1. **What advantages do Excel spreadsheets have over CSV spreadsheets?**

While excel (XLS and XLSX) file formats are better for storing more complex data, CSV files are supported by nearly all data upload interfaces. If you are planning to move your data between platforms, export and import it from one interface to another, you might be better off with the CSV file format.

**2.What do you pass to csv.reader() and csv.writer() to create reader and writer objects?**

To read a CSV file in Python, we can use the csv.reader() function. Suppose we have a csv file named **people.csv** in the current directory with the following entries.

### Example 1: Read CSV Having Comma Delimiter

import csv

with open('people.csv', 'r') as file:

reader = csv.reader(file)

for row in reader:

print(row)

**Output**

['Name', 'Age', 'Profession']

['Jack', '23', 'Doctor']

['Miller', '22', 'Engineer']

Here, we have opened the **people.csv** file in reading mode using:

with open('people.csv', 'r') as file:

.. .. ...

To write to a CSV file in Python, we can use the csv.writer() function.

The csv.writer() function returns a writer object that converts the user's data into a delimited string. This string can later be used to write into CSV files using the writerow() function. Let's take an example.

### Example 3: Write to a CSV file

import csv

with open('protagonist.csv', 'w', newline='') as file:

writer = csv.writer(file)

writer.writerow(["SN", "Movie", "Protagonist"])

writer.writerow([1, "Lord of the Rings", "Frodo Baggins"])

writer.writerow([2, "Harry Potter", "Harry Potter"])

When we run the above program, a **protagonist.csv** file is created with the following content:

SN,Movie,Protagonist

1,Lord of the Rings,Frodo Baggins

2,Harry Potter,Harry Potter

In the above program, we have opened the file in writing mode.

Then, we have passed each row as a list. These lists are converted to a delimited string and written into the CSV file.

1. **What modes do File objects for reader and writer objects need to be opened in?**

Here mode means the operation for which the file has to be opened like <r> for reading, <w>for writing <+> for both reading and writing <a> for appending at the eand of an existing file the default is the read mode.

1. **What method takes a list argument and writes it to a CSV file?**

CSV files a type of a plain text document in which tabular information is structured using a particular format. A CSV file is a bounded text format which uses a common to separate values. The most common method to write data from a list to CSV file is the write row() method of writer and dictwriter class,

Example 1: creating a CSV file and writing data row-wise in to it using writer class.

# Importing library

import csv

# data to be written row-wise in csv file

data = [['Geeks'], [4], ['geeks !']]

# opening the csv file in 'w+' mode

file = open('g4g.csv', 'w+', newline ='')

# writing the data into the file

with file:

write = csv.writer(file)

write.writerows(data)

1. **What do the keyword arguments delimiter and line terminator do?**

This changes the delimiter and line terminator characters in your file. The delimiter is the character that appears between cells on a row. By default, the delimiter for a CSV file is a comma. The line terminator is the character that comes at the end of a row. By default, the line terminator is a newline. It can change characters to different values by using the delimiter and line terminator keyword arguments with csv.writer().

1. **What function takes a string of JSON data and returns a Python data structure?**

**to convert JSON data into a custom Python object**. i.e., Parse and convert JSON into Python Class. For example, receive employee JSON data from the API or you are reading JSON from a file and wanted to convert it into a custom Employee type.

**7. What function takes a Python data structure and returns a string of JSON data?**

If you have a Python object, you can convert it into a JSON string by using the json.dumps() method.

### Example

Convert from Python to JSON:

import json  
# a Python object (dict):  
x = {  
  "name": "John",  
  "age": 30,  
  "city": "New York"  
}  
  
# convert into JSON:  
y = json.dumps(x)  
# the result is a JSON string:  
print(y)