

## Expense Tracker Program Documentation

---

### Overview

This program is designed to help users manage their expenses efficiently. It allows users to add new expenses, remove existing expenses, and list all expenses. The program uses a simple menu-based interface to interact with the user.

### Code Explanation

---

#### 1. Initialization:

---

```
expenses = []  
  
expense1 = {'amount': '51.00', 'category': 'shirt'}  
  
expenses.append(expense1)  
  
expense2 = {'amount': '21.55', 'category': 'groceries'}  
  
expenses.append(expense2)
```

---

☐ expenses is an empty list that will store all expense entries.

☐ Two initial expenses are added to the list for demonstration purposes.

Expense Tracker Program Documentation

Overview

This program is designed to help users manage their expenses efficiently. It allows users to add new expenses, remove existing expenses, and list all expenses. The program uses a simple menu-based interface to interact with the user.

### Code Explanation

#### 1. Initialization:

```
expenses = []  
  
expense1 = {'amount': '51.00', 'category': 'shirt'}  
  
expenses.append(expense1)  
  
expense2 = {'amount': '21.55', 'category': 'groceries'}  
  
expenses.append(expense2)
```

- `expenses` is an empty list that will store all expense entries.

- Two initial expenses are added to the list for demonstration purposes.

## 2. Functions:

```
def removeExpense():  
    while True:  
        listExpenses()  
        print("What expense would you like to remove?")  
        try:  
            expenseToRemove = int(input("> "))  
            del expenses[expenseToRemove]  
            break  
        except (IndexError, ValueError):  
            print("Invalid input. Please try again.")
```

- This function allows the user to remove an expense from the list.
- It lists all expenses and prompts the user to select which expense to remove by its index.
- It handles invalid input gracefully by catching exceptions.

### - addExpense:

```
def addExpense(amount, category):  
    expense = {'amount': amount, 'category': category}  
    expenses.append(expense)  
    ...
```

- This function adds a new expense to the list with the specified `amount` and `category`.

### - printMenu:

```
def printMenu():  
    print("Please choose from one of the following options...")  
    print("1. Add A New Expense")  
    print("2. Remove An Expense")  
    print("3. List All Expenses")
```

- This function displays the main menu with three options for the user.

- **\*\*listExpenses:\*\***

```
```python
```

```
def listExpenses():
```

```
    print("\nHere is a list of your expenses...")
```

```
    print("-----")
```

```
    for counter, expense in enumerate(expenses):
```

```
        print("#", counter, " - ", expense['amount'], " - ", expense['category'])
```

```
    print("\n")
```

```
```
```

- This function lists all current expenses with their indices, amounts, and categories.

### 3. Main Loop:

```
if __name__ == "__main__":
```

```
    while True:
```

```
        printMenu()
```

```
        optionSelected = input("> ")
```

```
        if optionSelected == "1":
```

```
            print("How much was this expense?")
```

```
            while True:
```

```
                try:
```

```
                    amountToAdd = float(input("> "))
```

```
                    break
```

```
                except ValueError:
```

```
                    print("Invalid input. Please try again.")
```

```
            print("What category was this expense?")
```

```
            category = input("> ")
```

```
            addExpense(amountToAdd, category)
```

```

elif optionSelected == "2":
    removeExpense()
elif optionSelected == "3":
    listExpenses()
else:
    print("Invalid input. Please try again.")
'''

```

- The main loop runs indefinitely, presenting the user with the menu options.
- Based on the user's selection, it calls the appropriate function to add, remove, or list expenses.
- It handles invalid input for menu selection and amount entry.

---

## Output Example:

### 1. Initial State:

Here is a list of your expenses...

-----

# 0 - 51.00 - shirt

# 1 - 21.55 - groceries

'''

### 2. Adding a New Expense:

'''

Please choose from one of the following options...

1. Add A New Expense

2. Remove An Expense

3. List All Expenses

> 1

How much was this expense?

> 30.00

What category was this expense?

> utilities

...

...

Here is a list of your expenses...

-----

# 0 - 51.00 - shirt

# 1 - 21.55 - groceries

# 2 - 30.00 - utilities

...

### 3.Removing an Expense:

Please choose from one of the following options...

1. Add A New Expense

2. Remove An Expense

3. List All Expenses

> 2

Here is a list of your expenses...

-----

# 0 - 51.00 - shirt

# 1 - 21.55 - groceries

# 2 - 30.00 - utilities

What expense would you like to remove?

> 1

...

...

Here is a list of your expenses...

-----

# 0 - 51.00 - shirt

# 1 - 30.00 - utilities

...

#### 4. \*\*Listing All Expenses:\*\*

...

Please choose from one of the following options...

1. Add A New Expense

2. Remove An Expense

3. List All Expenses

> 3

Here is a list of your expenses...

-----

# 0 - 51.00 - shirt

# 1 - 30.00 - utilities

---