Time Management App

Introduction

```
Welcome to the CLI To-Do List App! Type 'help' for usage instructions.
Usage Instructions:
        To record a task:
                record <DATE> <FROM TIME> <TO TIME> '<TASK DESCRIPTION>' :TAG
                Examples:
                        record today 09:30 10:30 'Study Python' :STUDY
                        record 2023/04/01 07:00PM 09:00PM 'Read a book'
        To query tasks:
                query <DATE> or 'TASK DESCRIPTION' or :TAG
                Examples:
                        query today
                        query 'Study Python'
                        query :STUDY
        To generate a time usage report:
                report <FROM DATE> <TO DATE>
                Examples:
                        report 2021/01/01 2022/01/01
        To see tasks sorted by time spent on them:
                priority
                Example:
                        priority
        Type 'exit' to close the application.
```

Objective

To provide a user-friendly tool for students to track and manage their time effectively.

• Significance

Elevates student organization and productivity by offering insights into their time allocation.

Feature 1: Activity Recording

```
To generate a time usage report:
        report <FROM DATE> <TO DATE>
        Examples:
            report 2021/01/01 2022/01/01
   To see tasks sorted by time spent on them:
        priority
        Example:
            priority
   Type 'exit' to close the application.
> record today 09:30 10:30 'Study Python' :STUDY
Task recorded successfully.
```

- Functionality allowing users to log activities with attributes such as date, time, task, and tags.
- Facilitates detailed tracking of daily routines and commitments.

Feature 2: Query Function

```
report 2021/01/01 2022/01/01
    To see tasks sorted by time spent on them:
        priority
        Example:
            priority
    Type 'exit' to close the application.
> record today 01:30PM 02:30PM 'Study Java' :STUDY
Task recorded successfully.
> query today
2023/12/10 09:30 10:30 Study Python STUDY
2023/12/10 13:30 14:30 Study Java STUDY
```

- Commands designed for querying activities such as "query today" for daily accomplishments or "query 'Java'" for task-specific tracking.
- Empowers students to review and adjust their time management strategies based on queries.

Feature 3: Customizable Reports

```
2023/12/09 17:30 18:30 Study Design Patterns STUDY
> record 2023/12/08 02:30PM 04:30PM 'Sprint Planning' :WORK
Task recorded successfully.
> report 2023/12/08 today
2023/12/10 09:30 10:30 Study Python STUDY
2023/12/10 13:30 14:30 Study Java STUDY
2023/12/09 17:30 18:30 Study Design Patterns STUDY
2023/12/08 14:30 16:30 Sprint Planning WORK
> priority
Task: Sprint Planning, Time Spent: 2h 0m 0s
Task: Study Python, Time Spent: 1h 0m 0s
Task: Study Java, Time Spent: 1h 0m 0s
Task: Study Design Patterns, Time Spent: 1h 0m 0s
>
```

- Generation of tailored reports for any specified date range.
- Prioritization of activities based on collected data to identify focus areas.
- Offers analytical perspective on time usage patterns, aiding in decision-making for better efficiency.

Testings

```
platform darwin -- Python 3.10.9, pytest-7.1.2, pluggy-1.0.0
rootdir: /Users/jasoncao/Desktop/Fall23/ASE420/todo
plugins: anyio-3.5.0, asyncio-0.21.1, mock-3.12.0
asyncio: mode=strict
collected 37 items
test/test_command_factory.py .
test/test_database.py ......
                                                                                                [ 35%]
test/test_formatter.py ...
test/test_input_handler.py .....
test/test_priority_command.py .
test/test_query_command.py .
test/test_record_command.py .
                                                                                                [ 78%]
                                                                                                [ 81%]
test/test_report_command.py .
test/test_todo_cli.py ......
```

UML Diagrams

- Design Patterns Used:
 - Command Pattern:
 CommandInterface,
 RecordCommand, QueryCommand.
 - Strategy Pattern: Formatter
 - Factory Pattern: CommandFactory

