**ADITYA AMIN ASSIGN : 13**

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

Excel spreadsheets offer more advanced features for data formatting, data analysis, data visualization, collaboration, data validation, and automation, making them a more powerful tool for managing and analyzing data compared to basic CSV files.

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

To create reader and writer objects using the csv module in Python, you typically pass file objects to csv.reader() and csv.writer() functions.

import csv

# Example of creating a reader object

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

csv\_reader = csv.reader(file)

for row in csv\_reader:

print(row)

import csv

# Example of creating a writer object

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

csv\_writer = csv.writer(file)

csv\_writer.writerow(['Name', 'Age', 'City'])

csv\_writer.writerow(['John', 30, 'New York'])

csv\_writer.writerow(['Alice', 25, 'Los Angeles'])

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

In Python, when working with file objects using the open() function, there are different modes that can be used to specify how the file should be opened. For reader objects, which are used for reading from a file, the common modes are:

'r': Read mode, which opens the file for reading. If the file does not exist, it raises a FileNotFoundError. This is the default mode when no mode is specified explicitly.

'r+': Read and write mode, which opens the file for both reading and writing. If the file does not exist, it raises a FileNotFoundError.

'rb': Binary read mode, which opens the file for reading binary data. This mode is used for reading non-text files like images, audio files, etc.

For writer objects, which are used for writing to a file, the common modes are:

'w': Write mode, which opens the file for writing. If the file already exists, it will be truncated (i.e., its contents will be deleted). If the file does not exist, it will be created.

'w+': Write and read mode, which opens the file for both writing and reading. If the file already exists, it will be truncated. If the file does not exist, it will be created.

'wb': Binary write mode, which opens the file for writing binary data. This mode is used for writing non-text files like images, audio files, etc.

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

In Python, you can use the csv module to write a list to a CSV (Comma-Separated Values) file. Here's an example method that takes a list argument and writes it to a CSV file:

import csv

def write\_list\_to\_csv(filename, data):

"""

Writes a list to a CSV file.

Args:

filename (str): The name of the CSV file.

data (list): The list to be written to the CSV file.

"""

with open(filename, 'w', newline='') as csvfile:

writer = csv.writer(csvfile)

writer.writerow(['Data']) # Optional: write a header row

writer.writerows(data)

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

The keyword arguments delimiter and line terminator are commonly used in programming to specify characters or strings that are used as separators or terminators when processing data.

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

The json module in Python provides functions for encoding and decoding JSON data. The json.loads() function can be used to parse a string containing JSON data and convert it into a Python data structure.

import json

# Example JSON string

json\_string = '{"name": "John", "age": 30, "city": "New York"}'

# Parse JSON string to Python data structure

data = json.loads(json\_string)

# Access data as a Python dictionary

print(data['name']) # Output: John

print(data['age']) # Output: 30

print(data['city']) # Output: New York

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

json.dumps() function can be used to convert a Python data structure into a JSON formatted string. Example :

import json

data = ... # Your Python data structure

json\_string = json.dumps(data)