1. To what does a relative path refer?

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

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

4. What are the . and .. folders?

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

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

7. What happens if an existing file is opened in write mode?

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

9. What data structure does a shelf value resemble?

**Answer**

1. A relative path refers to a location within the current directory or a location relative to the current working directory. It does not provide the full path from the root of the file system.

2. An absolute path starts with the root directory of the operating system. In Windows, an absolute path often starts with a drive letter and a backslash (e.g., "C:\path\to\file"), while in Unix-based systems, it usually starts with a forward slash (e.g., "/path/to/file").

3. The functions `os.getcwd()` returns the current working directory, which is the directory where your Python script is currently running. `os.chdir()` is used to change the current working directory to the specified path.

4. In file systems, the `.` represents the current directory, and `..` represents the parent directory. They are used to navigate the directory structure. For example, if you're in the directory `/home/user/docs`, `.` refers to `/home/user/docs`, and `..` refers to `/home/user`.

5. In the path "C:\bacon\eggs\spam.txt," the directory name is "C:\bacon\eggs," and the base name is "spam.txt."

6. The three "mode" arguments that can be passed to the `open()` function are:

- `'r'`: Read mode (default) to open a file for reading.

- `'w'`: Write mode to open a file for writing (creates a new file or truncates an existing file).

- `'a'`: Append mode to open a file for writing (creates a new file or appends to an existing file).

7. If an existing file is opened in write mode (`'w'`), it will overwrite the content of the file. Any previous content in the file will be lost.

8. `read()` reads the entire contents of the file as a single string, while `readlines()` reads the lines of the file into a list, where each element of the list represents a line in the file.

9. A shelf value in Python resembles a dictionary-like data structure that stores key-value pairs. It is provided by the `shelve` module and is used for persistent storage of data, similar to a dictionary. It allows you to store and retrieve data using keys, and it persists between program runs.