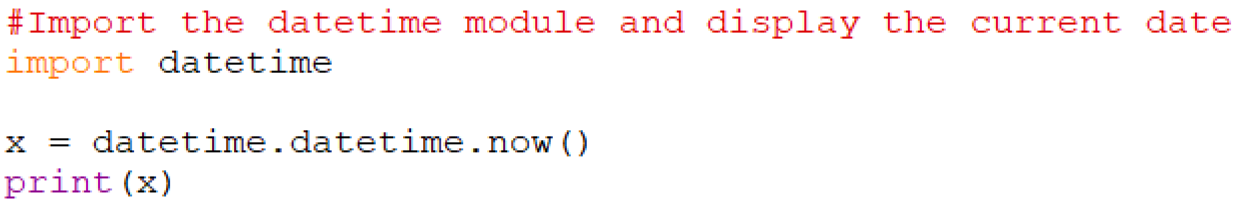


```
13
hello, world!
HELLO, WORLD!
Jello, World!
['Hello', ' World!']
```

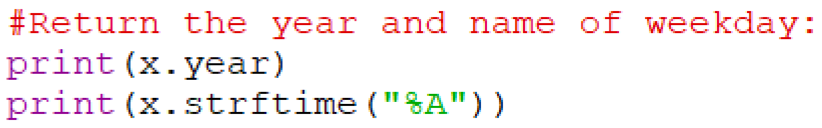
Python Date and Time

- A date in Python is not a data type of its own, but we can import a module named datetime to work with dates as date objects.
- The date contains year, month, day, hour, minute, second, and microsecond.
- The datetime module has many methods to return information about the date object.

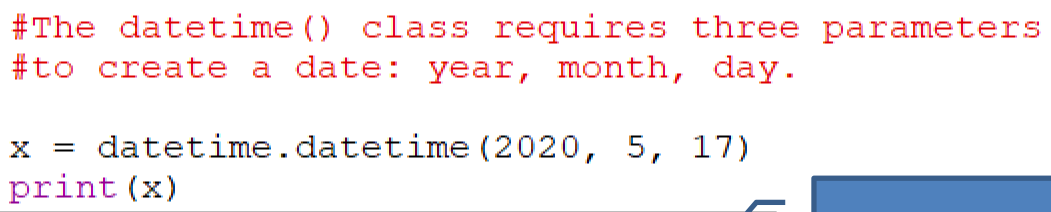
8.



```
#Import the datetime module and display the current date
import datetime

x = datetime.datetime.now()
print(x)
```

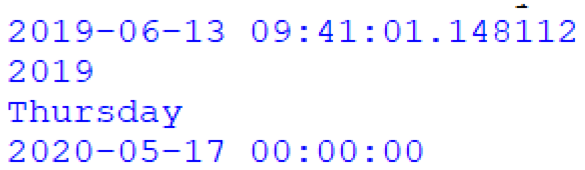


```
#Return the year and name of weekday:
print(x.year)
print(x.strftime("%A"))
```



```
#The datetime() class requires three parameters
#to create a date: year, month, day.

x = datetime.datetime(2020, 5, 17)
print(x)
```

OUTPUT:



```
2019-06-13 09:41:01.148112
2019
Thursday
2020-05-17 00:00:00
```

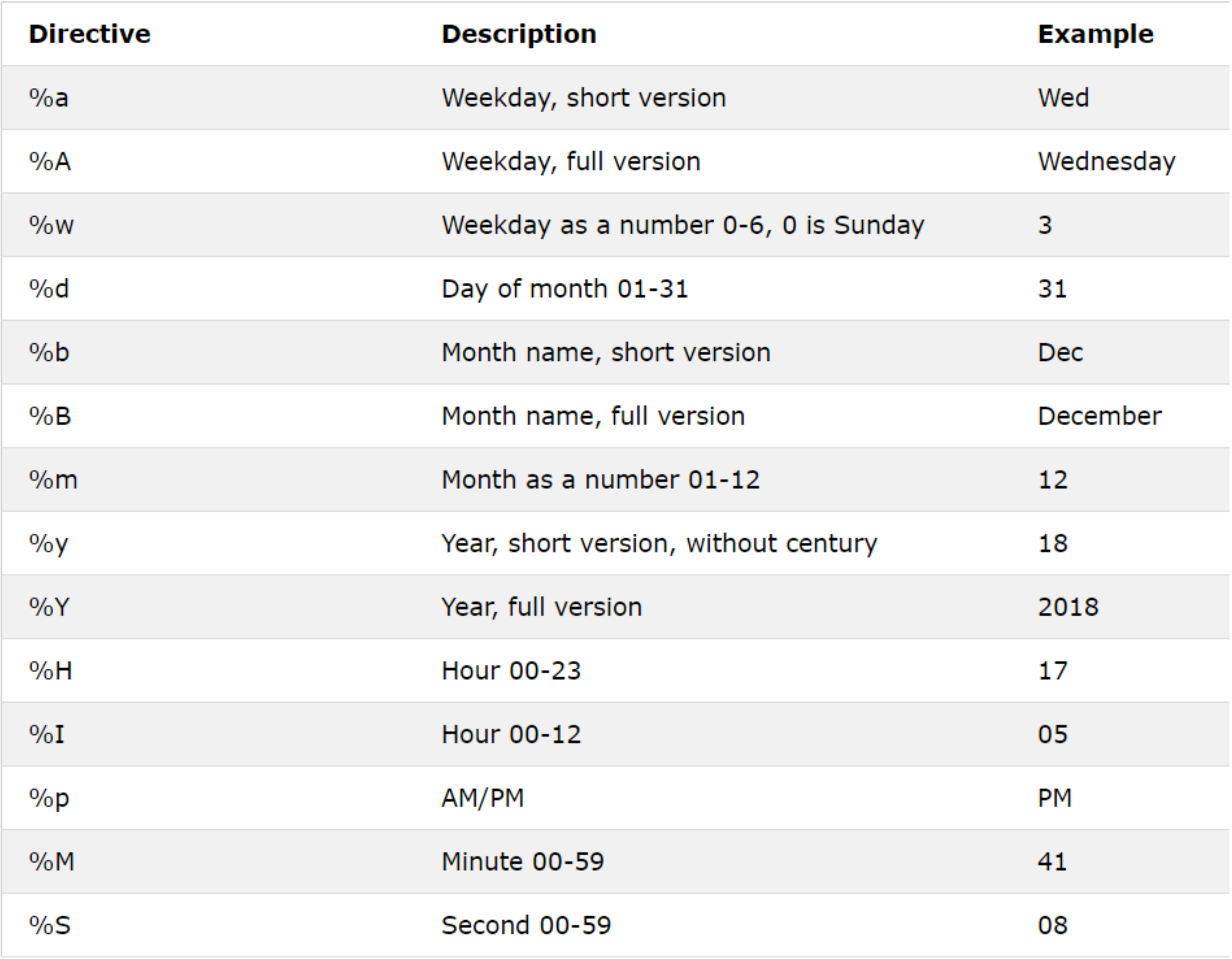
OUTPUT VARIES

The datetime() class also takes parameters for time and timezone (hour, minute, second, microsecond, tzone), but they are optional, and has a default value of 0, (None for timezone).

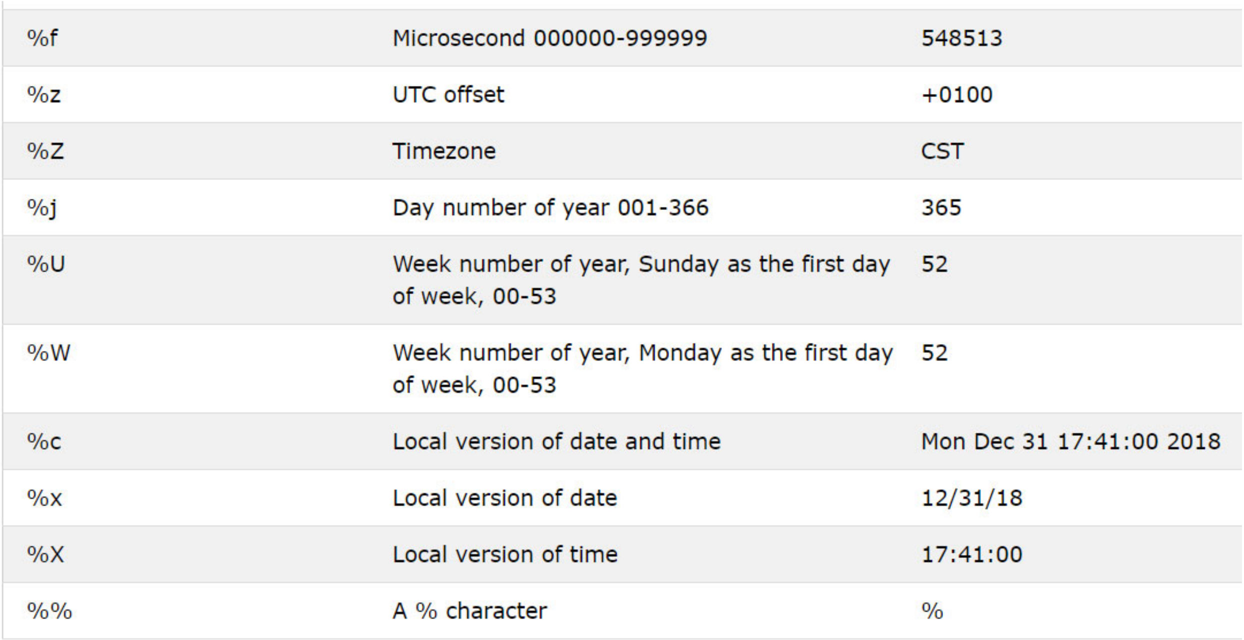
The datetime object has a method for formatting date objects into readable strings.

The method is called strftime(), and takes one parameter, format, to specify the format of the returned string.

9. You may try the following format codes:



Directive	Description	Example
%a	Weekday, short version	Wed
%A	Weekday, full version	Wednesday
%w	Weekday as a number 0-6, 0 is Sunday	3
%d	Day of month 01-31	31
%b	Month name, short version	Dec
%B	Month name, full version	December
%m	Month as a number 01-12	12
%y	Year, short version, without century	18
%Y	Year, full version	2018
%H	Hour 00-23	17
%I	Hour 00-12	05
%p	AM/PM	PM
%M	Minute 00-59	41
%S	Second 00-59	08



%f	Microsecond 000000-999999	548513
%z	UTC offset	+0100
%Z	Timezone	CST
%j	Day number of year 001-366	365
%U	Week number of year, Sunday as the first day of week, 00-53	52
%W	Week number of year, Monday as the first day of week, 00-53	52
%c	Local version of date and time	Mon Dec 31 17:41:00 2018
%x	Local version of date	12/31/18
%X	Local version of time	17:41:00
%%	A % character	%

10. Upload your work.