**ChauncyKent\_03\_01 Technical Design Document**

**Name:** Chauncy Kent

**Date Created:** 9/14/2025

**Program Description:** A class object that collects expense information and analyzes it by determining and displaying the total of all expenses, the highest expense, and the lowest expense.

**Functions used in the Program:**

1. **Function Name:** main

**Description:** The main function of the program that initializes all other steps.

**Parameters:** none.

**Variables:** none.

**Logical Steps:**

* Creates an ExpenseAnalyzer Object.
* Calls the collect\_expenses method.
* Calls the analyze\_expenses method.
* Calls the display\_expenses method.

**Returns:** none.

2. **Function Name:** collect\_expenses

**Description:** Creates a loop adding the entered expenses and expense types to the dictionary on each loop, then asks if there are more expenses to be entered. Includes input validation to make sure the user enters either "Y" or "N".

**Parameters:** none.

**Variables:**

* more (a loop control variable)
* answer (a string for storing the user’s answer to the input question)
* invalid (a loop control variable)

**Logical Steps:**

* Sets up a loop.
* Calls \_update\_records in that loop.
* Sets up an input validation loop.
* Asks the user if there are more expenses to enter.
* If yes, loops back through; otherwise, terminates the loop.

**Returns:** none.

3. **Function Name:** \_update\_records

**Description:** Takes the collected expense and expense type and adds them to the expenses dictionary.

**Parameters:** none.

**Variables:**

* expense (stores the expense collected by the \_get\_expense method.
* expense\_type (stores the expense type collected by the \_get\_type method.

**Logical Steps:**

* Calls the \_get\_expense method.
* Calls the \_get\_type method.
* Adds expense\_type: expense as a new key: value pair in the \_expenses dictionary.

**Returns:** none.

4. **Function Name:** \_get\_expense

**Description:** Handles the collection and storage of an expense. Includes data validation to ensure the entered value is a number. Returns expense.

**Parameters:** none.

**Variables:**

* invalid (a loop control variable)
* expense (stores the expense entered by the user)

**Logical Steps:**

* Initializes a try/except block.
* Collects an expense from the user.
* Attempts to convert it to a float.
* If there’s an error, it loops and asks again for an expense; otherwise, it ends the loop.
* Returns the collected expense.

**Returns:** expense

5. **Function Name:** \_get\_type

**Description:** Handles the collection and storage of an expense type. Includes data validation to ensure unique types are gathered for each expense. Returns expense\_type.

**Parameters:** none.

**Variables:**

* invalid (a loop control variable)
* expense\_type (stores the expense type entered by the user)

**Logical Steps:**

* Initializes a try/except block.
* Collects an expense type from the user.
* Checks to see if it’s in the dictionary already; if so, raises an exception.
* If there’s an error, it loops and asks again for an expense; otherwise, it ends the loop.
* Returns the collected expense type.

**Returns:** expense\_type

6. **Function Name:** analyze\_expenses

**Description:** Makes use of the reduce function to analyze the entered expenses, calculating the expense total, the highest expense, and the lowest expense.

**Parameters:** none.

**Variables:**

* key (the current dictionary key)
* value (the current dictionary value)

**Logical Steps:**

* Uses the reduce function to total all the expenses in the \_expenses dictionary.
* Uses the reduce function to determine the highest expense in the \_expenses dictionary.
* Uses a for loop and compares each value from each key: value pair to the highest expense. When a match is found, assigns that value’s key as the \_highest\_type.
* Uses the reduce function to determine the lowest expense in the \_expenses dictionary.
* Uses a for loop and compares each value from each key: value pair to the lowest expense. When a match is found, assigns that value’s key as the \_lowest\_type.

**Returns:** none.

7. **Function Name:** display\_analysis

**Description:** Displays the results of the analysis of the entered expenses, including the total of all expenses, the highest expense, and the lowest expense.

**Parameters:** none.

**Variables:** none.

**Logical Steps:**

* Prints a blank line.
* Prints the total of all expenses.
* Prints the highest expense and its type.
* Prints the lowest expense and its type.
* Prints a blank line.

**Returns:** none.

**Logical Steps:**

1. main
2. collect\_expenses
   1. \_update\_records
      1. \_get\_expenses
      2. \_get\_type
3. analyze\_expenses
4. display\_expenses

**Link to your repository:** <https://github.com/ChauncyJKent/COP2373/activity>

**Output Screenshot:**

**A black screen with colorful text

AI-generated content may be incorrect.**