

## Chapter 6: Advanced Input and Output (Page 98)

## :Multiple Choice (MCQs)

1. csv (b)

2. (b) Dictionary for each row with column names as keys (Section 6.2.1)

3. (c) json.dumps() (Section 6.3.2)

4. (b) Reads only the specified sheet into a DataFrame (Section 6.4.1)

5. (b) openpyxl (Section 6.4)

:True/False

1. False (CSV reads data as strings by default)

2. True (json.dump writes to a file object)

3. True (Pandas supports both)

4. True

False (It typically requires an engine like openpyxl or xlsxwriter).

```
# =====
# Chapter 6: Advanced Input/Output
# =====
print("\n--- Chapter 6 Solutions ---")

# 1. CSV Handling
print("CSV Output:")
# Mocking file content for demonstration
csv_content = "ID,Name,Grade\n1,Ali,85\n2,Mona,92\n3,Omar,78"
reader = csv.DictReader(StringIO(csv_content))
for row in reader:
    if int(row['Grade']) > 80:
        print(row['Name'])

# 2. JSON Handling
```

```
data = {"course": "Python", "duration": "3 months", "students": ["Ali",  
"Sara"]}  
json_str = json.dumps(data)  
# Simulate read back  
loaded = json.loads(json_str)  
print(f"JSON Students: {loaded['students']}")  
  
# 4. Data Transformation  
def csv_to_json(csv_data):  
    people = []  
    reader = csv.DictReader(StringIO(csv_data))  
    for row in reader:  
        people.append({  
            "Name": row["Name"], "Age": int(row["Age"]), "City": row["City"]  
        })  
    return json.dumps({"people": people})  
  
csv_in = "Name,Age,City\nAli,25,Cairo\nMona,30,Alex"  
print(f"Transformed JSON: {csv_to_json(csv_in)}")
```