**ADITYA AMIN ASSIGN : 09**

1. To what does a relative path refer?

A relative path is a file or directory path that is specified relative to the current working directory or another reference point, rather than from the root directory of a file system. In other words, it describes the location of a file or directory with respect to the current location or another specified location, rather than providing the full path from the root directory.

1. What does an absolute path start with your operating system?

On Unix-based systems (such as Linux, macOS, and some other Unix-like operating systems), absolute paths start with a forward slash (/). For example:

/home/user/documents/file1.txt

On Windows-based systems (such as Microsoft Windows), absolute paths start with a drive letter followed by a colon (:) and then a backslash (). For example:

C:\Users\user\Documents\file1.txt

1. What do the functions os.getcwd() and os.chdir() do?

The functions os.getcwd() and os.chdir() are Python functions from the os module that are used to interact with the operating system's file system.

os.getcwd(): This function is used to get the current working directory (CWD) of the Python script or process.

os.chdir(path): This function is used to change the current working directory to the specified path. The path parameter is a string that represents the directory path to which the current working directory needs to be changed.

1. What are the . and .. folders?

In file systems, the . and .. folders are special directory references that have specific meanings.

. (dot): In a file system, . (dot) represents the current directory. It is used to refer to the directory in which a file or command is currently being executed. For example, if you are in the directory /home/user/documents/ and you use a relative path of ./file.txt, it refers to the file file.txt in the current directory, which is /home/user/documents/file.txt.

.. (dot dot): In a file system, .. (dot dot) represents the parent directory. It is used to refer to the directory that is one level up in the hierarchy from the current directory. For example, if you are in the directory /home/user/documents/ and you use a relative path of ../pictures/, it refers to the directory pictures in the parent directory, which is /home/user/pictures/.

1. In C:\bacon\eggs\spam.txt, which part is the dir name, and which part is the base name?

Directory Name: C:\bacon\eggs

Base Name: spam.txt

1. What are the three “mode” arguments that can be passed to the open() function?

'r' (Read mode): This mode is used for reading the contents of an existing file. The file must already exist, otherwise a FileNotFoundError will be raised if the file is not found. Example usage: file = open('file.txt', 'r')

'w' (Write mode): This mode is used for creating a new file or overwriting the contents of an existing file. If the file does not exist, it will be created. If the file already exists, its contents will be truncated (i.e., deleted) before writing new data to it. Example usage: file = open('file.txt', 'w')

'a' (Append mode): This mode is used for appending data to the end of an existing file. If the file does not exist, it will be created. The new data will be added to the end of the file, without truncating or overwriting existing data. Example usage: file = open('file.txt', 'a')

1. What happens if an existing file is opened in write mode?
2. If the file does not exist: If the file specified in the open() function does not exist, it will be created as a new empty file.

(b) If the file exists: If the file specified in the open() function already exists, its contents will be deleted, and the file will be treated as empty. Any data that was previously stored in the file will be permanently lost.

8. How do you tell the difference between read() and readlines()?

read() returns a single string representing the entire contents of the file, while readlines() returns a list of strings representing individual lines in the file. read() is useful when you want to treat the entire file as a single string, whereas readlines() is useful when you want to process the file line by line.

1. What data structure does a shelf value resemble?

a shelve value in Python resembles a dictionary-like data structure that provides disk-based storage, serialization, lazy updates, and allows for keys and values of various data types.