In [ ]:

```
                    Assignment 9 Solutions
```

In [ ]:

```
1.To what does a relative path refer ?
Answer:-  The relative path is the path to some file with respect to your curr
          working directory (PWD).
For example: if Absolute path to a file called stuff.txt is:
        C:/users/admin/docs/stuff.txt If my PWD is C:/users/admin/ , then the
        docs/stuff.txt
     Note: PWD + relative path = absolute path
```

In [ ]:

```
2.Where does an absolute path start with your Operating System ?
Answer :- In Linux based systems the absolute path starts with /.
         Where as in Windows based systems absolute path starts with C:
```

In [3]:

```
3.What does the functions os.getcwd() and os.chdir() do ?
Answer :- os.getcwd() method tells us the location of current working director
        Whereas os.chdir() method in Python used to change the current working
        to specified path. These functions are similar to linux commands pwd a

#Example
import os
print(os.getcwd()) # Prints the current Working Directory
path = r'C:\Users\Rohini.basic assignment\Documents'
os.chdir(path)
print(os.getcwd())
```

```
C:\Users\Rohini

---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
Input In [3], in <cell line: 4>()
      2 print(os.getcwd()) # Prints the current Working Directory
      3 path = r'C:\Users\Rohini.basic assignment\Documents'
----> 4 os.chdir(path)
      5 print(os.getcwd())

FileNotFoundError: [WinError 3] The system cannot find the path specified: '
C:\\Users\\Rohini.basic assignment\\Documents'
```

In [ ]:

```
4.What are . and .. folders ?
Answer :-. Represents the Current Directory Whereas .. Represents the Parent D
          of the Current Directory
    For Example: if the below path is my absolute path:
    C:\\Users\\rohini\\Documents\\iNeuron-Assignments\\Python Basic Assignment
    Then . represents the path C:\\Users\\vishnu\\Documents\\iNeuron-Assignmen
    Basic Assignment
    Where as .. represents the path C:\\Users\\rohini\\Documents\\iNeuron-Assi
```

In [4]:

```
5.In C:\bacon\eggs\spam.txt which part is the dir name and which part is the b
Ans: For C:\bacon\eggs\spam.txt
    The dir name is C:\\bacon\\eggs
    The Base name is spam.txt
```

```
#Example
import os
path = r'C:\bacon\eggs\spam.txt'
print(os.path.dirname(path))
print(os.path.basename(path))
```

```
C:\bacon\eggs
spam.txt
```

```
In [ ]: 6.What are the three mode arguments that can be passed to the open() function
        Answer: A file can be Accessed in python using open() function. open function
            two arguments filename and mode of operation (optional). if mode is not pr
            the default mode of opening is read mode
            So, the syntax being: open(filename, mode)

            1) ‘r’ – Read Mode: This is the default mode for open(). The file is opene
                pointer is positioned at the beginning of the file’s content.
            2) ‘w’ – Write Mode: Using this mode will overwrite any existing content i
                If the given file does not exist, a new one will be created.
            3) ‘r+’ – Read/Write Mode: Use this mode if you need to simultaneously
                read and write to a file.
            4) ‘a’ – Append Mode: With this mode the user can append the data without
                overwriting any already existing data in the file.
            5) ‘a+’ – Append and Read Mode: In this mode you can read and append the
                data without overwriting the original file.
            6) ‘x’ – Exclusive Creating Mode: This mode is for the sole purpose of cre
                new files. Use this mode if you know the file to be written doesn’t ex
```

```
In [ ]: 7.What happens if an existing file is opened in write mode ?
        Answer: Using this mode will overwrite any existing content in a file.
            If the given file does not exist, a new one will be created.
```

```
In [ ]: 8.How do you tell the difference between read() and readlines() ?
        Answer: The main difference is that read() will read the whole file at
            once and then print out the first characters that take up as many
            bytes as you specify in the parenthesis

            1) Whereas the readline() that will read and print out
               only the first characters that take up as many bytes as you specify
                in the parenthesis. You may want to use readline() when you're reading
                files that are too big for your RAM.
            2) The read() would treat each character in the file separately,
               meaning that the iteration would happen for every character.
            3) The readline() function, on the other hand, only reads a single
                line of the file. This means that if the first line of the file were t
                lines long, the readline() function would only parse (or iterate/opera
                the first line of the file.
```

```
In [ ]: 9.What data structure does a shelf value resemble ?
        Answer:- it contains key and values it represents dictionary.
```