Data Structures Project

Title: **Personal Task Manager**

CSE-228

Lovely Professional University
Jalandhar, Punjab, India.

Delivered by:	**Received by:**
Names of the student: Nabajit Pahadsingh Reg. No.: 12210057 Roll No.: RK22UN24 Signature:	Name of the faculty: Shubham Sharma UID: 64339 Signature:

Acknowledgment

I am deeply grateful for the completion of the "Personal Task Manager" project, which I undertook as a solo endeavour. This project reflects my dedication and commitment to developing a personal task management solution.

I extend my heartfelt appreciation to my mentor, Shubham Sharma, who played an invaluable role in guiding me through the complexities of the project. Shubham's mentorship, expertise, and unwavering support have been instrumental in shaping the project and ensuring that it meets the highest standards of quality.

While this project was undertaken independently, I must also express my gratitude to the broader community, friends, and family who provided encouragement and understanding throughout the development process. Their unwavering support has been a constant source of motivation.

Furthermore, I appreciate the open-source community for providing essential resources and libraries that enriched my project. The collaborative spirit of the open-source world has been a source of inspiration and learning.

In conclusion, the "Personal Task Manager" project is a testament to what can be achieved through determination and mentorship. I am grateful to everyone who has played a part, directly or indirectly, in the successful completion of this project. It stands as a symbol of my commitment to personal organization and software development.

Table of Contents

Introduction

In an increasingly fast-paced world, effective personal organization has become paramount. Balancing the demands of work, studies, family, and personal life often requires a reliable system to manage tasks, notes, and reminders. To address this need, we introduce the "Personal Task Manager," a versatile and user-friendly Java-based application.

The modern lifestyle is characterized by an overwhelming flow of information and responsibilities. Tasks pile up, ideas are born and need to be captured, and crucial appointments and deadlines loom on the horizon. The Personal Task Manager serves as a digital personal assistant, providing a comprehensive solution for users to manage their lives efficiently.

This project is designed to empower individuals with the tools necessary to streamline their daily routines. It offers the flexibility to create, update, view, and delete tasks, notes, and reminders. Whether you need to prioritize tasks, jot down ideas, or receive timely alerts for essential commitments, the Personal Task Manager has you covered.

Using an intuitive menu-driven interface, the application ensures that users of all backgrounds can easily navigate and interact with the system. Furthermore, by incorporating data storage and persistence features, the application allows users to save and load their task, note, and reminder data, ensuring their information remains accessible and secure.

The "Personal Task Manager" is not just a productivity tool; it's a step towards regaining control over the chaos of daily life. It represents the fusion of modern technology and timeless organization principles to help users lead more organized, productive, and fulfilling lives.

Objects and Scopes of the Project

1. Task Management

Tasks are a fundamental component of the Personal Task Manager. The project allows users to create, update, view, and delete tasks. Each task consists of a name, due date, priority, and status. With task management, users can organize their to-do lists, set priorities, and track task completion.

2. Note Management

Notes serve as digital notepads, enabling users to store and access important information. The Personal Task Manager facilitates the creation, modification, viewing, and deletion of notes. Each note comprises a title and content, making it a versatile tool for recording ideas, thoughts, and reminders.

3. Reminder Management

Reminders help users stay on top of their commitments and appointments. Users can add, update, view, and delete reminders using this application. Reminders are categorized by text, date, and time, ensuring that important events are never missed.

4. User Interface and Interaction

The application offers a user-friendly menu-driven interface. Users can select options to create and manage tasks, notes, and reminders. The interface allows users to interact with the system easily, making it accessible to people of all backgrounds.

5. Data Storage and Persistence

The application incorporates a file I/O mechanism, allowing users to save and load their task, note, and reminder data. This feature ensures data persistence and enables users to continue working with their information across sessions.

Application Tools

1. Java Programming Language

The entire project is implemented in Java, a versatile and widely-used programming language. Java offers platform independence, making it an excellent choice for developing cross-platform applications.

2. Integrated Development Environment (IDE)

For coding and development, we employed the popular Integrated Development Environment, IntelliJ IDEA. This IDE provides essential tools for writing, testing, and debugging code efficiently.

3. Data Structures

To manage tasks, notes, and reminders, we utilize data structures such as ArrayLists. These structures provide dynamic storage and easy manipulation of data.

4. Object-Oriented Programming (OOP)

We've adopted OOP principles to design the project, ensuring code modularity and reusability. Object-oriented design allows us to create classes that represent tasks, notes, and reminders, making the application highly organized.

Algorithm Explanation

The project includes three core classes: Task, Note, and Reminder.

1. **Task Class:**
    - Properties: taskName, dueDate, priority, status.
    - Constructor: The Task class has a constructor that initializes these properties.
    - Getters and Setters: To access and modify these properties.
2. **Note Class:**
    - Properties: title, content.
    - Constructor: The Note class includes a constructor to initialize its properties.
    - Getters and Setters: To access and modify these properties.
3. **Reminder Class:**
    - Properties: reminderText, reminderDate, reminderTime.
    - Constructor: The Reminder class has a constructor to initialize its properties.
    - Getters and Setters: To access and modify these properties.
4. **TaskManager Class**

The TaskManager class is the core of the application, responsible for managing tasks, notes, and reminders. It utilizes ArrayLists to store these objects and provides methods for adding, viewing, and deleting them.

Methods:

- addTask(Task task): Adds a task to the list of tasks.
- addNote(Note note): Adds a note to the list of notes.

- addReminder(Reminder reminder): Adds a reminder to the list of reminders.
- viewTasks(): Displays a list of tasks.
- viewNotes(): Displays a list of notes.
- viewReminders(): Displays a list of reminders.
- deleteTask(String taskName): Deletes a task by matching its name.
- deleteNote(String noteTitle): Deletes a note by matching its title.
- deleteReminder(String reminderText): Deletes a reminder by matching its text.

5. **User Interface and Interaction**

The project's main class, PersonalTaskManager, provides a menu-driven user interface. It presents users with options to perform various actions, including adding, viewing, and deleting tasks, notes, and reminders. The interface ensures a straightforward and accessible user experience.

Summary

The "Personal Task Manager" project is a Java-based application designed to assist users in efficiently organizing their daily tasks, notes, and reminders. It offers a user-friendly interface and tools to manage these aspects of personal organization effectively.

The project encompasses four core aspects: task management, note management, reminder management, and data storage/persistence. Users can create, update, view, and delete tasks, notes, and reminders. Additionally, the application facilitates data persistence, allowing users to save and load their information for continued use.

The tools employed in this project include Java, an integrated development environment, data structures like ArrayLists, and object-oriented programming principles. These components work together to create a reliable and versatile personal task management solution.

With the "Personal Task Manager," users can streamline their daily activities, enhance productivity, and keep track of important information. This project aims to address the need for an efficient and accessible personal organization tool, making it a valuable addition to the world of time management and productivity applications.

Using the exchange rates obtained from the API, the program applies the conversion formula to calculate the converted amount. This involves multiplying the base currency amount by the exchange rate of the target currency.