**Upgraded Python Mini Project: Quiz App with Collections**

**🔧 Concepts Used:**

* **List**: To store all questions
* **Dictionary**: To store each question, options, and answer
* **Tuple**: For fixed options (for bonus explanation)

**✅ Step-by-Step Code**

python

Copy code

# Step 1: Define quiz questions using a list of dictionaries

questions = [

{

"question": "What is the capital of France?",

"options": ("Paris", "London", "Berlin", "Rome"), # Tuple for fixed options

"answer": "Paris"

},

{

"question": "What is 2 + 2?",

"options": ("3", "4", "5", "6"),

"answer": "4"

},

{

"question": "What color is the sky on a clear day?",

"options": ("Blue", "Green", "Red", "Yellow"),

"answer": "Blue"

}

]

**✅ Function to Run the Quiz**

python

Copy code

def run\_quiz():

score = 0

for i, q in enumerate(questions):

print(f"\nQuestion {i + 1}: {q['question']}")

# Display options using a loop

for index, option in enumerate(q["options"], start=1):

print(f"{index}. {option}")

# Get user input and validate

try:

choice = int(input("Choose the correct option (1-4): "))

selected\_option = q["options"][choice - 1]

if selected\_option.lower() == q["answer"].lower():

print("✅ Correct!")

score += 1

else:

print("❌ Wrong. The correct answer was:", q["answer"])

except (ValueError, IndexError):

print("⚠️ Invalid input. Skipping this question.")

print("\n🎯 Final Score:", score, "/", len(questions))

**✅ Run the App**

python

Copy code

print("🧠 Welcome to the Quiz App with Collections!")

run\_quiz()

**📘 Explanation of Collections Used:**

| **Collection** | **Used For** | **Example** |
| --- | --- | --- |
| list | Holding multiple questions | questions = [ {...}, {...} ] |
| dict | Each question, answer, options | {"question": ..., "answer": ...} |
| tuple | Fixed choices (options) | ("Paris", "London", ...) |

**💡 Extension Ideas:**

* Store questions in a **dictionary by topic**
* Randomize question order using random.shuffle()
* Store scores in a **list** and display best score