PROJECT ON A TASK MANAGEMENT APPLICATION USING PYTHON AND MYSQL

Project Submitted In Partial Fulfilment Of Practical Examination For Class XII, All India Senior Secondary Certificate Examination 2022 - 23, Science, Computer Science (083)

SUBMITTED TO:

NITHIYA R

SUBMITTED BY:

GOKULA RAMANAN

RAVICHANDRAN SELVARANI

SUGUNA PIP SCHOOL, COIMBATORE

CERTIFICATE

COMPUTER SCIENCE PROJECT

This is to certify that the Computer Science project report titled A TASK MANAGEMENT APPLICATION USING PYTHON AND MYSQL which has been submitted to SUGUNA PIP School, meeting the requirements of the CBSE Board practical examination for the academic year 2022 – 2023, is a bonafide work done by GOKULA RAMANAN RAVICHANDRAN SELVARANI

Signature of Principal

Teacher Incharge

Signature of Internal Examiner

Signature of External Examiner

DECLARATION

I, Gokula Ramanan Ravichandran Selvarani of class XII, hereby declare that the project titled " **'TASK MASTER', A TASK MANAGEMENT APPLICATION USING PYTHON AND MYSQL**", submitted to SUGUNA PIP SCHOOL, (Affiliation number-1930213), Nehru Nagar, Kalapatti Road, Coimbatore, Tamil Nadu - 641014, in regard to the class XII CBSE Board Practical Examination, is a record of original project work done by me, under the supervision of Ms. NITHIYA R, faculty in the Information Technology department, Suguna PIP School.

I further certify that:

1. The work contained in the report is original and has been done by me under the general supervision of the faculty.

2. The work has not been submitted to any other Institution/Board for class XII CBSE Board Practical Examination.

Student Name: Gokula Ramanan Ravichandran Selvarani

ID No: 4231

Class and Section: XII – A1

Date of Submission:

ACKNOWLEDGEMENT

I would like to thank Mr. POOVANNAN, Principal, Suguna PIP School, Coimbatore for his immeasurable direction towards course of action and support throughout the project.

I would also like to thank my Faculty Guide, Ms. NITHIYA R, for her valuable mentoring throughout the project. It was the interesting lectures of my knowledgeable teacher that helped me to understand the various concepts related to my PROJECT ON A TASK MANAGEMENT APPLICATION USING PYTHON AND MYSQL and see their presence in the actual world.

I got ample opportunity to do research, which enriched and broadened my knowledge and understanding of this area. I am seriously indebted to them.

Thanks to my parents, friends and everyone else who have been directly or indirectly supportive during the course of my project.

ABSTRACT

This is project is a MySQL server backed task management application with which we can create, edit, delete, view and store the user's tasks seamlessly. It uses a Command-Line Interface, has Python as the front-end language and uses database connectivity to connect to MySQL server.

The tasks are stored on the MySQL server, from which they are accessed and modified. Since they are stored on a database, they can be easily accessed remotely.

TABLE OF CONTENTS

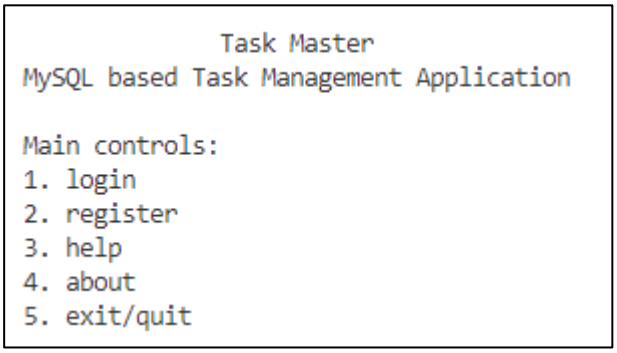
SNO	TITLE	Page No
1	Introduction	1
2	Code	3
3	Output	15
4	Scope for Further Enhancement	23
5	Conclusion	24
6	Bibliography	25

INTRODUCTION

TASKMASTER

A TASK MANAGEMENT APPLICATION

- Task Master is a task management application using which we can create, edit delete and store tasks on MySQL database.



(The command line interface of the main control page)

- This application supports multiple users with login and register menus. The data of all registered users is stored on the MySQL database "taskmaster" under the table "users". Each user's tasks are protected with a password created while registering and cannot be accessed without it.

- After successfully registering, a special key (denoted by variable skey), which is a unique number randomly generated between 0 – 100 is assigned to each user. This special key Skey, along with the username and password are added to the database "taskmaster". A unique table is created in the same database for each user automatically with the default name as: tasktable<Skey>

- Then, after logging in to the user space, a set of controls called, "user controls" are shown. These include: new, edit, view, delete, clear, logout, docs and help.

    1. New - add new tasks
    2. Edit – edit existing tasks

3. View – view the tasks stored on the database as a table
4. Delete – delete existing tasks
5. Clear – clear all the tasks
6. Logout – logout of the current use session
7. Docs – details about the project
8. Help – opens up help with controls

- Concepts used are:

1) Python (An interpreted, high-level object-oriented programming language):
    a) Basic python looping constructs – for and while loop
    b) List functions
    c) User defined functions
2) MySQL:
    a) Simple queries
    b) MySQL interface with python –
        i. PyMySQL – purely python-based MySQL-Python connector
        ii. Cursor object - <cursorobject> = <connectionobject>.cursor()
        iii. Commit - <connectionobject>.commit()
        iv. <cursorobject>.execute()

- Modules used are:
    1. **pymysql** - purely python-based MySQL-Python connector

    2. **time –** Delays execution for a given number of seconds. The argument may be a floating point number for subsecond precision.

    3. **tqdm -** A third-party module from Python Package Index (PyPI) used to decorate an iterable object, returning an iterator which acts exactly like the original iterable, but prints a dynamically updating progressbar every time a value is requested.

    4. **tabulate** - A third-party module from Python Package Index (PyPI), used to print a table from a nested list or a nested tuple.

    5. **random** - Random variable generators.

Code

```
import pymysql as pym # connecting to mysql server

import time # for progressbar animation
from tqdm import tqdm # for progressbar animation

from tabulate import tabulate # for printing mysql table in
python
import random

con = pym.connect(host="localhost", user="root",
passwd="mypass", database="taskmaster")
cur = con.cursor()
cur.execute("show tables")


def progressbar():
    for r in tqdm([1, 2]):
        time.sleep(0.5)
    return progressbar


def faultyprogressbar():
    for r in tqdm([1, 2]):
        if r > 1:
            break
        time.sleep(0.5)
    return faultyprogressbar


def helpm():

    print("-> Main controls: \n")
    print("I. login - to login to your account")
    print("II. register - to register you account")
    print("III. help - help with controls\n")
    print("IV. about - about the project")
```

```
        print("V. exit/quit - to quit the program\n")
        print("-> User-controls: \n")
        print("1. new - to add new tasks")
        print("2. edit - to edit existing task(s)")
        print("3. view - to view your task in the form of a table")
        print("4. delete - to delete your task")
        print("5. clear - to clear all tasks")
        print("6. logout - to log out of your account")
        print("7. docs - to view the brief documentation")
        print("8. help - to use the help command")
        print()
        return helpm

    list_tables = []
    for table_name in cur:
        list_tables += table_name
    if "users" in list_tables:
        pass
    else:
        cur.execute(
            "create table users (SKey integer(100) primary key not
    null, username varchar(200) not null, passwd char(200) not
    null)"
        )  # Skey (special key) is a unique identity number mapped
    to a specific user; a random number generated between 1 and 100
    print("\n               Task Master")
    print("MySQL based Task Management Application\n")
    print("Main controls: ")
    print("1. login")
    print("2. register")
    print("3. help")
    print("4. about")
    print("5. exit/quit\n")
    emaster = 1
    while emaster > 0:
        print("=================")
        input1 = input("Enter main control: ")
        print("=================\n")
        if input1 in ["1", "login"]:
            # print("---------------------------------")
```

```
        print(
            "---------------------------------
\n                Login\n-----------------------------------\n"
        )
        usern = input("Enter username: ")
        passww = input("Enter password: ")
        print()
        cur.execute("Select username,passwd from users")
        listuser = []
        for i in cur:
            listuser += i
        # print(listuser)
        if usern in listuser and passww in listuser:
            if listuser.index(usern) == (listuser.index(passww)
- 1):
                progressbar()
                print("\nLogin successful\n")
                str1 = "select Skey from users where username =
'{}'".format(usern)
                cur.execute(str1)
                for j in cur:
                    l = j
                skey = int(
                    str(j)[1:][::-1][2:][::-1]
                )  # for eg, l = (21,) => int(l) is giving
error, so this kind of conversion
                print(
                    "Welcome back :-)\n\nUsername: ",
                    usern,
                    "\nSpecial key: ",
                    skey,
                    "\n",
                )
                print("Your tasks are shown below\n")
                cur.execute("show tables")
                list_tables = []
                for table_name in cur:
                    list_tables += table_name
                # print(a)
                tablename = "tasktable" + str(skey)
```

```
                if (tablename in list_tables):  # assigning one
tasktable to each user. Each table has the table name as
"tasktable<skey>"
                    pass
                else:
                    str2 = (
                        "create table "
                        + tablename
                        + " (SNo integer(20) primary key not
null, Task varchar(40) not null, Priority char(10) not null)"
                    )
                    cur.execute(str2)

                def view():
                    str5 = "select * from " + tablename
                    cur.execute(str5)
                    data = cur.fetchall()
                    count = cur.rowcount
                    print(
                        tabulate(
                            data,
                            headers=["SNo", "Task",
"Priority"],
                            tablefmt="grid",
                            stralign="center",
                        )
                    )
                    return view

                view()
                str8high = (
                    "select count(Task) from " + tablename +
"where Priority = 'high'"
                )

                print(
                    "\nControls: \n 1. new (New Task) \n 2.
edit (Edit tasks) \n 3. view (View Tasks) \n 4. delete (Delete
tasks) \n 5. clear (Clear all your tasks) \n 6. logout \n 7.
doc (Docs)\n 8. help\n"
```

```
                )
                ecurrentuser = 1
                while ecurrentuser > 0:
                    print("================")
                    control = input("Enter user control: ")
                    print("================\n")
                    # control - 4
                    if control in ["4", "delete", "del"]:
                        edel = 1
                        print("Delete command selected, your
tasks are shown below")
                        view()
                        while edel > 0:
                            sinput10 = int(
                                input("Enter the serial no of
the task to delete: ")
                            )
                            str7 = "select SNo from " +
tablename
                            cur.execute(str7)
                            snolist = []
                            for q in cur:
                                snolist += q
                            if sinput10 in snolist:
                                str8 = "delete from {} where
Sno = {}".format(
                                    tablename, sinput10
                                )
                                cur.execute(str8)
                                print("Task deleted
succesfully...")
                            else:
                                print("Given sno does not
exist. Try again.")
                            input11 = input("Continue deleting
tasks? (y/n): ")
                            if input11 in ["y", "yes"]:
                                continue
                            elif input11 in ["n", "no"]:
```

```
                                print("Returning to user
menu...\n")
                                edel = 0
                            con.commit()
                    # control - 5
                    elif control in ["clear", "5"]:
                        print()
                        input12 = input("Do you want to clear
all your tasks? (y/n): ")
                        if input12 in ["y", "yes"]:
                            str9 = "truncate table " +
tablename
                            cur.execute(str9)
                            print("All tasks cleared")
                            print("Returing to user menu...\n")
                        elif input12 in ["n", "no"]:
                            print("Returing to user menu...\n")
                    # control - 1
                    elif control in ["new", "New", "1"]:
                        enew = 1
                        loopconstant = (
                            1  # just the number of times the
loop has been executed
                        )
                        while enew > 0:
                            task = input("Enter new task: ")
                            str10 = "select * from "+tablename
                            cur.execute(str10)
                            sno = cur.rowcount
                            e = 1
                            # while e>0:
                            input3 = input(
                                "Enter priority level (i.e. 1
for low  2 for medium  3 for high):"
                            )
                            if input3 == "1":
                                priority = "low"

                            elif input3 == "2":
                                priority = "medium"
```

```
                            elif input3 == "3":
                                priority = "high"
                            else:
                                print("Incorrect input. Try
again.\n")
                            values = (
                                "insert into "
                                + tablename
                                + " (SNo,Task,Priority) values
({},'{}','{}')".format(
                                    sno + 1, task, priority
                                )
                            )
                            cur.execute(values)
                            input4 = input("Continue adding new
tasks? (y/n): ")
                            if input4 in ["y", "yes"]:
                                continue
                                loopconstant += 1
                            elif input4 in ["n", "no"]:
                                if loopconstant == 1:
                                    print(
                                        "Query successfully
executed. Task added to database.\n"
                                    )
                                    break
                                else:
                                    print(
                                        "Quries successfully
executed. "
                                        + str(loopconstant)
                                        + " tasks added to
database."
                                    )
                                    print("Returning to user
menu...\n")
                                    enew = 0
                            con.commit()
                    # control - 2
```

```
                    elif control in ["edit", "2"]:
                        print("Edit command selected\n")
                        view()
                        print()
                        eedit = 1
                        while eedit > 0:
                            loopconstant = 1
                            sno = int(input("Enter the serial
number of the task: "))
                            task = input("Enter new task: ")
                            input3 = input(
                                "Enter priority level (i.e. 1
for low  2 for medium  3 for high):"
                            )
                            eedit2 = 1
                            while eedit2 > 0:
                                if input3 == "1":
                                    priority = "low"
                                    eedit2 = 0
                                elif input3 == "2":
                                    priority = "medium"
                                    eedit2 = 0
                                elif input3 == "3":
                                    priority = "high"
                                    eedit2 = 0
                                else:
                                    print("Incorrect input. Try
again.\n")
                            str4 = (
                                "update "
                                + tablename
                                + " set Task = '{}',
Priority='{}' where SNo= {}"
                            ).format(task, priority, sno)
                            cur.execute(str4)
                            input5 = input("Continue editing
tasks? (y/n): ")
                            if input5 in ["y", "yes", "ye",
"Y", "Yes", "Ye"]:
                                enew = 1
```

```
                                loopconstant += 1
                            else:
                                if loopconstant == 1:
                                    print(
                                        "Query successfully
executed. Task modified in the database."
                                    )

                                else:
                                    print(
                                        "Quries successfully
executed. "
                                        + str(loopconstant)
                                        + " tasks modified in
the database."
                                    )
                                print("Returning to user
menu...\n")
                                eedit = 0
                    # control - 3
                    elif control in ["view", "3"]:
                        print("View command selected\n")
                        view()
                        print()
                    # control - 6
                    elif control in ["6", "logout"]:
                        input13 = input("Please confirm to
logout (y/n): ")
                        if input13 in ["y","yes"]:
                            progressbar()
                            print("Logged out successfully.\n")
                            ecurrentuser = 0
                        else:
                            print("Logout aborted...\nReturning
to user menu\n")
                    # control - 7
                    elif control in ["7", "doc", "docs"]:
                        print(
```

```
                            "--------------\nPython Project\n--
------------\n\n Title: Task Master, a MySQL based Task
Management Application"
                        )
                        print(
                            " Name: Gokula Ramanan R S\n Class
and Section: XII A1\n ID No: 4231\n School: Suguna PIP School,
Coimbatore\n"
                        )
                    # control - 7
                    elif control in ["8", "help"]:
                        print("Help menu\n")
                        helpm()
        else:
            faultyprogressbar()
            print("\n :-( Username or password incorrect. Try
again. \n")
    elif input1 in ["2", "register"]:
        print(
            "---------------------------------
\n               Register\n---------------------------------\n"
        )
        # print("Register menu")
        print("Requirements: username and password must atleast
have 4 characters\n")
        eregister = 1
        while eregister > 0:
            input8 = input("Enter a username: ")
            input9 = input("Enter a password: ")
            cur.execute("select username from users")
            userlist = []
            for o in cur:
                userlist += o
            if len(input8) >= 4 and len(input9) >= 4:
                if input8 not in userlist:
                    eregister = 0
                else:
                    print("Username already exists. Try
again.\n")
            else:
```

```
                print("Username or password not meeting the
requirements. Try again.\n")
        while True:
            skeygenerator = random.randint(0, 100)
            cur.execute("select SKey from users")
            listkey = []
            for p in cur:
                listkey += p
            if skeygenerator not in listkey:
                break
        str6 = "insert into users (SKey,username,passwd)
values({},'{}','{}')".format(skeygenerator, input8, input9)
        cur.execute(str6)
        print()
        progressbar()
        print("\nNew user successfully created. Redirecting to
main menu....\n")
    elif input1 in ["quit", "exit", "5"]:
        print("Quitting the program")
        progressbar()
        emaster = 0
    elif input1 in ["3", "help"]:
        print("Help menu\n")
        helpm()
    elif input1 in ['about',"4"]:
        print(
                            "--------------\nPython Project\n--
------------\n\n Title: Task Master, a MySQL based Task
Management Application"
                        )
        print(
                            " Name: Gokula Ramanan R S\n Class
and Section: XII A1\n ID No: 4231\n School: Suguna PIP School,
Coimbatore\n"
                        )
    else:
        print("Control not recognized!\n")
        a88 = input("Want help with controls? (y/n): ")
        print()
        if a88 in ["y", "yes", "Yes"]:
```

```
            print()
            helpm()
        else:
            continue
con.commit()
con.close()
```

OUTPUT

```
              Task Master
MySQL based Task Management Application

Main controls:
1. login
2. register
3. help
4. about
5. exit/quit

=================
Enter main control: 2
=================

--------------------------------
               Register
--------------------------------

Requirements: username and password must atleast have 4 characters

Enter a username: test0
Enter a password: pass0

100%|████████████████████████████████████████| 2/2 [00:01<00:00,  1.98it/s]

New user successfully created. Redirecting to main menu....

=================
Enter main control: 1
=================

--------------------------------
               Login
--------------------------------

Enter username: test0
Enter password: pass0

100%|████████████████████████████████████████| 2/2 [00:01<00:00,  1.97it/s]

Login successful

Welcome back :-)

Username:  test0
Special key:  60
```

```
================
Enter main control: 1
================

--------------------------------
              Login
--------------------------------

Enter username: test0
Enter password: pass0

100%|████████████████████████████████████████| 2/2 [00:01<00:00,  1.97it/s]

Login successful

Welcome back :-)

Username:  test0
Special key:  60

Your tasks are shown below

+-------+--------+------------+
|  SNo  |  Task  |  Priority  |
+=======+========+============+
+-------+--------+------------+

Controls:
 1. new (New Task)
 2. edit (Edit tasks)
 3. view (View Tasks)
 4. delete (Delete tasks)
 5. clear (Clear all your tasks)
 6. logout
 7. doc (Docs)
 8. help

===============
Enter user control:
```

```
mysql> select * from users;
+------+-----------+--------+
| SKey | username  | passwd |
+------+-----------+--------+
|   71 | ravigokul | mykey  |
+------+-----------+--------+
1 row in set (0.00 sec)
```

Before registering new user test0

```
===============
Enter user control: 1
===============

Enter new task: sample1
Enter priority level (i.e. 1 for low  2 for medium  3 for high):2
Continue adding new tasks? (y/n): y
Enter new task: sample2
Enter priority level (i.e. 1 for low  2 for medium  3 for high):1
Continue adding new tasks? (y/n): n
Query successfully executed. Task added to database.

===============
Enter user control: 3
===============

View command selected

+-------+---------+------------+
|   SNo |  Task   |  Priority  |
+=======+=========+============+
|     1 | sample1 |   medium   |
+-------+---------+------------+
|     2 | sample2 |    low     |
+-------+---------+------------+
```

```
mysql> select * from users;
+------+-----------+--------+
| SKey | username  | passwd |
+------+-----------+--------+
|   60 | test0     | pass0  |
|   71 | ravigokul | mykey  |
+------+-----------+--------+
2 rows in set (0.00 sec)

mysql> select * from tasktable60;
+-----+---------+----------+
| SNo | Task    | Priority |
+-----+---------+----------+
|   1 | sample1 | medium   |
|   2 | sample2 | low      |
+-----+---------+----------+
2 rows in set (0.00 sec)
```

```
================
Enter user control: 2
================

Edit command selected

+-------+---------+------------+
|   SNo |  Task   |  Priority  |
+=======+=========+============+
|     1 | sample1 |   medium   |
+-------+---------+------------+
|     2 | sample2 |    low     |
+-------+---------+------------+

Enter the serial number of the task: 1
Enter new task: sample11
Enter priority level (i.e. 1 for low  2 for medium  3 for high):3
Continue editing tasks? (y/n): n
Query successfully executed. Task modified in the database.
Returning to user menu...

================
Enter user control: view
================

View command selected

+-------+----------+------------+
|   SNo |   Task   |  Priority  |
+=======+==========+============+
|     1 | sample11 |    high    |
+-------+----------+------------+
|     2 | sample2  |    low     |
+-------+----------+------------+
```

```
mysql> select * from tasktable60;
+-----+----------+----------+
| SNo | Task     | Priority |
+-----+----------+----------+
|   1 | sample11 | high     |
|   2 | sample2  | low      |
+-----+----------+----------+
2 rows in set (0.00 sec)
```

```
================
Enter user control: 4
================

Delete command selected, your tasks are shown below
+-------+----------+------------+
|   SNo |   Task   |  Priority  |
+=======+==========+============+
|     1 | sample11 |    high    |
+-------+----------+------------+
|     2 | sample2  |    low     |
+-------+----------+------------+
Enter the serial no of the task to delete: 2
Task deleted succesfully...
Continue deleting tasks? (y/n): n
Returning to user menu...
```

```
mysql> select * from tasktable60;
+-----+----------+----------+
| SNo | Task     | Priority |
+-----+----------+----------+
|   1 | sample11 | high     |
+-----+----------+----------+
1 row in set (0.00 sec)
```

```
================
Enter user control: 5
================


Do you want to clear all your tasks? (y/n): y
All tasks cleared
Returing to user menu...

================
Enter user control: view
================

View command selected

+-------+--------+------------+
|  SNo  |  Task  |  Priority  |
+=======+========+============+
+-------+--------+------------+
```

```
mysql> select * from tasktable60;
Empty set (0.00 sec)
```

```
================
Enter user control: 7
================

--------------
Python Project
--------------

 Title: Task Master, a MySQL based Task Management Application
 Name: Gokula Ramanan R S
 Class and Section: XII A1
 ID No: 4231
 School: Suguna PIP School, Coimbatore

================
Enter user control: 8
================

Help menu

-> Main controls:

I. login - to login to your account
II. register - to register you account
III. help - help with controls

IV. about - about the project
V. exit/quit - to quit the program

-> User-controls:

1. new - to add new tasks
2. edit - to edit existing task(s)
3. view - to view your task in the form of a table
4. delete - to delete your task
5. clear - to clear all tasks
6. logout - to log out of your account
7. docs - to view the brief documentation
8. help - to use the help command
```

```
================
Enter user control: logout
================

Please confirm to logout (y/n): y
100%|████████████████████████████████████████| 2/2 [00:01<00:00,  1.97it/s]
Logged out successfully.
```

```
=================
Enter main control: exit
=================

Quitting the program
100%|████████████████████████████████████████| 2/2 [00:01<00:00,  1.99it/s]
PS C:\Users\ravig>
```

- The main controls 3 (about) and 4(help) are the same as user controls 7 (docs) and 8 (help).
- If suppose the username or the password entered is incorrect, the following message is displayed:

```
=================
Enter main control: 1
=================

---------------------------------
               Login
---------------------------------

Enter username: tes0
Enter password: sw

 50%|████████████████████████                | 1/2 [00:00<00:00,  1.94it/s]

 :-( Username or password incorrect. Try again.
```

- The user space (after creating tasks) is displayed as:

```
---------------------------------
               Login
---------------------------------

Enter username: test0
Enter password: pass0

100%|████████████████████████████████████████| 2/2 [00:01<00:00,  1.97it/s]

Login successful

Welcome back :-)

Username:  test0
Special key:  60

Your tasks are shown below
```

```
+-------+------------+------------+
|   SNo |    Task    |  Priority  |
+=======+============+============+
|     1 |   task1    |    high    |
+-------+------------+------------+
|     2 |   task2    |    low     |
+-------+------------+------------+
|     3 |   task3    |   medium   |
+-------+------------+------------+
|     4 |   task4    |   medium   |
+-------+------------+------------+
|     5 | sample1212 |   medium   |
+-------+------------+------------+

Controls:
 1. new (New Task)
 2. edit (Edit tasks)
 3. view (View Tasks)
 4. delete (Delete tasks)
 5. clear (Clear all your tasks)
 6. logout
 7. doc (Docs)
 8. help
```

SCOPE FOR FURTHER ENHANCEMENT

- Using Graphical User Interface instead of Command Line Interface.
- Adding reminders and notification alerts.

CONCLUSION

Task Master, a task management application using Python and MySQL effectively stores the user's tasks in an organised way. The application offers users a number of features to modify and manipulate their tasks.

Having their tasks organised can help users make effective use of their time, plan properly and help them achieve the goals that they set.

BIBLIOGRAPHY

- Sumita Arora, Computer Science with Python, 13th ed., Textbook for class XII (2022), Dhanpat Rai and Co., (ISBN 978-81-7700-236-2)
- https://pypi.org/project/tabulate/
- https://pypi.org/project/tqdm/