TDD Worksheet

Table of contents

## Introduction

You have been asked to developing a piece of software that will automate the processing of his bank statements so client can get better insights into his finances. This text is structured using a comma- separated values (CSV) format. Here is a sample of bank transactions:

2023-01-30,-100,Delivery  
2023-01-30,-50,Coles  
2023-01-02,6000,Salary  
2023-01-12,2000,Royalties  
2023-01-12,-4000,Rent  
2023-01-22,3000,Repairs  
2023-01-22,-30,Cinema

The client would like to get an answer for the following queries:

* What is the total profit and loss from a list of bank statements? Is it positive or negative?
* How many bank transactions are there in a particular month?
* What are his top-10 expenses?
* Which category does he spend most of his money on?

## Objectives

Practice test-driven development (TDD) by implementing functions that analyze transaction data stored in Pandas DataFrames.

## Task

Using the information below, follow TDD process to create each of the following functions.

### Sample data:

You can use the CSV supplied of generate you own data for testing:

import pandas as pd  
  
transactions = pd.DataFrame({  
 'Date': [datetime(2023, 1, 1), datetime(2023, 1, 2)],  
 'Amount': [100.50, 200.75],   
 'Description': ['Groceries', 'Electric bill']   
})

## Function 1: total\_profit\_and\_loss

Calculate the total profit or loss from a list of transactions.

Examples:

transactions = [  
 [100.50, 'Groceries'],  
 [-50.00, 'Clothing'],  
 [25.00, 'Breakfast']   
]  
  
Manual Calculation:   
100.50 - 50.00 + 25.00 = 75.50 (Profit)

Test Cases:

* All positive amounts = Profit
* Mix of positive/negative = Profit/Loss based on sum
* Empty list = No profit/loss

## Function 2: transactions\_in\_month

Count number of transactions in a specific month and year.

Examples:

transactions = [  
 ['2023/1/2', 10.50],  
 ['2023/1/2', 20.00],  
 ['2023/1/3', 15.00]  
]  
  
Manual Calculation:   
Transactions in February 2023 = 3

Test Cases:

* Month with multiple transactions
* Month with no transactions
* Invalid month

## Function 3: top\_10\_expenses

Get the top 10 expense amounts.

Examples:

transactions = [  
 [12.32, 'Groceries'],  
 [5.21, 'Dining Out'],  
 [15.98, 'Electric Bill']  
]  
  
Manual Calculation:  
Top Expenses = [15.98, 12.32]

Test Cases:

* More than 10 expenses
* Less than 10 expenses
* Empty list

## Function 4: category\_spending

Get the category with the highest total spend.

Examples:

transactions = [  
 [25.00, 'Food'],  
 [10.50, 'Entertainment'],  
 [50.00, 'Food']  
]  
  
Manual Calculation:   
Highest Spending Category = Food

Test Cases:

* Two categories with close totals
* One category much higher
* All spending in one category