**Exercise 3 Technical Design Document**

**Name:** Vanessa Belous

**Date Created:** September 23rd, 2025

**Program Description:** This program will take an input of the user’s monthly expenses and return the highest and lowest expenses of the month as well as the total expense for that month.

**Functions used in the Program (listed in the order they’re called in):**

1. **Function Name:** user\_input( )

**Description:** uses a dictionary to store expenses and their amounts to keep them associated with each other. Also uses a separate list for the amounts so that the values can be easily compared to find the highest and lowest amounts

**Parameters:** None

**Variables:**

1. expenses\_dict = dictionary that stores the amount as the key and the expense name as the value
2. amount\_list = a list that stores only the amounts as floats, to use the reduce method
3. temp\_input = is the user’s input, in the format Expense: $Amount
4. temp\_expense = used to store the expense in dictionary as a string
5. temp\_amount = used to store the amount in dictionary and list as a float

**Logical Steps:**

1. prints statement instructing user how to input their information and what to input when they’re finished
2. uses a while loop to continuously accept data until an if statement inside it breaks the loop (condition to break loop is when user inputs “Done”
   1. uses .split( ) method with delimiter “: $” to separate a single input into two different variables (one for expense and one for amount)
   2. stores values in dictionary so that the amount is the key to the expense name
   3. appends the amount to the initially empty amount\_list so that the lambda function in another function to iterate over the values

**Returns:** Nothing

1. **Function Name:** main( )

**Description:** will calculate highest, lowest, and total expense using user data taken in through other function, through lambda functions. Then, it displays those results to the screen

**Parameters:** None

**Variables:**

1. highest\_amount = stores result of reduce method using a lambda function that finds the highest value in the amount list
2. highest\_expense = finds and stores the associated expense name to the highest value in the dictionary
3. lowest\_amount = stores result of reduce method using a lambda function that finds the lowest value in the amount list
4. lowest\_expense = finds and stores the associated expense name to the lowest value in the dictionary
5. total\_expenses = stores result of the reduce method using a lambda function to calculate the total value of expenses in the amount list

**Logical Steps:**

1. calls the user\_input( ) function to store values in the dictionary and amount list so they can be used
2. calculates the highest amount using the reduce method, and finds the associated expense name in the dictionary
3. calculates the lowest amount using the reduce method, and finds the associated expense name in the dictionary
4. calculates the total value of expenses
5. prints the results in a formatted way

**Returns:** Nothing

**Program Logical Steps:**

1. functools is imported so that the reduce method can be used
2. the dictionary and list used throughout the program are initiated as empty
3. the main( ) function is called
   1. The user\_input ( ) method is called, which populates the dictionary and list
   2. The lowest, highest, and total values are found/calculated
   3. The results are displayed to the screen

**Link to Repository:** <https://github.com/VBelous1/COP2373/tree/master/Week%206>

**Output Screenshot:**

**A screenshot of a computer

AI-generated content may be incorrect.**