from pathlib import Path # Here “P” should be capitalized

Path.cwd()

os.chdir()

Path.home()

os.makedirs()

**Dictionary Functions**

someDic.item() # will get all keys & values in dic

someDic.values() or keys() # will get values or keys to corresponding keys or values

**The get() method**

**someDic.get(‘somekey’, “fall back value”)**

**# this methods first checks if the key exits in Dic**

1. **If yes then it returns its value**
2. **If not then it return fall back value.**

**Pretty printing**

pprint.pprint(somedic) # will print Dic tems on every new line

print(pprint.pformat(somedic) # Will convert Dic into A single string

**The Setdefault() method**

**someDic.get(‘somekey’, “default value”)**

**# this methods first checks if the key exits in Dic**

1. **If yes then it returns its value**
2. **If not then it adds the key & the default value to it. Then returns Default value.**

**6. Manipulating String**

**The upper(), lower(), isupper() & islower() method :-**

veriable\_or\_string.upper or lower()

veriable\_or\_string.isupper or islower()

**The isx() method:-**

isalpha() Returns True if the string consists only of letters and isn’t blank

isalnum() Returns True if the string consists only of letters and numbers and is not blank

isdecimal() Returns True if the string consists only of numeric characters and is not blank

isspace() Returns True if the string consists only of spaces, tabs, and newlines and is not blank

istitle() Returns True if the string consists only of words that begin with an uppercase letter followed by only lowercase letters

Use like – “veriable\_or\_string.isx()

**The strartswith() & endswith() method:-**

veriable\_or\_string.startswith()\_or\_endswith()

**The join() & split() method:-**

String.join([some\_list])

String.split(‘string to separate’) # The “string” used to separate will be erased.

Splitting string with partition method:-

>>> 'Hello, world!'.partition('w')

***Output-***('Hello, ', 'w', 'orld!')

>>> 'Hello, world!'.partition('XYZ')

***output-*** ('Hello, world!', '', '') # xyz isn’t in given string

Justifying Text with the rjust(), ljust(), and center() Methods

>>> 'Hello'.rjust(10)

***output-*** ' Hello'

>>> 'Hello'.ljust(10)

***output-*** 'Hello '

>>> 'Hello'.rjust(20, '\*')

'\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Hello'

Removing Whitespace with the strip(), rstrip(), and lstrip() Methods

string or variable containing string.strip() # Will strip white space from both side of string (start & end) because there is no value in bracket.

Another example

>>> spam = 'SpamSpamBaconSpamEggsSpamSpam'

>>> spam.strip('ampS')

'BaconSpamEggs'

Numeric Values of Characters with the ord() and chr() Functions

>>> ord('A')

65

>>> chr(65)

'A'

Copying and Pasting Strings with the pyperclip Module

>>> import pyperclip

>>> pyperclip.copy('Hello, world!')

>>> pyperclip.paste()

'Hello, world!'

**7 PATTERN MATCHING WITH REGULAR EXPRESSIONS**