**Abstraction - Design Activity**

Based on the program specification, here are classes, their responsibilities, behaviors, and attributes:

**1. Journal Class**

• Responsibility: Managing the collection of journal entries.

• Behaviors/Methods:

• addEntry(entry): Add a new entry to the journal.

• displayEntries(): Iterate through and display all entries.

• saveToFile(filename): Save the journal to a specified file.

• loadFromFile(filename): Load the journal from a specified file, replacing current entries.

• Attributes/Member Variables:

• entries: A list (or array) of Entry objects to store individual journal entries.

**2. Entry Class**

• Responsibility: Representing an individual journal entry.

• Behaviors/Methods:

• display(): Display the details of the entry, including the prompt, response, and date.

• Attributes/Member Variables:

• prompt: A string representing the prompt/question associated with the entry.

• response: A string representing the user's response to the prompt.

• date: A string or date object representing the date when the entry was made.

**3. PromptGenerator Class**

• Responsibility: Generating prompts for the journal entries.

• Behaviors/Methods:

• generatePrompt(): Generate a random prompt from a list of prompts.

• Attributes/Member Variables:

• promptsList: A list (or array) of prompt strings to choose from.

These classes encapsulate the major components of the program and demonstrate the principle of abstraction by separating concerns and responsibilities. The Journal class manages entries, the Entry class represents individual entries, and the PromptGenerator class handles the generation of prompts. File handling operations (saving and loading) will be part of the Journal class.