



GEMS
United Indian School
ABU DHABI

PROJECT RECORD

2022-23

WORK MANAGEMENT SYSTEM

Roll No. _____

Name: _____

Class: _____ Sec: _____

Subject: _____





Certificate

Roll No. _____

This is to certify that
Miss/Master _____ of grade
_____ section _____ has carried
out project work in Assignment prescribed by the Central
Board of Secondary Education, New Delhi during the
academic year 2021-2022.

Teacher-in charge: **Dr Harini Priyadharshini**

Date: _____

External Examiner

Internal Examiner

CONTENTS

ACKNOWLEDGEMENTS	4
INTRODUCTION.....	5
SYNOPSIS	6
PROJECT DESCRIPTION	7
PROGRAM CODING	11
OUTPUT	19
CONCLUSION	25
BIBLIOGRAPHY	26

ACKNOWLEDGEMENTS

**“We must find time to stop and thank the people who make a difference in our lives.”
- John F Kennedy.**

While bringing out my project idea to its final form, I came across a number of individuals whose invaluable support in various ways helped my research. I am extremely privileged to have this constant system of support surrounding me and I would like to take this opportunity to extend my sincere gratitude to all of them.

Firstly, I would like to express my deep gratitude to Dr Harini Priyadharsini, the lighthouse that guided me from inception of the project to its final stages. I owe all my gratitude to her for being my mentor and enriching my project with her valuable inputs.

My heartfelt gratitude to my parents without whose constant support and the encouragement given to me, I would not be able to excel in all spheres.

I am highly indebted to the school faculty, especially our Senior Leadership Team, for providing me with the facilities required during the making of this project.

Last but not least, a warm thank you to my dear peers who I was fortunate enough to have as a guidance throughout the project.

INTRODUCTION

In today's hectic pace of life, our thoughts are racing to get our attention. Tasks to be done, events to attend and places to go crowd up in our head. Sometimes through the small cracks, it slips and we fail to achieve our target for the day.

It's hard enough managing your own schedule, let alone balancing team resources between projects and clients – and contemporary ways of working throw even more challenges into the blend, making collaboration increasing virtual, globally distributed and out-of-house.

But failing to get your team in order comes with serious operational and cultural consequences, with 86% of employees and executives citing a lack of collaboration or ineffective communication for workplace failures.

Fortunately, having a team planner can enhance the experience of a collaborative workspace ensuring 100% productivity. With my project, I aim to develop a software system designed to help employees to stay aligned with their colleagues, identify optimal times for collaboration and effectively manage team resources for future commitments.

SYNOPSIS

AllSeated is a planning and management tool, developed with accommodating the work and personal schedules of your employees. The application helps coordinate schedules and events to ensure everyone stays on the same page, and on-track to meet deadlines.

The application is developed using Python and SQL languages. It is a menu driven programme, with separate menus for admin and employee users. Each user/employee has a unique 6 digit identification number that helps the programme recognise the user.

Scope of work

USER SIGN IN
User signs in using password
- User is either admin or employee

ADMIN
Admin can add employees, resulting in creation of a new table in MySQL and hence, a new user
Admin can schedule a new meeting with expected attendees
Admin can view the work schedule of an employee to identify optimal times for collaboration
Admin can view the attendees of a particular meeting
Admin can change the password of any other user. A 6 digit password will be randomly generated upon request

USER
User can view their schedule for the day
User can track meetings scheduled for the day by adding on events in their to do list
User can modify details in their schedule
User can cross of tasks done and meetings attended and if all tasks aren't achieved, the tasks migrates to the next day
Users can prioritize events on their schedule in terms of importance
Users can change their password into a 10 character string.

PROJECT DESCRIPTION

PYTHON

FILE NAME: ALLSEATED

IMPORT STATEMENTS:

MODULE	USE
mysql.connector	To provide connectivity to MySQL servers
Datetime	To maintain records in a chronological manner
Random	To generate unique passwords and identification numbers
Tabulate	To give the data in a tabular format

BUILT IN FUNCTIONS:

	FUNCTION
1	mysql.connect()
2	is_connected()
3	cursor()
4	def <definition_name>()
5	print()
6	int()
7	input()
8	split()
9	execute()
10	fetchall()
11	commit()
12	tabulate()
13	range()

USER DEFINED FUNCTION:

	FUNCTION	USE
WHEN USER IS ADMIN:		
1	add_employee():	To add new employee
2	new_meeting()	To schedule new meeting
3	display_schedule()	To display schedule of an employee
4	count_attendees()	To display attendees of a meeting
5	change_pwd_admin()	To change the password of an user by the admin
WHEN USER IS EMPLOYEE:		
6	show_my_schedule()	To display the daily schedule of employee
7	update_schedule()	To add events/tasks to schedule
8	modify_schedule()	To modify the event details
9	delete_schedule()	To check off tasks done in a schedule
10	show_schedule()	To display number of personal/work events for a day
12	change_pwd()	To change password

VARIABLES:

	VARIABLE	USE
1	con	Assigned to the value of connection object
2	cursor	Assigned to the value of cursor object
3	EMP_ID	Assigned to the value of employee id, inputted by user
4	EMP_NAME	Assigned to the value of employee name, inputted by user
5	ADM_DATE	Assigned to the value of admission date of employee, inputted by user
6	DESIGNATION	Assigned to the value of designation of employee, inputted by user
7	DEPARTMENT	Assigned to the value of department the employee belongs to, inputted by user
8	PWD	Assigned to value to the password randomly generated
9	cont	Assigned to value of Yes/No, inputted by the user, to further iterate the code
10	EVENT	Assigned to the event title, inputted by the user
11	START	Assigned to the value of commencement time of event, inputted by the user

12	END	Assigned to the value of conclusion time of event, inputted by the user
13	P_OR_W	Assigned to the value of whether it is a personal or work event, inputted by the user
14	NOTES	Assigned to the value of notes, inputted by the user
15	update	Assigned to the insert query in line 45
16	display	Assigned to select query at line 54
17	rs	Assigned to the result set of select queries in MySQL throughout the code
18	headers	Assigned to an iterable object that contains the headers for tables throughout the code
19	emp_id	Assigned to the select query in line 85
20	emp	Assigned to the value of employee id inputted by user in line 91
21	I,j,k,m,n,rows,emp	Assigned to the elements of iterable object throughout the project
22	no_of_rows	Assigned to the result set of select query of line 171
23	choice	Assigned to value of Yes/No, inputted by the user, to further iterate the code
24	ch	Assigned to value of Yes/No, inputted by the user, to further iterate the code
25	done	Assigned to value of Yes/No, inputted by the user, to further proceed
26	delete	Assigned to the select query to delete content from table
27	truncate	Assigned to the select query to truncate the table
28	show	Assigned to select query in line 175
29	show_schedule	Assigned to select query in line 202
30	event	Assigned to the value of event to be changed, inputted by the user
31	change	Assigned to the value of change to be made, inputted by the user
32	enter_change	Assigned to the value of change to be modified, inputted by the user
33	modify	Assigned to update query in line 213
34	password	Assigned to the value of password, inputted by user
35	table	Assigned to an iterable object that contains the headers for tables throughout the code
36	pwd	Assigned to the result set of select query in line 264

MySQL

DATABASE: AllSeated

CREATE AND USE COMMANDS:

Creating database for programme

```
mysql> create database AllSeated;  
Query OK, 1 row affected (0.10 sec)
```

Using database for programme

```
mysql> use AllSeated  
Database changed
```

Create EMPLOYEE table for GEMS UIS

```
mysql> create table EMPLOYEE(EMP_ID int(10) Primary Key,EMP_NAME  
varchar(10) not null,ADM_DATE date);  
Query OK, 0 rows affected (0.18 sec)
```

TABLE DESCRIPTION:

TABLE: EMPLOYEE

```
mysql> desc employee;
```

Field	Type	Null	Key	Default	Extra
EMP_ID	int(10)	NO	PRI	NULL	
EMP_NAME	varchar(90)	YES		NULL	
ADM_DATE	date	YES		NULL	
DEPARTMENT	varchar(20)	YES		NULL	
designation	varchar(20)	YES		NULL	
PASSWORD	varchar(10)	YES		NULL	

6 rows in set (0.05 sec)

TABLE: <EMPLOYEE>_DAILY

```
mysql> desc shenooja_daily;
```

Field	Type	Null	Key	Default	Extra
EVENT	varchar(50)	YES		NULL	
START	varchar(10)	YES		NULL	
END	varchar(10)	YES		NULL	
P_or_W	char(2)	YES		NULL	
NOTES	varchar(200)	YES		NULL	
EMP_ID	int(10)	YES		10110	

PROGRAM CODING

```
import mysql.connector as m
con=m.connect(user='root', password='3v1yn', host='localhost',
database='ALLSEATED')
if con.is_connected():
    print('You have been connected.')
print("")
cursor=con.cursor()
from datetime import date,datetime
import random as r
from tabulate import tabulate
from colorama import init
from termcolor import colored
init()
import fontstyle
#+-----+

def add_employee():
    while True:
        print("Enter the following information. Please don't leave
any empty.")
        print("")
        EMP_ID=int(input(colored("Enter employee id:", "red")))
        EMP_NAME=input(colored("Enter employee name:", "red"))
        ADM_DATE=input(colored("Enter date of admission:", "red"))
        DESIGNATION=input(colored("Enter designation of employee:", "red"))
        DEPARTMENT=input(colored("Enter department:", "red"))
        print("")
        PWD=r.randint(100000,1000000)
        cursor.execute("insert into EMPLOYEE values({},'{}','{}','{}',
'{}','{}';".format(EMP_ID,EMP_NAME,ADM_DATE,DESIGNATION,DEPARTMENT
,PWD))
        con.commit()
        cursor.execute("create table %s_daily(EVENT varchar(50),START
varchar(10),END varchar(10),P_or_W char(2),),NOTES varchar(200),
EMP_ID int(10 DEFAULT '{}')".format(EMP_ID)%EMP_NAME.split()[0])
        con.commit()
        print("")
        print("Employee has been added. Welcome",EMP_NAME,"!")
        print(EMP_NAME, ", your AllSeated planner has been created. The
password for", EMP_NAME, "is", PWD, ".")
        cont=input("Do you wish to continue?")
        print("")
        if cont.lower()=='n':
            print("")
            print("Employee(s) have been added.")
            print("Going back to main menu...")
            break
    print("")

#+-----+
```

```

def update_schedule():
    while True:
        EVENT=input("Event title:")
        START=input("It starts at ")
        END=input("It ends at ")
        P_or_W=input("Personal(P)/Work(W):")
        NOTES=input("Additional information to note")
        update="insert into %s_daily values('{}','{}','{}','{}','{}');".
        format(EVENT,START,END,P_or_W,NOTES)%i[1].split()[0]
        cursor.execute(update)
        con.commit()
        print("Event has been successfully added.")
        print("")
        cont=input("Do you wish to continue? Press n to discontinue and
        any key, otherwise.")
        if cont.lower()=='n':
            print("Your schedule is as follows:")
            print("-----")
            display="select * from %s_daily;%i[1].split()[0]
            cursor.execute(display)
            rs=cursor.fetchall()
            headers=('EVENT TITLE','START','END','PERSONAL/WORK','NOTES')
            rs.insert(0,headers)
            print(tabulate(rs, headers='firstrow', tablefmt='fancy_grid'))
            print("")
            print(i[1],", have a productive day ahead with AllSeated.")
            print("Going back to main menu...")
            break

```

#+-----+

```

def new_meeting():
    while True:
        attendee=input("Enter name of attendee:")
        EVENT=input("Event title:")
        START=input("It starts at ")
        END=input("It ends at ")
        NOTES=input("Additional information to note:")
        update="insert into %s_daily values('{}','{}','{}','W','{}');".
        .format(EVENT,START,END,NOTES)%attendee.split()[0]
        cursor.execute(update)
        con.commit()
        print("Event has been successfully added.")
        cont=input("Do you wish to continue?")
        if cont.lower()=='n':
            print("All have been invited to be seated for meeting
            at",START, ".")
            print("Going back to main menu...")
            break

```

#+-----+

```

def display_schedule():
    print("Choose employee from below")
    print("-----")
    emp_id="select EMP_ID,EMP_NAME from EMPLOYEE;"
    cursor.execute(emp_id)
    rs=cursor.fetchall()
    headers=('Employee ID','Employee Name')
    rs.insert(0,headers)
    print(tabulate(rs, headers='firstrow', tablefmt='fancy_grid'))
    emp=int(input("Enter employee id:"))
    display="select EMP_NAME from EMPLOYEE where EMP_ID='{ }'".format(emp)
    cursor.execute(display)
    rs=cursor.fetchall()
    for m in rs:
        for n in m:
            print(n,"'s schedule")
            print("-----")
            cursor.execute("select * from %s_daily where P_or_W='W'"
                             %n.split()[0])
            rs=cursor.fetchall()
            headers=('EVENT TITLE','START','END','PERSONAL/WORK','NOTES')
            rs.insert(0,headers)
            print(tabulate(rs, headers='firstrow', tablefmt='fancy_grid'))
            print("")
#-----+

def show_my_schedule():
    display= "select * from %s_daily;%i[1].split()[0]"
    cursor.execute(display)
    rs=cursor.fetchall()
    headers=('EVENT TITLE','START','END','PERSONAL/WORK','NOTES')
    rs.insert(0,headers)
    print(tabulate(rs, headers='firstrow', tablefmt='fancy_grid'))
    print("")
#-----+

def change_pwd():
    old_pwd=input("Please enter your old password:")
    cursor.execute("select * from EMPLOYEE;")
    pwd=cursor.fetchall()
    for i in pwd:
        if i[5]==old_pwd:
            new_pwd=input("Please enter new password: ")
            cursor.execute("update EMPLOYEE set password='{ }' where
                             EMP_ID={ }".format(new_pwd,i[0]))
            con.commit()
            print("Your password has been changed. Please try logging
                  in.")

```

```

def change_pwd_admin():
    print("Choose employee from below")
    print("-----")
    emp_id="select EMP_ID,EMP_NAME from EMPLOYEE;"
    cursor.execute(emp_id)
    rs=cursor.fetchall()
    headers=('Employee ID','Employee Name')
    rs.insert(0,headers)
    print(tabulate(rs, headers='firstrow', tablefmt= 'fancy_grid'))
    EMP_ID=int(input( "Enter employee id:" ))
    cursor.execute("select EMP_NAME from EMPLOYEE where
EMP_ID={}" .format(EMP_ID))
    for i in cursor.fetchall():
        for emp in i:
            choice=input("You would like to change {} password. Is that
correct? (Y/N) ".format(emp))
            if choice.lower()=='y':
                new_pwd=r.randint(100000,1000000)
                old_pwd= "select PASSWORD from EMPLOYEE"
                cursor.execute(old_pwd)
                rs=cursor.fetchall()
                for pwd in rs:
                    for e_pwd in pwd:
                        if e_pwd==new_pwd:
                            new_pwd=r.randint(100000,1000000)
                        else:
                            continue
                cursor.execute("update EMPLOYEE set password='{}' where
EMP_ID={}" .format(new_pwd,EMP_ID))
                con.commit()
                print("")
                print(emp,"password has been changed. New password
is",new_pwd,".")
                print("Returning to main menu.....")
                print("-----")
            else:
                print("You have entered an incorrect employee id. Try
again later.")
                print("Returning to main menu.....")
                print("-----")

#-----+
def delete_schedule():
    global i
    choice= "Have you finished all tasks for today?(Y/N)"
    if choice.lower()=="y":
        truncate= "truncate %s_daily"%i[1].split()[0]
        cursor.execute(truncate)
        con.commit()
        print("Congratulations",i[1].split()[0],", you have successfully
completed all your tasks for today.")
    else:

```

```

        cursor.execute("select count(EVENT) from%s_daily"%i[1]
        .split()[0])
    for i in cursor.fetchall():
        for j in i:
            no_of_events=j
    while i>no_of_events:
        show="select EVENT from %s_daily;"%i[1].split()[0]
        cursor.execute(show)
        for k in cursor.fetchall():
            for j in k:
                done=input("Have you completed the task '{}':
                (Y/N)" .format(j))
                if done.lower()=="y":
                    delete="delete from %s_daily where EVENT='{}'".
                    format(j)%i[1].split()[0]
                    cursor.execute(delete)
                    con.commit()
                    print("Congratulations",i[1].split()[0], ", you
                    have completed",j,".")
                elif done.lower()=="n":
                    ch=input("Do you wish to delete task from your
                    planner? (Y/N)")
                    if ch.lower()=="y":
                        delete="delete from %s_daily where
                        EVENT='{}'".format(j)%i[1].split()[0]
                        cursor.execute(delete)
                        con.commit()
                    else:
                        print("The task has been added to your tasks
                        for tomorrow")
                        print("Returning to main menu.....")
                        print("-----")
            else:
                print("Sorry, the application fails to understand
                you. Please try again.")
        print(i[1].split()[0],", hope you had a productive day with
        AllSeated. See you tomorrow!")
#-----+
def show_schedule():
    global i
    show_schedule= "select P_or_W,count(*) from %s_daily group by P_or_W
    ;"%i[1].split()[0]
    cursor.execute(show_schedule)
    rs=cursor.fetchall()
    headers=('EVENT TITLE','START','END','PERSONAL/WORK','NOTES')
    rs.insert(0,headers)
    print(tabulate(rs, headers='firstrow', tablefmt='fancy_grid'))
#-----+

def modify_schedule():
    event=input("Enter event you would like to change:")
    change=input("What would you like to change?")
    enter_change=input("Enter your change here:")

```

```

modify="update %s_daily set {}='{}' where EVENT='{}'".format(change
,enter_change, event)%i[1].split()[0]
cursor.execute(modify)
con.commit()
print( "The change has been made successfully")
sql="select * from %s_daily where {}='{}'".format(change,enter_change)
cursor.execute(sql)
rs=cursor.fetchall()
headers=('EVENT TITLE','START','END','PERSONAL/WORK','NOTES')
rs.insert(0,headers)
print(tabulate(rs, headers='firstrow', tablefmt='fancy_grid'))
print("")
print("Returning to main menu.....")
print("-----")
#+-----+
def count_attendees():
    EVENT_TITLE=input( "Enter event you wish to display attendees:" )
    print("")
    select="select EMP_NAME from EMPLOYEE"
    cursor.execute(select)
    EMP_NAME={}
    for m in cursor.fetchall():
        for n in m:
            show="show tables"
            cursor.execute(show)
            EVENT=[]
            for i in cursor.fetchall():
                for j in i:
                    if i in j:
                        sql= "select EMP_ID,EVENT from %s where EVENT
                        = '{}'" .format(EVENT_TITLE)%j
                        cursor.execute(sql)
                        for a in cursor.fetchall():
                            for b in a:
                                EVENT.append(b)

    EVENT_ATTENDEES=EVENT[::2]
    ATTEENDEES=[]
    for i in EVENT_ATTENDEES:
        sql= "select EMP_NAME from EMPLOYEE where EMP_ID={} ".format(i)
        cursor.execute(sql)
        for a in cursor.fetchall():
            ATTEENDEES.append(a)
    ATTEENDEES.insert(0,["ATTENDEE NAME"])
    print(tabulate(ATTEENDEES, headers='firstrow', tablefmt='fancy_grid'))
    print("")
    print("There are",len(EVENT_ATTENDEES),"attendees for this meeting.")
    print('')
#+-----+
print(fontstyle.apply( " AllSeated","bold/italic/purple/"))
print(colored("Welcome to AllSeated, your ultimate work planner!","blue"))
print( "-----" )
print("")
password=str(input("Enter your password to continue:"))
print("")

```



```

if password=="root4242":
    print("Welcome Admin!")
    print("-----")
    print("Today is the",date.today())
    print("")
    while True:
        print( "Admin, how can we help you today?")
        table=[(1, "Add new employee"),(2, "Schedule new
meeting"),(3, "Display employee's schedule"),(4,"Display number of
attendees of a scheduled event"),(5, "Change employee
password"),(6, "Exit the programme")]
        print(tabulate(table))
        ch=int(input("Enter number corresponding to choice:"))
        print("")
        if ch==6:
            print("Thank you, have a productive day ahead with
AllSeated.")
            break
        elif ch==1:
            add_employee()
            print("")
        elif ch==2:
            new_meeting()
            print("")
        elif ch==3:
            display_schedule()
        elif ch==4:
            count_attendees()
        elif ch==5:
            change_pwd_admin()
            print("")
        else:
            print( "You have entered an incorrect choice. Please try again")

else:
    cursor.execute( "select * from EMPLOYEE;")
    pwd=cursor.fetchall()
    for i in pwd:
        if i[5]==password:
            print("Welcome",i[1],"!")
            print("-----")
            print("Today is the",date.today())
            while True:
                print("")
                print(i[1].split()[0], " , how can we help you today?")
                table=[(1,"View schedule"),(2, "Update
schedule" ),(3,"Modify your schedule"),(4, "Cross off
events concluded" ),(5, "Display schedule according to
priority"),(6, "Change your password" ),(7, "Exit the
programme")]
                print(tabulate(table))
                ch=int(input( "Enter number corresponding to choice:"))
                print("")
                if ch==7:

```

```

        print("Thank you, have a productive day ahead with
        AllSeated")
        break
    elif ch==1:
        print( "Your schedule for the day is as follows")
        show_my_schedule()
        print("")
    elif ch==2:
        update_schedule()
        print("")
    elif ch==3:
        modify_schedule()
        print("")
    elif ch==4:
        delete_schedule()
        print("")
    elif ch==5:
        show_schedule()
        print("")
    elif ch==6:
        change_pwd()
        break
    else:
        print("You have entered a wrong choice. Please try
        again.")
        print("")

```

OUTPUT

Admin User

You have been connected.

AllSeated

Welcome to All Seated, your ultimate work planner!

Enter your password to continue:root4242

Welcome Admin!

Today is the 2022-07-26

Admin, how can we help you today?

- -----
- 1 Add new employee
- 2 Schedule new meeting
- 3 Display employee's schedule
- 4 Display number of attendees of a scheduled event
- 5 Change employee password
- 6 Exit the programme
- -----

Enter number corresponding to choice:1

Enter the following information. Please don't leave any empty.

Enter employee id:10191

Enter employee name:Sanyo Thomas

Enter date of admission:2014-10-10

Enter designation of employee:PRO

Enter department:Administration

Employee has been added. Welcome Sanyo Thomas !

Sanyo Thomas , your AllSeated planner has been created. The password for Sanyo Thomas is 670164 .

Do you wish to continue?n

Employee(s) have been added.

Going back to main menu...

Admin, how can we help you today?

- -----
- 1 Add new employee
- 2 Schedule new meeting
- 3 Display employee's schedule
- 4 Display number of attendees of a scheduled event

```

5 Change employee password
6 Exit the programme
- -----
Enter number corresponding to choice:2

Enter name of attendee:Lalita Krishnamoorthy
Event title:Admin meeting
It starts at 13:00
It ends at 14:00
Additional information to note: Please report to admin office in A
Block
Event has been successfully added.
Do you wish to continue?y

Enter name of attendee:Sanyo Thomas
Event title:Admin meeting
It starts at 13:00
It ends at 14:00
Additional information to note:Please report to admin office in A
Block
Event has been successfully added.
Do you wish to continue?n

All have been invited to be seated for meeting at 13:00 .
Going back to main menu...

Admin, how can we help you today?
- -----
1 Add new employee
2 Schedule new meeting
3 Display employee's schedule
4 Display number of attendees of a scheduled event
5 Change employee password
6 Exit the programme
- -----
Enter number corresponding to choice:4

Enter event you wish to display attendees: HOD Meeting

```

ATTENDEE NAME
Shenooja Pareed
Tina Maria D'Souza

```

There are 2 attendees for this meeting.

Admin, how can we help you today?
- -----
1 Add new employee

```

- 2 Schedule new meeting
- 3 Display employee's schedule
- 4 Display number of attendees of a scheduled event
- 5 Change employee password
- 6 Exit the programme

- - - - -

Enter number corresponding to choice:5

Choose employee from below

Employee ID	Employee Name
10101	Shenooja Pareed
10136	Tina Maria DSouza
10141	Suja Wilson
10999	Evlyn Giju

Enter employee id:10999

You would like to change Evlyn Giju password. Is that correct?
(Y/N)Y

Evlyn Giju password has been changed. New password is 249560 .
Returning to main menu.....

- - - - -

Admin, how can we help you today?

- - - - -

- 1 Add new employee
- 2 Schedule new meeting
- 3 Display employee's schedule
- 4 Display number of attendees of a scheduled event
- 5 Change employee password
- 6 Exit the programme

- - - - -

Enter number corresponding to choice:6

Thank you, have a productive day ahead with AllSeated.

Employee User

AllSeated

Welcome to AllSeated, your ultimate work planner!

- - - - -

Enter your password to continue:571176

Welcome Shenooja Pareed !

- - - - -

Today is the 2022-09-04

Shenooja , how can we help you today?

- -----
- 1 View schedule
 - 2 Update schedule
 - 3 Modify your schedule
 - 4 Cross off events concluded
 - 5 Display schedule according to priority
 - 6 Change your password
 - 7 Exit the programme
- -----

Enter number corresponding to choice:1

Your schedule for the day is as follows

EVENT	START	END	PERSONAL/WORK	NOTES
HOD Meeting	11:00	12:00	W	-NIL-

Shenooja , how can we help you today?

- -----
- 1 View schedule
 - 2 Update schedule
 - 3 Modify your schedule
 - 4 Cross off events concluded
 - 5 Display schedule according to priority
 - 6 Change your password
 - 7 Exit the programme
- -----

Enter number corresponding to choice:2

Event title:Pulse Call with Parents

It starts at 18:00

It ends at 20:00

Personal(P)/Work(W):P

Additional information to note Talk about Gems Rewards

Event has been successfully added.

Do you wish to continue? Press any key to continue and 'n' to discontinue. n

Your schedule is as follows:

- -----

EVENT	START	END	PERSONAL/WORK	NOTES
HOD Meeting	11:00	12:00	W	-NIL-
Pulse Call with Parents	18:00	20:00	P	Talk about Gems Rewards

Shenooja Pareed , have a productive day ahead with AllSeated.

Going back to main menu...

Shenooja , how can we help you today?

- -----
- 1 View schedule
 - 2 Update schedule
 - 3 Modify your schedule
 - 4 Cross off events concluded
 - 5 Display schedule according to priority
 - 6 Change your password
 - 7 Exit the programme
- -----

Enter number corresponding to choice:4

Have you completed the task 'Staff Meeting': (Y/N)Y

Congratulations Shenooja , you have completed Staff Meeting .

Have you completed the task 'Corporate meeting': (Y/N)Y

Congratulations Shenooja , you have completed Corporate meeting .

Have you completed the task 'Class with 12 B2': (Y/N)N

Do you wish to delete task from your planner? (Y/N)N

The task has been added to your tasks for tomorrow

Have you completed the task 'Class with 11 B1': (Y/N)Y

Congratulations Shenooja , you have completed Class with 11 B1 .

Have you completed the task 'Department Meeting': (Y/N)y

Congratulations Shenooja , you have completed Department Meeting .

Have you completed the task 'Class with 10B2': (Y/N)y

Congratulations Shenooja , you have completed Class with 10B2 .

Have you completed the task 'Pulse Call ': (Y/N)y

Congratulations Shenooja , you have completed Pulse Call .

Shenooja , hope you had a productive day with AllSeated. See you tomorrow!

Shenooja , how can we help you today?

- -----
- 1 View schedule
 - 2 Update schedule
 - 3 Modify your schedule
 - 4 Cross off events concluded
 - 5 Display schedule according to priority
 - 6 Change your password
 - 7 Exit the programme
- -----

Enter number corresponding to choice:5

EVENT TITLE	START
P	2
W	1

Shenooja , how can we help you today?

- -----

- 1 View schedule
- 2 Update schedule
- 3 Modify your schedule
- 4 Cross off events concluded
- 5 Display schedule according to priority
- 6 Change your password
- 7 Exit the programme

- -----

Enter number corresponding to choice:6

Please enter your old password: 571176

Please enter new password:sh3n00ja

Your password has been changed. Please try logging in.

Shenooja , how can we help you today?

- -----

- 1 View schedule
- 2 Update schedule
- 3 Modify your schedule
- 4 Cross off events concluded
- 5 Display schedule according to priority
- 6 Change your password
- 7 Exit the programme

- -----

Enter number corresponding to choice:7

Thank you, have a productive day ahead with AllSeated

CONCLUSION

AllSeated effectively integrate your colleagues' calendars into one intelligent space, helping you visualize all work, coordinate schedules and plan resources thoughtfully across teams and projects.

When you work in an collaborative workspace a team planner can provide transparency and visibility in one's teams schedule. It also helps in identifying optimal times for collaboration. Delegation of tasks becomes easier in the work environment, ensuring no employee is overworked resulting in burn out. All Seated acts as your team's personal assistant by efficiently managing your team's resources and by avoiding time managing multiple, manually intensive team calendars.

AllSeated; For you and your team.

BIBLIOGRAPHY

	BOOKS REFERRED
1.	Arora, Preeti. <i>Computer Science with Python for CBSE Class XI</i> . DHANPAT RAI & Co., 2022.
2.	Arora, Preeti. <i>Computer Science with Python for CBSE Class XI</i> . DHANPAT RAI & Co., 2022.

	WEBSITES REFERRED
1.	Aggarwal, N., 2022. <i>Python datetime module - GeeksforGeeks</i> . [online] GeeksforGeeks. Available at: < https://www.geeksforgeeks.org/python-datetime-module/ >
2.	McKenzie, Paul, and Paul Collingwood. "Python Script To Do Something At The Same Time Every Day". <i>Stack Overflow</i> , 2022, https://stackoverflow.com/questions/15088037/python-script-to-do-something-at-the-same-time-every-day .
3.	Donovan, Ryan et al. "Getting Started With... Python". <i>Stack Overflow Blog</i> , 2022, https://stackoverflow.blog/2021/07/14/getting-started-with-python/ .