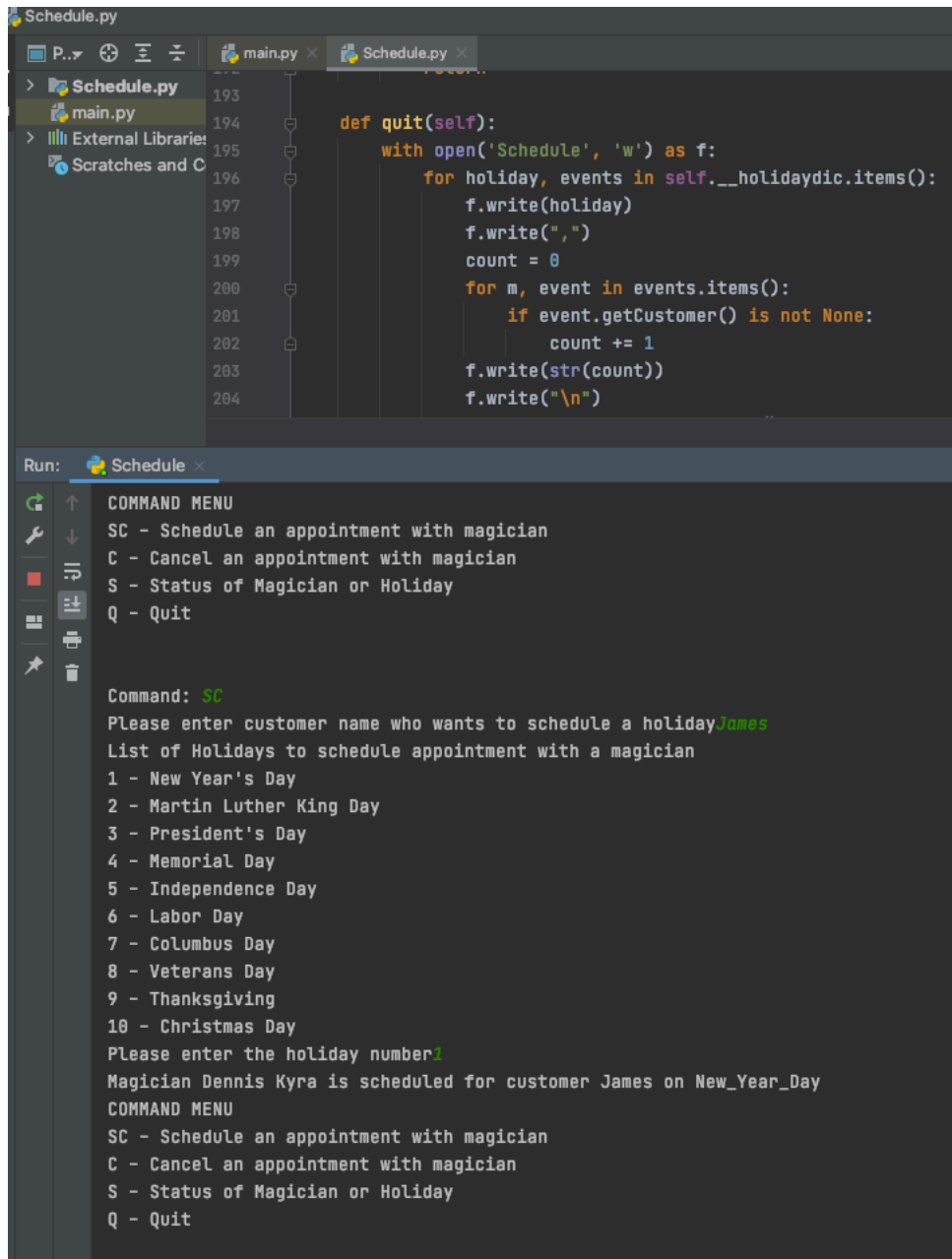


Execution Output

1. Scheduling (SC) for an appointment with a magician.



```
def quit(self):
    with open('Schedule', 'w') as f:
        for holiday, events in self.__holidaydic.items():
            f.write(holiday)
            f.write(",")
            count = 0
            for m, event in events.items():
                if event.getCustomer() is not None:
                    count += 1
            f.write(str(count))
            f.write("\n")
```

Run: Schedule

COMMAND MENU

SC - Schedule an appointment with magician

C - Cancel an appointment with magician

S - Status of Magician or Holiday

Q - Quit

Command: SC

Please enter customer name who wants to schedule a holidayJames

List of Holidays to schedule appointment with a magician

1 - New Year's Day

2 - Martin Luther King Day

3 - President's Day

4 - Memorial Day

5 - Independence Day

6 - Labor Day

7 - Columbus Day

8 - Veterans Day

9 - Thanksgiving

10 - Christmas Day

Please enter the holiday number1

Magician Dennis Kyra is scheduled for customer James on New_Year_Day

COMMAND MENU

SC - Schedule an appointment with magician

C - Cancel an appointment with magician

S - Status of Magician or Holiday

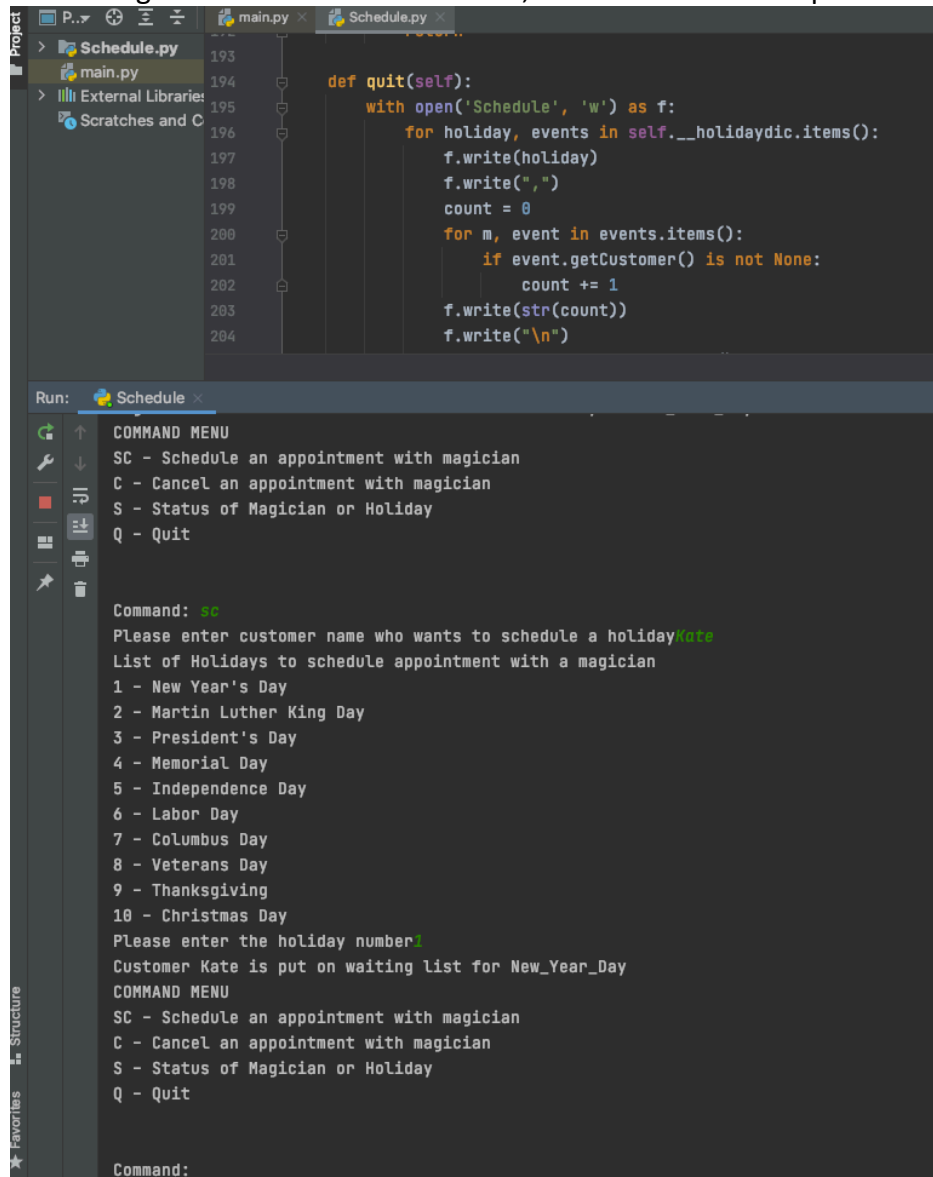
Q - Quit

2. Scheduling another customer with an magician

```
Command: SC
Please enter customer name who wants to schedule a holidayHarshinee
List of Holidays to schedule appointment with a magician
1 - New Year's Day
2 - Martin Luther King Day
3 - President's Day
4 - Memorial Day
5 - Independence Day
6 - Labor Day
7 - Columbus Day
8 - Veterans Day
9 - Thanksgiving
10 - Christmas Day
Please enter the holiday number1
Magician James Maxwell is scheduled for customer Harshinee on New_Year_Day
COMMAND MENU
SC - Schedule an appointment with magician
C - Cancel an appointment with magician
S - Status of Magician or Holiday
Q - Quit
```

Similarly scheduling for another 8 customers.

3. Scheduling for more than 10 customers, the customer will be put in waiting list.



The screenshot displays a Python IDE with two main panels. The top panel shows the code for the `quit` method in `Schedule.py`, which writes the current state of the program to a file named `Schedule`. The bottom panel shows the output of the program in the `Run` console, which includes a command menu and the execution of the `sc` command to schedule an appointment for a customer named Kate.

```
193
194     def quit(self):
195         with open('Schedule', 'w') as f:
196             for holiday, events in self.__holidaydic.items():
197                 f.write(holiday)
198                 f.write(",")
199                 count = 0
200                 for m, event in events.items():
201                     if event.getCustomer() is not None:
202                         count += 1
203                 f.write(str(count))
204                 f.write("\n")
```

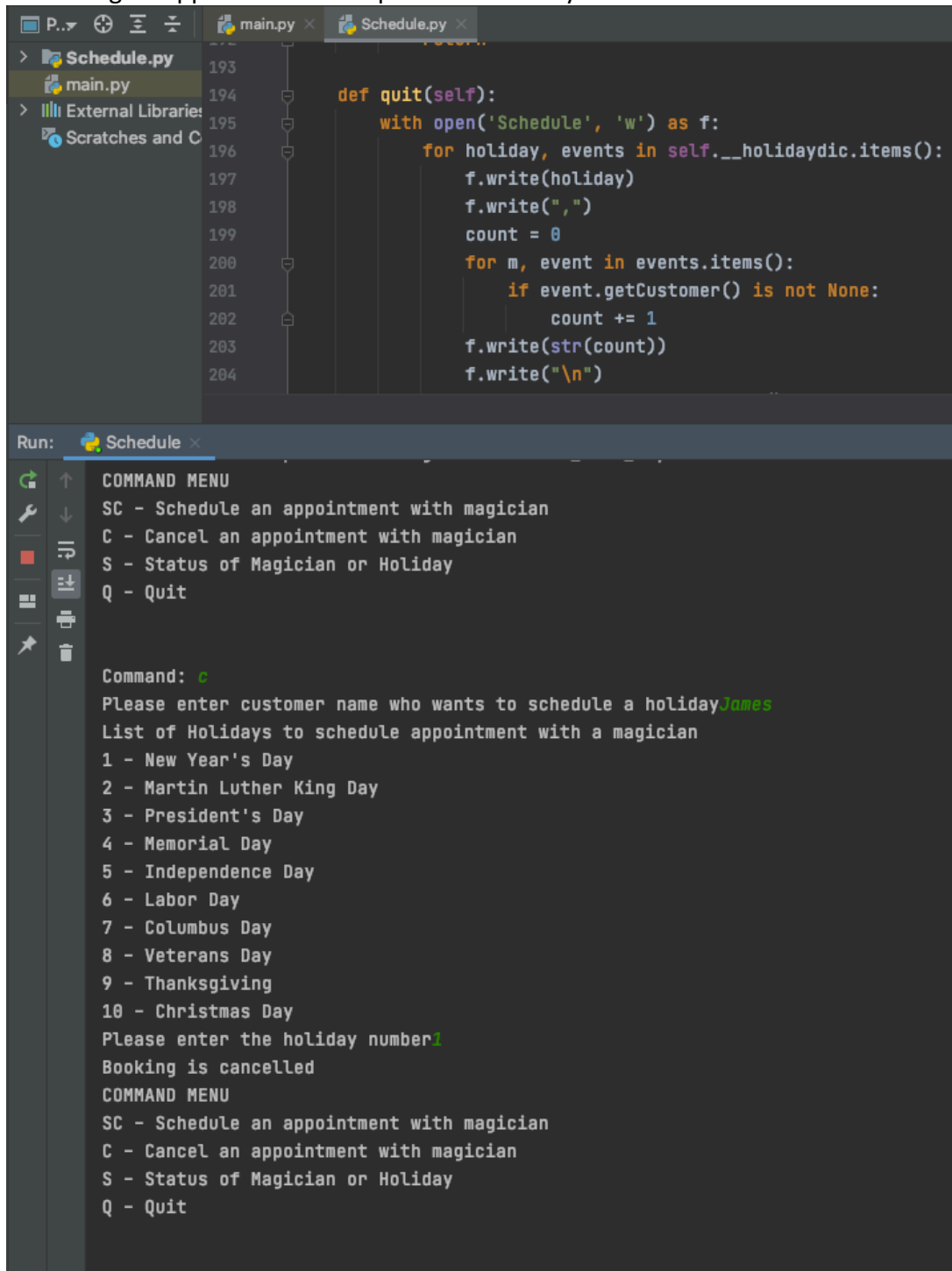
Run: Schedule x

COMMAND MENU
SC - Schedule an appointment with magician
C - Cancel an appointment with magician
S - Status of Magician or Holiday
Q - Quit

Command: **sc**
Please enter customer name who wants to schedule a holiday **Kate**
List of Holidays to schedule appointment with a magician
1 - New Year's Day
2 - Martin Luther King Day
3 - President's Day
4 - Memorial Day
5 - Independence Day
6 - Labor Day
7 - Columbus Day
8 - Veterans Day
9 - Thanksgiving
10 - Christmas Day
Please enter the holiday number **1**
Customer Kate is put on waiting list for New_Year_Day
COMMAND MENU
SC - Schedule an appointment with magician
C - Cancel an appointment with magician
S - Status of Magician or Holiday
Q - Quit

Command:

4. Cancelling an appointment for a particular holiday



The screenshot displays a Python IDE with two tabs: `main.py` and `Schedule.py`. The `Schedule.py` tab is active, showing the `quit` method implementation. The code iterates through `self.__holidaydic.items()` to write holiday names to a file named `Schedule`, then iterates through `events.items()` to count the number of appointments for each holiday, excluding those with a `None` customer. The `Run` console shows the program's execution, including a command menu, user input for a customer name (`James`) and a holiday number (`1`), and the resulting output: "Booking is cancelled".

```
def quit(self):
    with open('Schedule', 'w') as f:
        for holiday, events in self.__holidaydic.items():
            f.write(holiday)
            f.write(",")
            count = 0
            for m, event in events.items():
                if event.getCustomer() is not None:
                    count += 1
            f.write(str(count))
            f.write("\n")
```

Run: Schedule

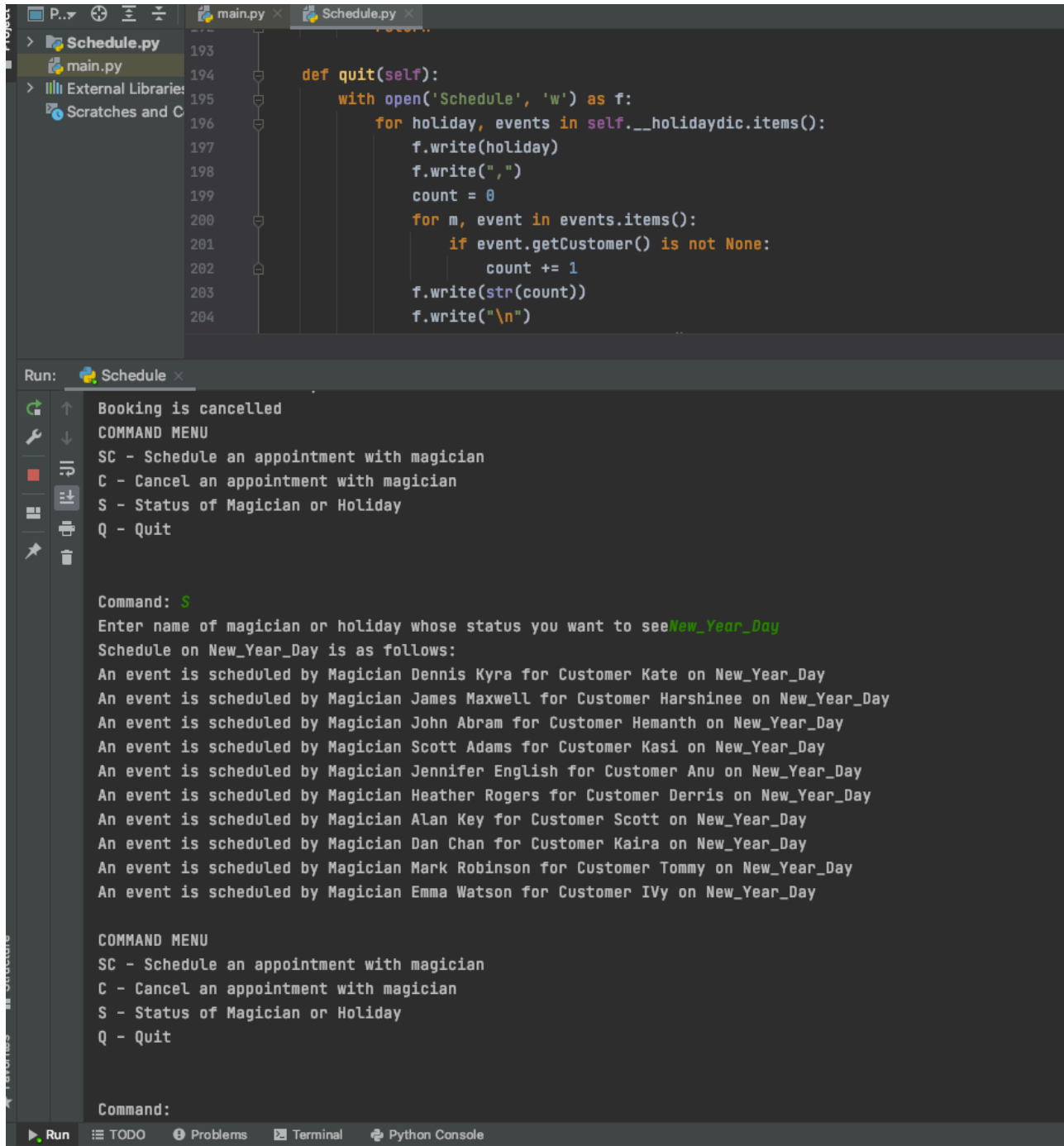
COMMAND MENU
SC - Schedule an appointment with magician
C - Cancel an appointment with magician
S - Status of Magician or Holiday
Q - Quit

Command: **c**
Please enter customer name who wants to schedule a holiday **James**
List of Holidays to schedule appointment with a magician
1 - New Year's Day
2 - Martin Luther King Day
3 - President's Day
4 - Memorial Day
5 - Independence Day
6 - Labor Day
7 - Columbus Day
8 - Veterans Day
9 - Thanksgiving
10 - Christmas Day
Please enter the holiday number **1**
Booking is cancelled
COMMAND MENU
SC - Schedule an appointment with magician
C - Cancel an appointment with magician
S - Status of Magician or Holiday
Q - Quit

5. Checking Status (S) of a particular holiday or magician.

First let's check for holiday

A person on waiting list is automatically scheduled for a magician if someone cancels.



The screenshot shows a Python IDE with two files: `main.py` and `Schedule.py`. The `Schedule.py` file contains a `quit` method that writes the current state of the `__holidaydic` and `events` to a file named `Schedule`. The `Run` console shows the output of the program, including a command menu and the status of the `New_Year_Day` holiday.

```
def quit(self):
    with open('Schedule', 'w') as f:
        for holiday, events in self.__holidaydic.items():
            f.write(holiday)
            f.write(",")
            count = 0
            for m, event in events.items():
                if event.getCustomer() is not None:
                    count += 1
            f.write(str(count))
        f.write("\n")
```

Run: Schedule x

Booking is cancelled
COMMAND MENU
SC - Schedule an appointment with magician
C - Cancel an appointment with magician
S - Status of Magician or Holiday
Q - Quit

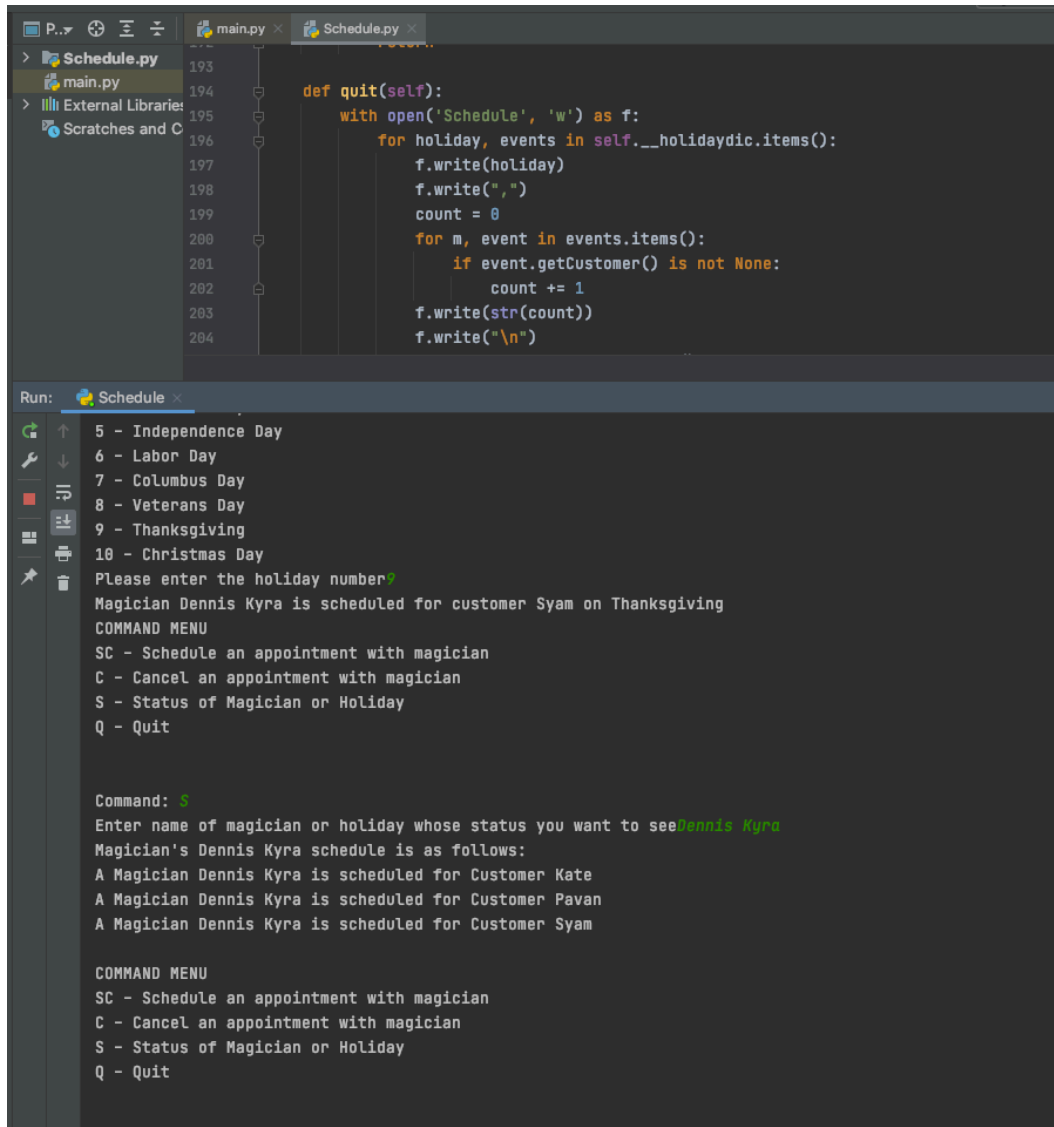
Command: S
Enter name of magician or holiday whose status you want to see **New_Year_Day**
Schedule on New_Year_Day is as follows:
An event is scheduled by Magician Dennis Kyra for Customer Kate on New_Year_Day
An event is scheduled by Magician James Maxwell for Customer Harshinee on New_Year_Day
An event is scheduled by Magician John Abram for Customer Hemanth on New_Year_Day
An event is scheduled by Magician Scott Adams for Customer Kasi on New_Year_Day
An event is scheduled by Magician Jennifer English for Customer Anu on New_Year_Day
An event is scheduled by Magician Heather Rogers for Customer Derris on New_Year_Day
An event is scheduled by Magician Alan Key for Customer Scott on New_Year_Day
An event is scheduled by Magician Dan Chan for Customer Kaira on New_Year_Day
An event is scheduled by Magician Mark Robinson for Customer Tommy on New_Year_Day
An event is scheduled by Magician Emma Watson for Customer IVy on New_Year_Day

COMMAND MENU
SC - Schedule an appointment with magician
C - Cancel an appointment with magician
S - Status of Magician or Holiday
Q - Quit

Command:

Similarly scheduling for more customers.

6. Checking Status (S) of a particular holiday or magician.
Let's check for magician



```
193
194
195
196
197
198
199
200
201
202
203
204
```

```
def quit(self):
    with open('Schedule', 'w') as f:
        for holiday, events in self.__holidaydic.items():
            f.write(holiday)
            f.write(",")
            count = 0
            for m, event in events.items():
                if event.getCustomer() is not None:
                    count += 1
            f.write(str(count))
            f.write("\n")
```

Run: Schedule

```
5 - Independence Day
6 - Labor Day
7 - Columbus Day
8 - Veterans Day
9 - Thanksgiving
10 - Christmas Day
Please enter the holiday number?
Magician Dennis Kyra is scheduled for customer Syam on Thanksgiving
COMMAND MENU
SC - Schedule an appointment with magician
C - Cancel an appointment with magician
S - Status of Magician or Holiday
Q - Quit

Command: S
Enter name of magician or holiday whose status you want to seeDennis Kyra
Magician's Dennis Kyra schedule is as follows:
A Magician Dennis Kyra is scheduled for Customer Kate
A Magician Dennis Kyra is scheduled for Customer Pavan
A Magician Dennis Kyra is scheduled for Customer Syam

COMMAND MENU
SC - Schedule an appointment with magician
C - Cancel an appointment with magician
S - Status of Magician or Holiday
Q - Quit
```

7. Quit – Saving the schedule and exiting the program

The screenshot shows a Python IDE with two tabs: `main.py` and `Schedule.py`. The `Schedule.py` tab is active, displaying the following code:

```
193
194
195
196
197
198
199
200
201
202
203
204
def quit(self):
    with open('Schedule', 'w') as f:
        for holiday, events in self.__holidaydic.items():
            f.write(holiday)
            f.write(",")
            count = 0
            for m, event in events.items():
                if event.getCustomer() is not None:
                    count += 1
            f.write(str(count))
            f.write("\n")
```

Below the code editor is the `Run:` console, which shows the output of the program. The output includes a list of holidays, a command menu, and the status of a magician named Dennis Kyra.

```
Run: Schedule
9 - Thanksgiving
10 - Christmas Day
Please enter the holiday number
Magician Dennis Kyra is scheduled for customer Syam on Thanksgiving
COMMAND MENU
SC - Schedule an appointment with magician
C - Cancel an appointment with magician
S - Status of Magician or Holiday
Q - Quit

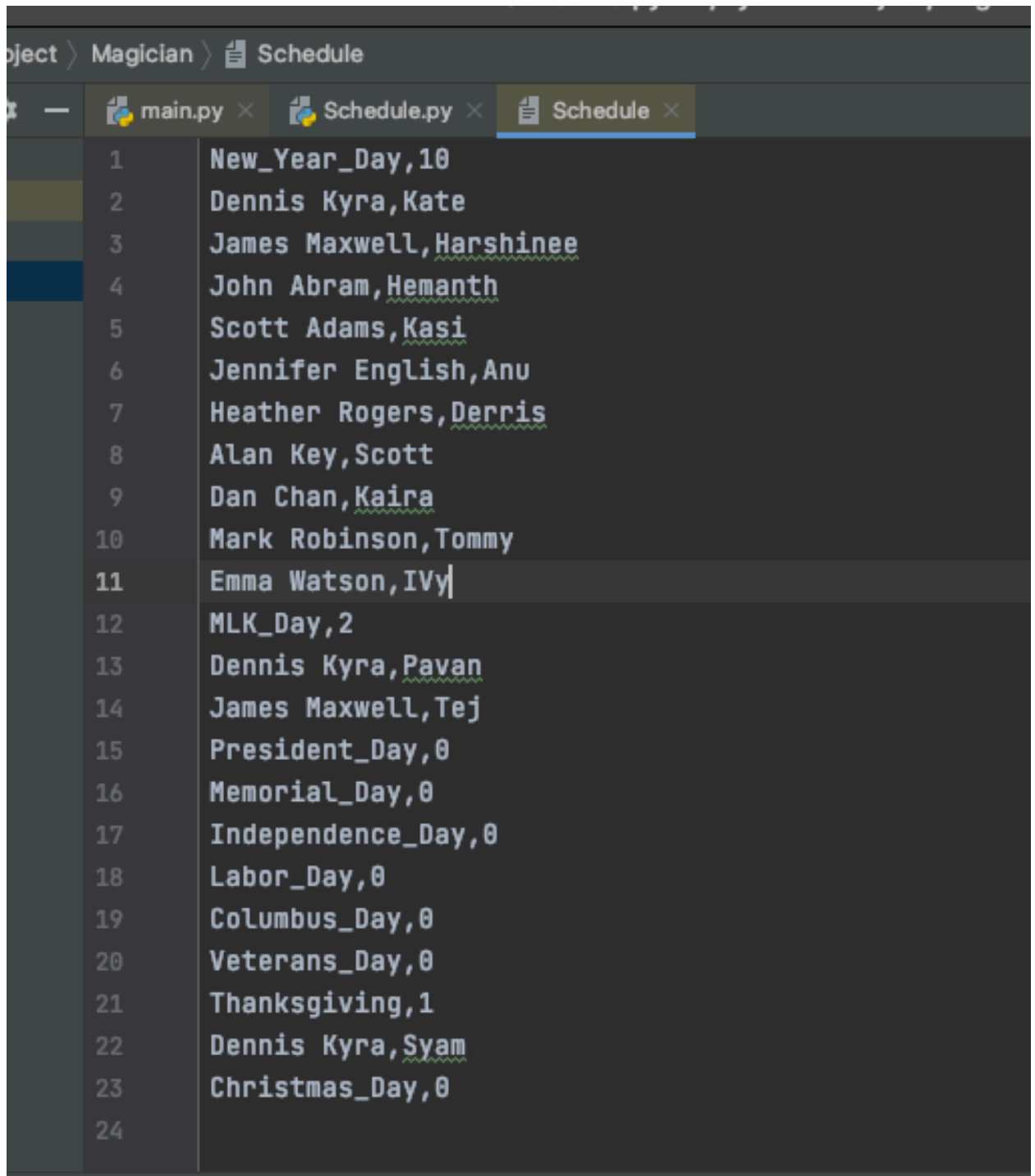
Command: S
Enter name of magician or holiday whose status you want to seeDennis Kyra
Magician's Dennis Kyra schedule is as follows:
A Magician Dennis Kyra is scheduled for Customer Kate
A Magician Dennis Kyra is scheduled for Customer Pavan
A Magician Dennis Kyra is scheduled for Customer Syam

COMMAND MENU
SC - Schedule an appointment with magician
C - Cancel an appointment with magician
S - Status of Magician or Holiday
Q - Quit

Command: Q
Bye!

Process finished with exit code 0
```

8. The Schedule is now written in the Schedule file.



The screenshot shows a code editor with three tabs: 'main.py', 'Schedule.py', and 'Schedule'. The 'Schedule' tab is active, displaying a list of names and dates. The list is as follows:

Line Number	Text
1	New_Year_Day,10
2	Dennis Kyra,Kate
3	James Maxwell,Harshinee
4	John Abram,Hemanth
5	Scott Adams,Kasi
6	Jennifer English,Anu
7	Heather Rogers,Derris
8	Alan Key,Scott
9	Dan Chan,Kaira
10	Mark Robinson,Tommy
11	Emma Watson,IVy
12	MLK_Day,2
13	Dennis Kyra,Pavan
14	James Maxwell,Tej
15	President_Day,0
16	Memorial_Day,0
17	Independence_Day,0
18	Labor_Day,0
19	Columbus_Day,0
20	Veterans_Day,0
21	Thanksgiving,1
22	Dennis Kyra,Syam
23	Christmas_Day,0
24	