CHRIST NAGAR CENTRAL SCHOOL

Central Board of Secondary Education

For AISSCE EXAM

Session 2021-22

Project Report on

To-Do List Application

Submitted by

Name	:	**Arunima Sanjeev**
Class	:	**XII-G**
Roll No	:	**02**
Reg No	:	**24604612**
Subject	:	**Computer Science (083)**
Year	:	**2021-22**

Certificate

This is to certify that the project in the subject of Computer Science entitled **ToDo List Project** is a bonafide record of work done by **Arunima Sanjeev** (Reg no: **24604612**) for the requirement of XII Practical examination of AISSCE during the academic year 2021-22.

Teacher in Charge External Examiner

Head of the Institution

Acknowledgement

I would like to express my special thanks to my teacher Jithin J as well as our principal Rev Fr. Bino Pattarkalam CMI who gave me the golden opportunity to do this wonderful project on the topic “ToDo List” which also helped me in doing a lot of research and I came to know about so many new things.

I would also like to thank my parents and friends who helped a lot in finalizing this project within the limited time frame.

Last but not least, I would like to thank all those who had helped directly or indirectly towards the completion of this project.

I. Introduction

We will develop a ToDo List application using which users can manage their tasks. Inspired by Google Tasks, this application can be used by people of all age groups for managing their tasks in a systematic manner. We will use Python for developing this application, and MySQL database will be used for storing data.

II. Objectives and Scope

The idea is to develop an application which will help in making task management easier for users. The application will help manage and edit tasks, helping in getting tasks done faster, and in a systematic manner. Task tracking allows the users to see which tasks are most important or time intensive. Having all the tasks listed at a place also helps in seeing the pending tasks and their status changes.

Definition of a task

A task will consist of the following information:

#	Information	Details
1	Category	The type of the task, or which category a task belongs to. Examples are “School”, “Homework”, etc. This will be optional.
2	Deadline	The due date (or time) of the task. This will be optional.
3	Description	Description of the task.
4	Status	The task status - pending or completed.

Key Features

The following are some of the key features that will be supported.

★ Task operations

- Creating a task.
- List all open tasks.
- List all open tasks of a given category.
- List all open tasks due before a given deadline.
- Update an existing task.
- Mark an open task as completed.
- List all completed tasks.

★ Reports

In addition to the above, users will also have the option to generate reports which will contain their usage information, containing information like number of open tasks, closed tasks, etc. The report will be generated in the form of a CSV file.

III. Modules

We will have the following modules in the application:

- Task
- Database
- Report

Database module

This module will be responsible for all the interactions with the database, i.e., the create, read, update and delete operations. It will use the MySQL Connector for interacting with the MySQL database using SQL queries.

Task module

This module will handle the functionalities related to tasks (or “todo”s). Some such functionalities include creating a new task, fetching existing tasks, etc. This module will depend on the database module for operations involving the database.

Report module

This module will be responsible for the various kinds of report generations. Again, the database module will be a dependency for fetching the tasks details. All reports will be produced in the form of CSV files.

Apart from the above modules, we will also have the main module that interacts with the user and processes the user queries.

IV. Hardware and Software Specification

Hardware	
Processor	Pentium IV +
Clock speed	2 GHz
RAM	512 MB
Hard disk capacity	80 GB
Keyboard	101 Keys
Mouse	Optical Mouse

Software	
Operating System	Windows/Linux/MacOS
Front End	Python
Back End	MySQL Server

V. Technologies Used

Python

Python is a powerful general-purpose programming language. It is currently the most widely used multi-purpose, high-level programming language. We will be developing the application in the Python language.

MySQL

MySQL is an Open Source relational database management system (RDBMS). MySQL is developed, distributed, and supported by Oracle Corporation. We will be using MySQL as our database for storing the tasks information.

Interface Python with MySQL

Python needs a MySQL driver to access the MySQL database. MySQL Python/Connector is one such interface for connecting to a MySQL database server from Python. We will be using the connector for all the database interactions.

VI. Source Code

• main.py

```
'''
The main module.

Command to run:
>> python3 main.py
'''

import task as task_module
import report as report_module
import database as database_module

def get_user_choice():
    print('1. Create a task')
    print('2. List tasks')
    print('\ta. Open tasks')
    print('\tb. Open tasks of a category')
    print('\tc. Open tasks before deadline')
    print('\td. Completed tasks')
    print('\te. All tasks')
    print('3. Update a task')
    print('\ta. Modify attribute')
    print('\tb. Mark as completed')
    print('\tc. Mark as pending')
    print('4. Generate report')
    print('5. Exit the application')
    print()
    choice = input('Enter your choice (eg. 1, 2c, 3b, etc.): ')
    print()
    valid_choices = ['1', '2a', '2b', '2c', '2d', '2e', '3a', '3b', '3c', '4', '5']
    if choice not in valid_choices:
        print('Invalid choice entered!')
        return None
    return choice


print('=====================================')
print('+++++++ ToDo List Application +++++++')
print('=====================================')
print()

while(1):
    choice = get_user_choice()
    if choice == None:
        print('\n' * 5)
        continue
    elif choice[0] == '1':
        task_module.create_task()
    elif choice[0] == '2':
        if choice[1] == 'a':
            task_module.list_open_tasks()
        elif choice[1] == 'b':
            task_module.list_open_tasks_of_category()
        elif choice[1] == 'c':
            task_module.list_open_tasks_before_deadline()
        elif choice[1] == 'd':
            task_module.list_completed_tasks()
        else:
            task_module.list_all_tasks()
    elif choice[0] == '3':
        if choice[1] == 'a':
            task_module.update_task()
        elif choice[1] == 'b':
            task_module.mark_task_completed()
        else:
            task_module.mark_task_pending()
    elif choice[0] == '4':
        report_module.generate_report()
    else:
        break
    print('\n' * 5)

print('==================')
print('+++ Thank you! +++')
print('==================')
print()
```

- **task.py**

```
'''
The task module.  Responsible for creating, updating, listing tasks.
'''

import database as database_module

def create_task():
        category = input('Enter the category (press enter to skip): ')
        deadline = input('Enter the deadline in yyyy-mm-dd format (press enter to
skip): ')
        description = input('Enter the description (max 25 chars): ')
        database_module.create_task(category, deadline, description)
        print('Done!')

def list_open_tasks():
        tasks = database_module.fetch_open_tasks()
        print_tasks(tasks)

def list_open_tasks_of_category():
        category = input('Enter the category: ')
        print()
        tasks = database_module.fetch_open_tasks_of_category(category)
        print_tasks(tasks)

def list_open_tasks_before_deadline():
        deadline = input('Enter the deadline in yyyy-mm-dd format: ')
        tasks = database_module.fetch_open_tasks_before_deadline(deadline)
        print_tasks(tasks)

def list_completed_tasks():
        tasks = database_module.fetch_completed_tasks()
        print_tasks(tasks)

def list_all_tasks():
        tasks = database_module.fetch_all_tasks()
        print_tasks(tasks)

def print_tasks(tasks):
        if len(tasks) == 0:
                print('No tasks found!')
                return
        print('-' * 76)
        print('|', '%4s' % 'Id', '|', '%10s' % 'Category', '|', '%11s' % 'Deadline',
'|', '%25s' % 'Description', '|', '%10s' % 'Status', '|')
        print('-' * 76)
        for task in tasks:
                print('|', '%4s' % task[0], '|', '%10s' % task[1], '|', '%11s' %
task[2], '|', '%25s' % task[3], '|', '%10s' % task[4], '|')
        print('-' * 76)

def update_task():
        id = int(input('Enter the task id: '))
        print()
        print('1. Category')
        print('2. Deadline')
        print('3. Description')
        attribute = ['category', 'deadline', 'description'][int(input('Enter the
attribute to update (1-3): ')) - 1]
        if attribute == 'deadline':
                value = input('Enter the deadline in yyyy-mm-dd format: ')
        else:
                value = input('Enter the new value: ')
        database_module.update_task(id, attribute, value)
        print('Done!')
        print()
```

```
def mark_task_pending():
        id = int(input('Enter the task id: '))
        print()
        database_module.update_task(id, 'status', 'Pending')
        print('Done!')
        print()

def mark_task_completed():
        id = int(input('Enter the task id: '))
        print()
        database_module.update_task(id, 'status', 'Completed')
        print('Done!')
        print()
```

- ## database.py

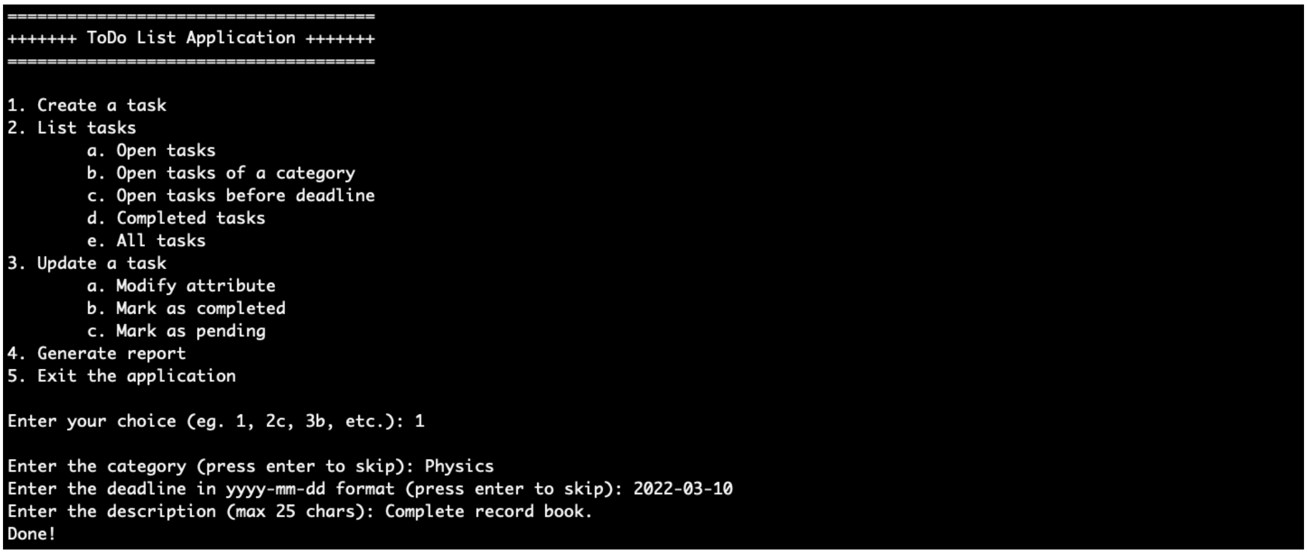
```
'''
The database module.  Responsible for all interactions with the database.
'''

import mysql.connector

taskdb = mysql.connector.connect(host='localhost', user='root', database='taskdb')
cursor = taskdb.cursor()

def create_task(category, deadline, description):
        if category == '':
                category = None
        if deadline == '':
                deadline = None
        query = 'insert into task(category, deadline, description) values(%s, %s, %s)'
        cursor.execute(query, [category, deadline, description])
        taskdb.commit()

def update_task(id, attribute, value):
        query = 'update task set {}=%s where id=%s'.format(attribute)
        cursor.execute(query, [value, id])
        taskdb.commit()

def fetch_open_tasks():
        query = 'select * from task where status=\'Pending\''
        cursor.execute(query)
        return cursor.fetchall()

def fetch_all_tasks():
        query = 'select * from task'
        cursor.execute(query)
        return cursor.fetchall()
```

```
def fetch_open_tasks_of_category(category):
        query = 'select * from task where status=\'Pending\' and category=%s'
        cursor.execute(query, [category])
        return cursor.fetchall()

def fetch_completed_tasks():
        query = 'select * from task where status=\'Completed\''
        cursor.execute(query)
        return cursor.fetchall()

def fetch_open_tasks_before_deadline(deadline):
        query = 'select * from task where deadline <= date %s and status=0'
        cursor.execute(query, [deadline])
        return cursor.fetchall()
```

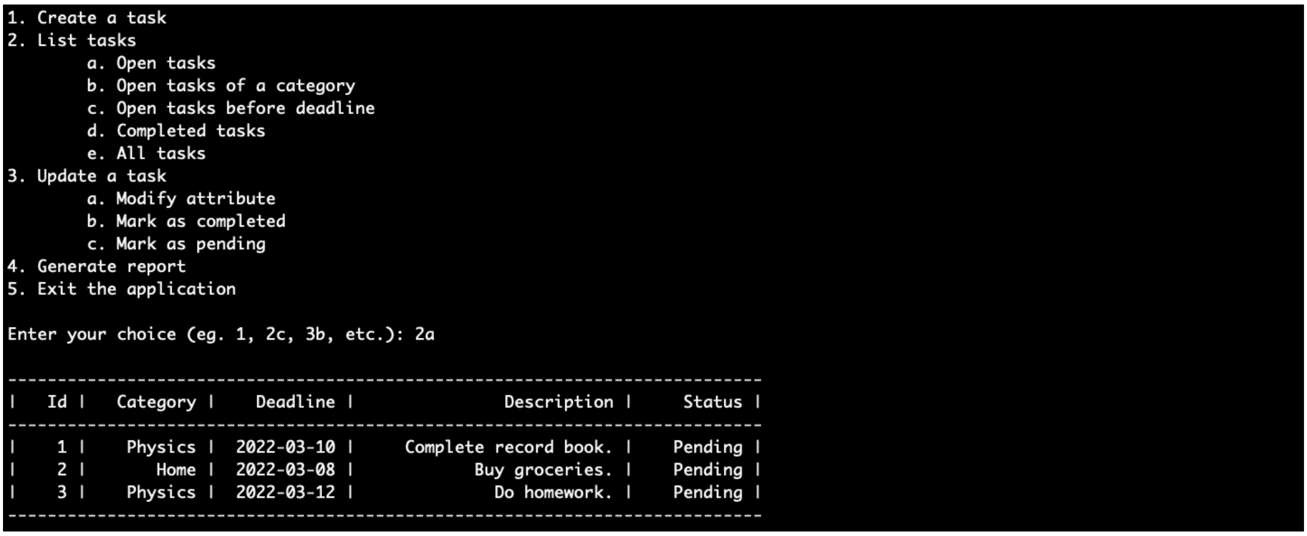
- **report.py**

```
'''
The report module.  Generates a report of all the tasks as a CSV file.
'''

import csv
import database as database_module

def generate_report():
        tasks = database_module.fetch_all_tasks()
        generate_csv(tasks)
        print('task_report.csv has been generated!')

def generate_csv(tasks):
        with open('task_report.csv', 'w', newline='') as f:
                csv_writer = csv.writer(f, delimiter=',')
                fields = ['Id', 'Category', 'Deadline', 'Description', 'Status']
                csv_writer.writerow(fields)
                for i in tasks:
                        csv_writer.writerow(i)
```

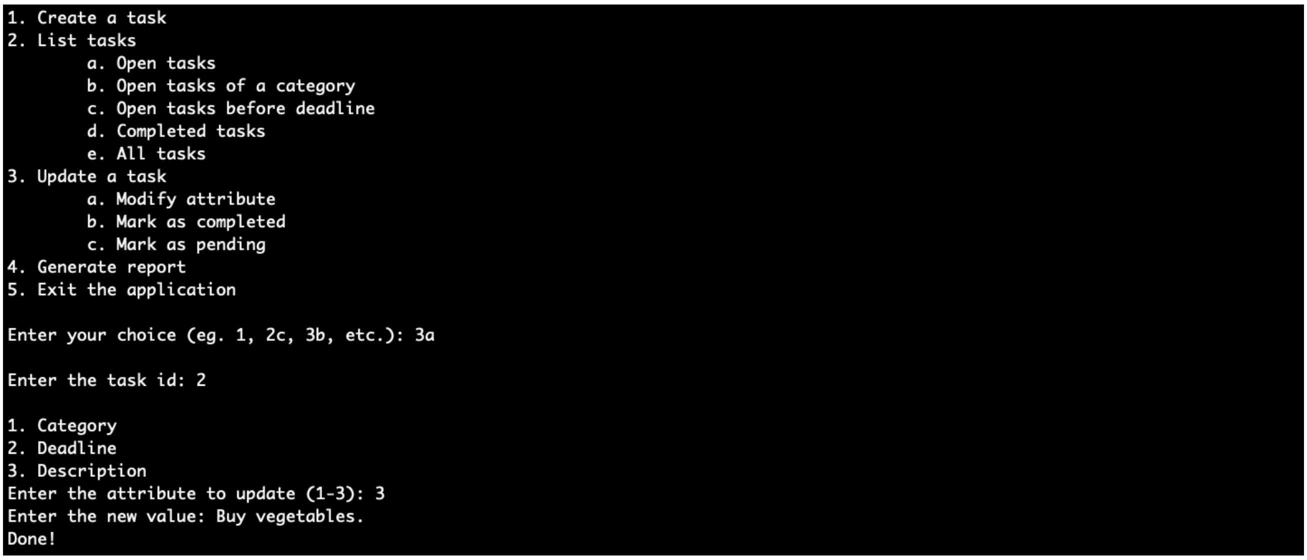
VII. Output



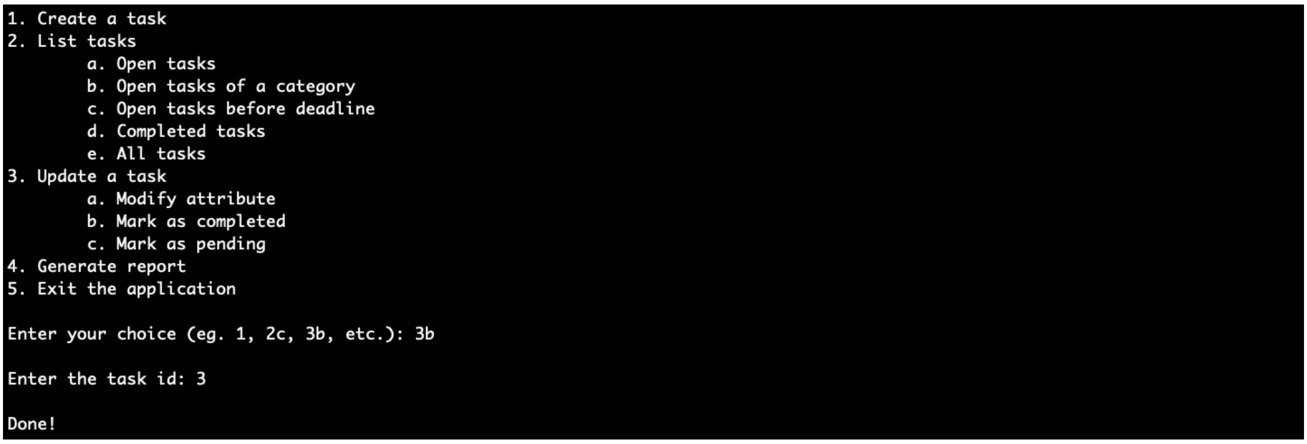
```
=====================================
+++++++ ToDo List Application +++++++
=====================================

1. Create a task
2. List tasks
        a. Open tasks
        b. Open tasks of a category
        c. Open tasks before deadline
        d. Completed tasks
        e. All tasks
3. Update a task
        a. Modify attribute
        b. Mark as completed
        c. Mark as pending
4. Generate report
5. Exit the application

Enter your choice (eg. 1, 2c, 3b, etc.): 1

Enter the category (press enter to skip): Physics
Enter the deadline in yyyy-mm-dd format (press enter to skip): 2022-03-10
Enter the description (max 25 chars): Complete record book.
Done!
```



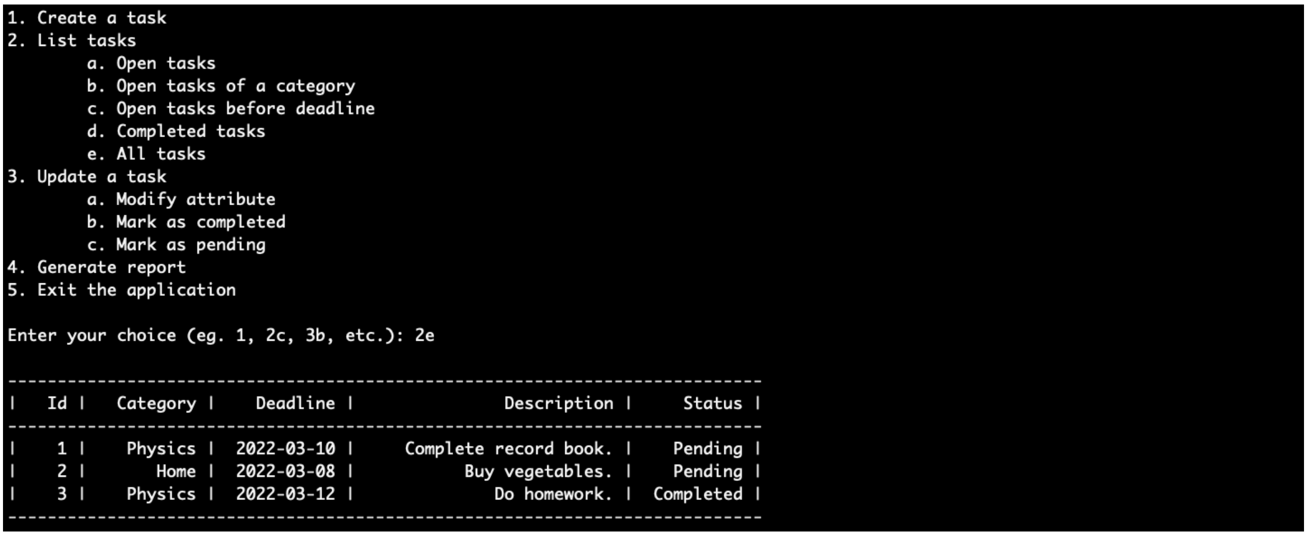
```
1. Create a task
2. List tasks
        a. Open tasks
        b. Open tasks of a category
        c. Open tasks before deadline
        d. Completed tasks
        e. All tasks
3. Update a task
        a. Modify attribute
        b. Mark as completed
        c. Mark as pending
4. Generate report
5. Exit the application

Enter your choice (eg. 1, 2c, 3b, etc.): 2a

----------------------------------------------------------------------------
|   Id |   Category |    Deadline |               Description |     Status |
----------------------------------------------------------------------------
|    1 |    Physics |  2022-03-10 |     Complete record book. |    Pending |
|    2 |       Home |  2022-03-08 |            Buy groceries. |    Pending |
|    3 |    Physics |  2022-03-12 |              Do homework. |    Pending |
----------------------------------------------------------------------------
```



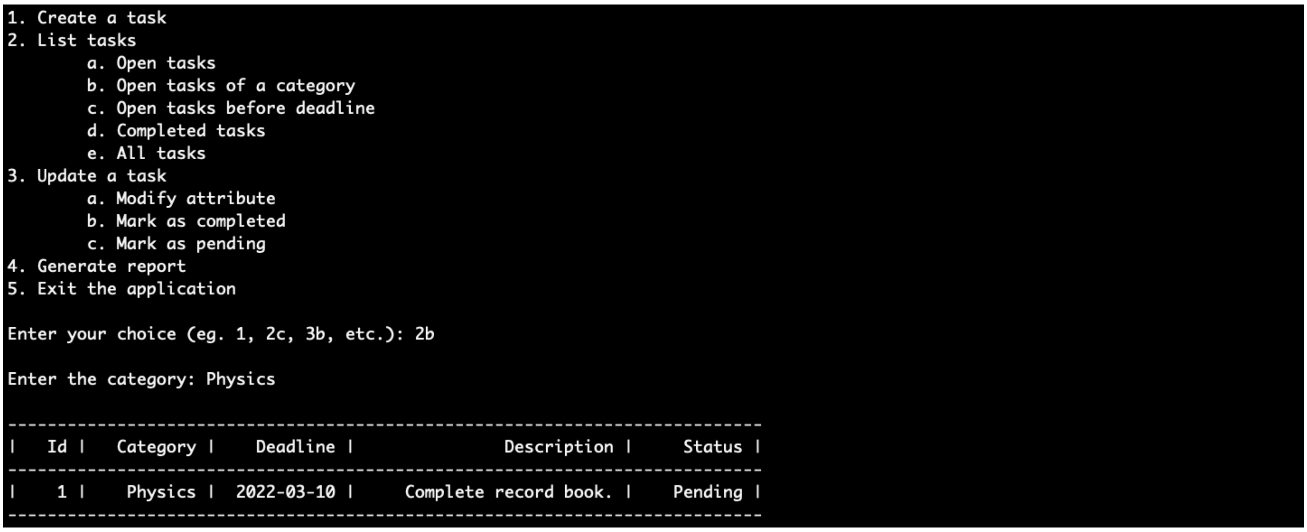
```
1. Create a task
2. List tasks
        a. Open tasks
        b. Open tasks of a category
        c. Open tasks before deadline
        d. Completed tasks
        e. All tasks
3. Update a task
        a. Modify attribute
        b. Mark as completed
        c. Mark as pending
4. Generate report
5. Exit the application

Enter your choice (eg. 1, 2c, 3b, etc.): 3a

Enter the task id: 2

1. Category
2. Deadline
3. Description
Enter the attribute to update (1-3): 3
Enter the new value: Buy vegetables.
Done!
```



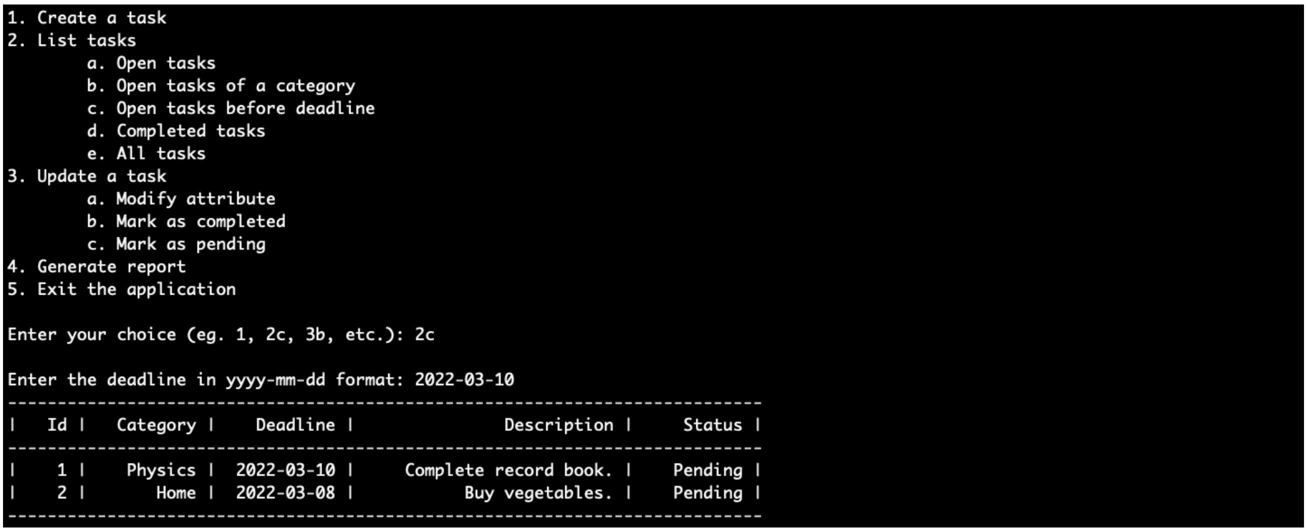
```
1. Create a task
2. List tasks
        a. Open tasks
        b. Open tasks of a category
        c. Open tasks before deadline
        d. Completed tasks
        e. All tasks
3. Update a task
        a. Modify attribute
        b. Mark as completed
        c. Mark as pending
4. Generate report
5. Exit the application

Enter your choice (eg. 1, 2c, 3b, etc.): 3b

Enter the task id: 3

Done!
```



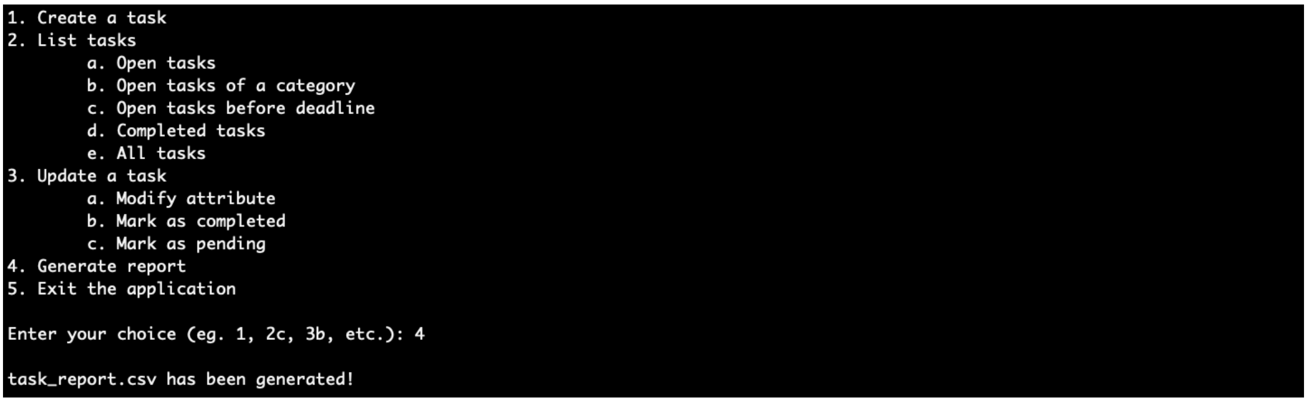
```
1. Create a task
2. List tasks
        a. Open tasks
        b. Open tasks of a category
        c. Open tasks before deadline
        d. Completed tasks
        e. All tasks
3. Update a task
        a. Modify attribute
        b. Mark as completed
        c. Mark as pending
4. Generate report
5. Exit the application

Enter your choice (eg. 1, 2c, 3b, etc.): 2e

----------------------------------------------------------------------------
|   Id |   Category |    Deadline |               Description |     Status |
----------------------------------------------------------------------------
|    1 |    Physics |  2022-03-10 |     Complete record book. |    Pending |
|    2 |       Home |  2022-03-08 |           Buy vegetables. |    Pending |
|    3 |    Physics |  2022-03-12 |              Do homework. |  Completed |
----------------------------------------------------------------------------
```



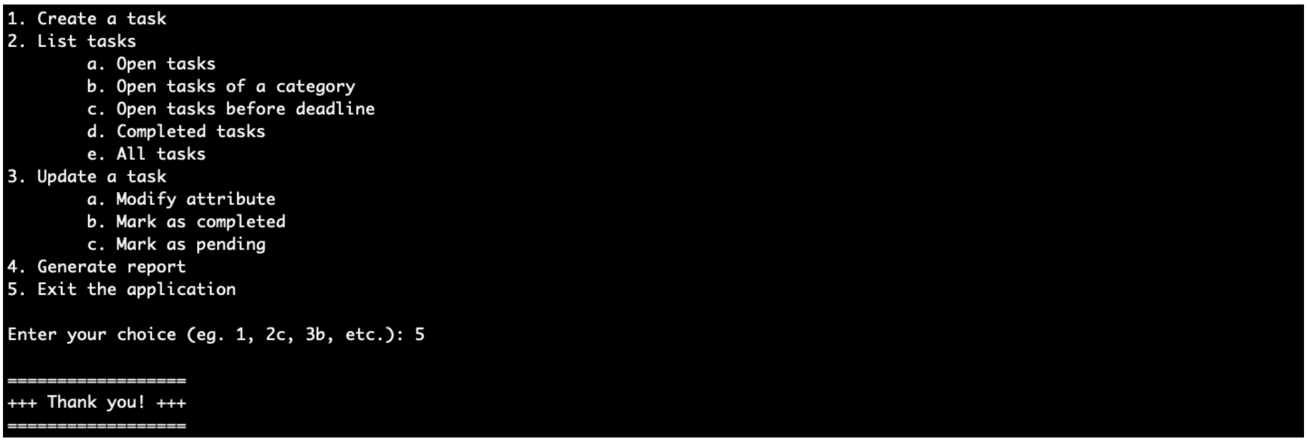
```
1. Create a task
2. List tasks
        a. Open tasks
        b. Open tasks of a category
        c. Open tasks before deadline
        d. Completed tasks
        e. All tasks
3. Update a task
        a. Modify attribute
        b. Mark as completed
        c. Mark as pending
4. Generate report
5. Exit the application

Enter your choice (eg. 1, 2c, 3b, etc.): 2b

Enter the category: Physics

----------------------------------------------------------------------------
|   Id |   Category |    Deadline |               Description |     Status |
----------------------------------------------------------------------------
|    1 |    Physics |  2022-03-10 |     Complete record book. |    Pending |
----------------------------------------------------------------------------
```



```
1. Create a task
2. List tasks
        a. Open tasks
        b. Open tasks of a category
        c. Open tasks before deadline
        d. Completed tasks
        e. All tasks
3. Update a task
        a. Modify attribute
        b. Mark as completed
        c. Mark as pending
4. Generate report
5. Exit the application

Enter your choice (eg. 1, 2c, 3b, etc.): 2c

Enter the deadline in yyyy-mm-dd format: 2022-03-10
----------------------------------------------------------------------------
|   Id |   Category |    Deadline |               Description |     Status |
----------------------------------------------------------------------------
|    1 |    Physics |  2022-03-10 |     Complete record book. |    Pending |
|    2 |       Home |  2022-03-08 |           Buy vegetables. |    Pending |
----------------------------------------------------------------------------
```



```
1. Create a task
2. List tasks
        a. Open tasks
        b. Open tasks of a category
        c. Open tasks before deadline
        d. Completed tasks
        e. All tasks
3. Update a task
        a. Modify attribute
        b. Mark as completed
        c. Mark as pending
4. Generate report
5. Exit the application

Enter your choice (eg. 1, 2c, 3b, etc.): 4

task_report.csv has been generated!
```



```
1. Create a task
2. List tasks
        a. Open tasks
        b. Open tasks of a category
        c. Open tasks before deadline
        d. Completed tasks
        e. All tasks
3. Update a task
        a. Modify attribute
        b. Mark as completed
        c. Mark as pending
4. Generate report
5. Exit the application

Enter your choice (eg. 1, 2c, 3b, etc.): 5

==================
+++ Thank you! +++
==================
```

VIII. Conclusion

The ToDo List application will help the users in efficiently managing their tasks. With features like report generation, the application will take no time in becoming popular. It even has the potential to be adopted by businesses for tracking their internal works and will help in task collaborations across teams.

IX. Bibliography

The following references were used:

- “Computer Science with Python” by Preeti Arora
    - www.amazon.in/dp/9387907961
- Google Search
    - www.google.com
- Official website of Python language
    - www.python.org
- Official website of MySQL database service
    - www.mysql.org