### **Project 2: Personal Finance Tracker**

#### **Objective:**

Create a command-line personal finance tracker that allows users to record daily expenses and savings. The program should calculate the monthly total and provide a breakdown of expenses by category (e.g., food, transport, entertainment).

### **Instructions**

#### **Step 1: Set Up the Environment**

1. Create a new Python file called personal\_finance\_tracker.py.
2. Open the file in your preferred IDE or text editor.

#### **Step 2: Create a Function to Add Expenses and Savings**

1. Define a function called add\_transaction() that accepts two parameters: type (either "expense" or "saving") and amount.

Use a dictionary to categorize expenses and savings:  
python  
Copy code  
transactions = {

"expenses": {},

"savings": 0

}

1. Inside the function, prompt the user to enter a category for expenses and store the amount under that category. If the type is "saving," simply add the amount to transactions["savings"].

Example:  
python  
Copy code  
def add\_transaction(type, amount):

if type == "expense":

category = input("Enter the category (e.g., food, transport, entertainment): ").lower()

if category in transactions["expenses"]:

transactions["expenses"][category] += amount

else:

transactions["expenses"][category] = amount

elif type == "saving":

transactions["savings"] += amount

#### **Step 3: Create a Function to Display the Report**

1. Define a function called display\_report() that prints a summary of expenses and savings for the current month.
2. Iterate through transactions["expenses"] to display each category and its corresponding total amount.

Display the total amount of savings and the overall balance (savings minus expenses).  
python  
Copy code  
def display\_report():

print("\n--- Monthly Report ---")

total\_expenses = sum(transactions["expenses"].values())

print("Expenses by category:")

for category, amount in transactions["expenses"].items():

print(f"- {category.capitalize()}: ${amount:.2f}")

print(f"Total Expenses: ${total\_expenses:.2f}")

print(f"Total Savings: ${transactions['savings']:.2f}")

print(f"Net Balance: ${transactions['savings'] - total\_expenses:.2f}")

#### **Step 4: Create a Function to Save and Load Data**

Create a function called save\_to\_file() that saves transactions to a text file called finance\_data.txt using basic file handling.  
python  
Copy code  
def save\_to\_file():

with open("finance\_data.txt", "w") as file:

for category, amount in transactions["expenses"].items():

file.write(f"{category}:{amount}\n")

file.write(f"savings:{transactions['savings']}")

Create a function called load\_from\_file() that reads the data from finance\_data.txt (if it exists) and populates the transactions dictionary.  
python  
Copy code  
def load\_from\_file():

try:

with open("finance\_data.txt", "r") as file:

for line in file:

key, value = line.strip().split(":")

if key == "savings":

transactions["savings"] = float(value)

else:

transactions["expenses"][key] = float(value)

except FileNotFoundError:

pass # If no file exists, start with an empty dictionary

#### **Step 5: Create a Main Menu**

1. Define a main function called main() that serves as the entry point to the program.
2. Display a menu with options such as:
   * Add an Expense
   * Add a Saving
   * Display Report
   * Save Data
   * Load Data
   * Exit

Based on the user's selection, call the corresponding function.  
python  
Copy code  
def main():

load\_from\_file()

while True:

print("\n--- Personal Finance Tracker ---")

print("1. Add an Expense")

print("2. Add a Saving")

print("3. Display Report")

print("4. Save Data")

print("5. Load Data")

print("6. Exit")

choice = input("Enter your choice: ")

if choice == "1":

amount = float(input("Enter the expense amount: "))

add\_transaction("expense", amount)

elif choice == "2":

amount = float(input("Enter the saving amount: "))

add\_transaction("saving", amount)

elif choice == "3":

display\_report()

elif choice == "4":

save\_to\_file()

print("Data saved successfully!")

elif choice == "5":

load\_from\_file()

print("Data loaded successfully!")

elif choice == "6":

print("Exiting the program. Have a great day!")

break

else:

print("Invalid choice. Please select an option from the menu.")

#### **Step 6: Run the Program**

Add a line at the end of the script to ensure main() runs when the script is executed:  
python  
Copy code  
if \_\_name\_\_ == "\_\_main\_\_":

main()

#### **Step 7: Final Code**

The entire code should look like this:

python

Copy code

# Step 1: Create the dictionary to store transactions

transactions = {

"expenses": {},

"savings": 0

}

# Step 2: Function to add a transaction

def add\_transaction(type, amount):

if type == "expense":

category = input("Enter the category (e.g., food, transport, entertainment): ").lower()

if category in transactions["expenses"]:

transactions["expenses"][category] += amount

else:

transactions["expenses"][category] = amount

elif type == "saving":

transactions["savings"] += amount

# Step 3: Function to display a report

def display\_report():

print("\n--- Monthly Report ---")

total\_expenses = sum(transactions["expenses"].values())

print("Expenses by category:")

for category, amount in transactions["expenses"].items():

print(f"- {category.capitalize()}: ${amount:.2f}")

print(f"Total Expenses: ${total\_expenses:.2f}")

print(f"Total Savings: ${transactions['savings']:.2f}")

print(f"Net Balance: ${transactions['savings'] - total\_expenses:.2f}")

# Step 4: Function to save data to a file

def save\_to\_file():

with open("finance\_data.txt", "w") as file:

for category, amount in transactions["expenses"].items():

file.write(f"{category}:{amount}\n")

file.write(f"savings:{transactions['savings']}")

# Step 5: Function to load data from a file

def load\_from\_file():

try:

with open("finance\_data.txt", "r") as file:

for line in file:

key, value = line.strip().split(":")

if key == "savings":

transactions["savings"] = float(value)

else:

transactions["expenses"][key] = float(value)

except FileNotFoundError:

pass # If no file exists, start with an empty dictionary

# Step 6: Main function to run the program

def main():

load\_from\_file()

while True:

print("\n--- Personal Finance Tracker ---")

print("1. Add an Expense")

print("2. Add a Saving")

print("3. Display Report")

print("4. Save Data")

print("5. Load Data")

print("6. Exit")

choice = input("Enter your choice: ")

if choice == "1":

amount = float(input("Enter the expense amount: "))

add\_transaction("expense", amount)

elif choice == "2":

amount = float(input("Enter the saving amount: "))

add\_transaction("saving", amount)

elif choice == "3":

display\_report()

elif choice == "4":

save\_to\_file()

print("Data saved successfully!")

elif choice == "5":

load\_from\_file()

print("Data loaded successfully!")

elif choice == "6":

print("Exiting the program. Have a great day!")

break

else:

print("Invalid choice. Please select an option from the menu.")

# Step 7: Ensure main() runs when the script is executed

if \_\_name\_\_ == "\_\_main\_\_":

main()