Unit 6 Lesson 1: Reading CSV Files

Quick Reference Guide

📦 Import

```
import csv
```

No installation needed — `csv` is built into Python!

📖 Reading with DictReader (Recommended)

```
with open("players.csv", "r") as file:
    reader = csv.DictReader(file)
    for row in reader:
        print(row['username'])
```

- Each row becomes a **dictionary**
- Keys = column headers from first row
- Access values by column name: `row['column_name']`

📦 Loading All Data into a List

```python
with open("players.csv", "r") as file:
    reader = csv.DictReader(file)
    players = list(reader)

# Now use it after file closes
print(len(players))
print(players[0]['username'])
```

Or the shorthand:

```python
with open("players.csv", "r") as file:
    players = list(csv.DictReader(file))
```

⚠️ Key Rules

Rule	Why It Matters
All values are strings	`"25"` not `25` — convert with `int()` or `float()`
Headers are automatic	DictReader uses row 1 as keys, no skip needed
File closes after `with`	Load into a list if you need data later
Column names are case-sensitive	`row['Level']` ≠ `row['level']`

🔄 Converting Types

```python
# ❌ Wrong - comparing strings
if row['level'] > 10:  # "25" > 10 doesn't work right!

# ✅ Correct - convert first
if int(row['level']) > 10:
```

Common conversions:

```
level = int(row['level'])      # String to integer
price = float(row['price'])    # String to decimal
name = row['username']         # Keep as string
```

🔍 Finding a Record

```
def find_player(players, username):
    for player in players:
        if player['username'].lower() == username.lower():
            return player
    return None
```

🎯 Filtering Data

```
# Get players level 20+
high_level = []
for player in players:
    if int(player['level']) >= 20:
        high_level.append(player)
```

Or with list comprehension:

```
high_level = [p for p in players if int(p['level']) >= 20]
```

📊 Calculating Statistics

```
# Total
total_gold = sum(int(p['gold']) for p in players)

# Average
avg_level = sum(int(p['level']) for p in players) / len(players)

# Maximum
top_player = max(players, key=lambda p: int(p['wins']))
```

🛡️ Error Handling

```
try:
    with open("players.csv", "r") as file:
        players = list(csv.DictReader(file))
except FileNotFoundError:
    print("File not found!")
    players = []
```

📋 csv.reader vs csv.DictReader

Feature	csv.reader	csv.DictReader
Row format	List `['val1', 'val2']`	Dict `{'col': 'val'}`
Access data	By index `row[0]`	By name `row['username']`
Headers	Must skip manually	Automatic
Best for	Simple/quick scripts	Most use cases ⭐

🗂️ Our Data File: players.csv

Column	Type	Description
username	string	Player's display name
level	int	Current level (1-50)
xp	int	Experience points
gold	int	In-game currency
wins	int	Total victories