Reservation System in Python

Documentation

Author: Michał Deć

Nowy Sącz, 2024

1. Program Scope

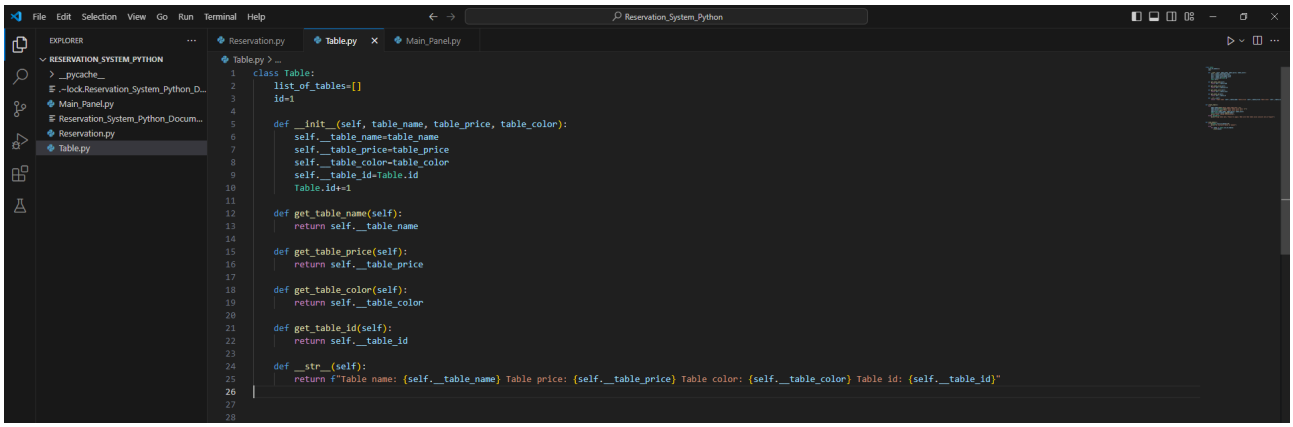
We will code simple application in Python. The goal of application is to map reservation system. We will have three modules. In module named „Main_Panel” we will have control panel to create table, reserve table, view table, and view reservations. In module „Table” there will be class named „Table” to create new table. In module „Reservation” there will be class named „Reservation” to order a table. Written application will also let to control basic exceptions that may rise during program execution. Application has been written with the use of Python 3.12.3 version in Visual Studio Code IDE.

2. Source Code

2.1 Table Module

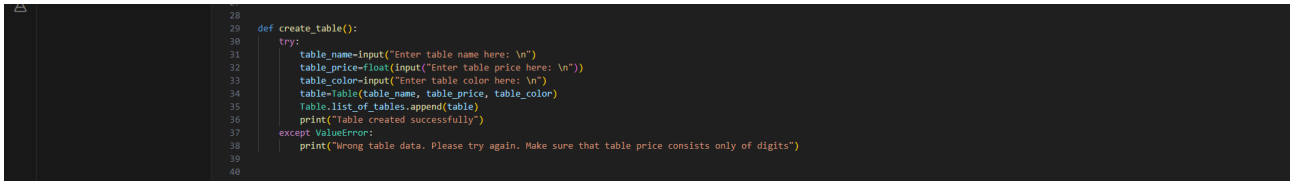
2.1.1 Class Table

We have class „Table” inside „Table” module. This class allows us to create new table. Every table has his name, price, and color. Inside class we have also defined Accessors to return object attribute. What is more, inside class we have defined table id for every table. On the beginning it is set to 1 and with every new created table it is increased by one. Every table is saved to list defined on the class scope.



2.1.2 Function create_table()

Function „create_table()” written inside „Table” module allows for creating new table. We enter table price, name, and color. If data is correct, object constructor from „Table” class is initiated and table object is created, and added to table list stored inside „Table” class.



```
def create_table():
    try:
        table_name=input("Enter table name here: \n")
        table_price=float(input("Enter table price here: \n"))
        table_color=input("Enter table color here: \n")
        table=Table(table_name, table_price, table_color)
        Table.list_of_tables.append(table)
        print("Table created successfully")
    except ValueError:
        print("Wrong table data. Please try again. Make sure that table price consists only of digits")
```

2.1.3 Function view_tables()

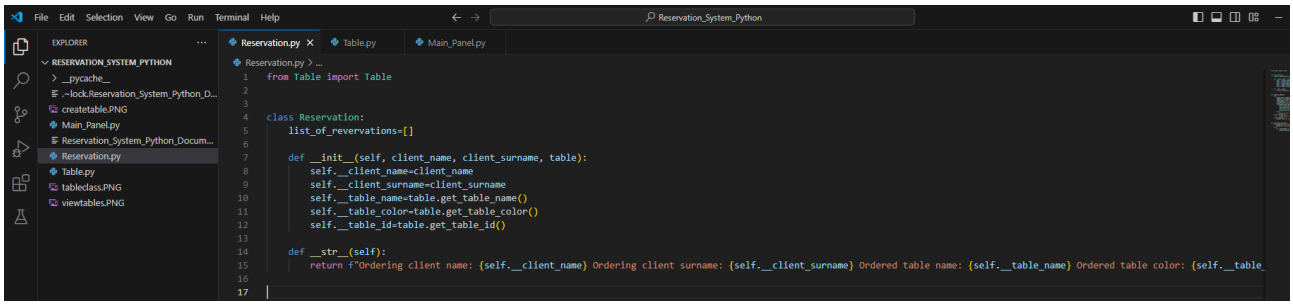
Function „view_tables()” written inside „Table” module allows for listing all available tables from table list stored inside „Table” class. If length of table list is zero, so there are no available tables at the moment, we are notified that there are no tables at the moment. Otherwise, this function lists all table objects from the list.

```
def view_tables():
    if len(Table.list_of_tables)==0:
        print("No available tables at moment")
    else:
        for table in Table.list_of_tables:
            print(table)
```

2.2 Reservation Module

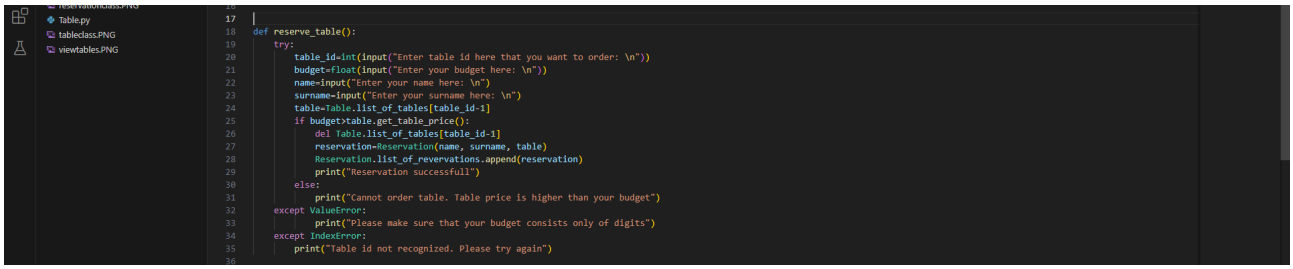
2.2.1 Class Reservation

Inside „Reservation” module we have defined „Reservation” class. This class allows for creating new objects as reservations. What is more, we store reservation list at the class scope. Inside this class we also override string method to print all object’s attributes.



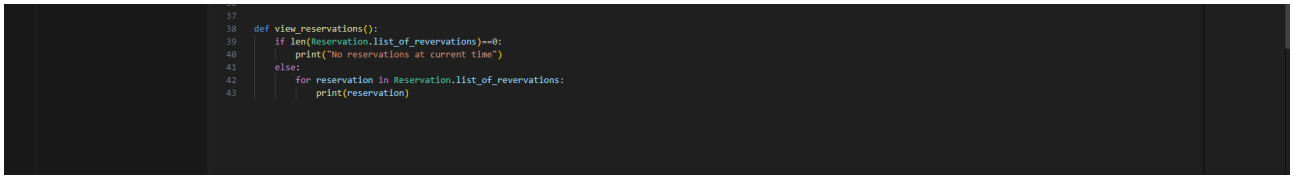
2.2.2 Function reserve_table()

Function „reserve_table()” inside „Reservation” module allows us to reserve a table. We reserve table with the use of table id. Every table has his id, which is auto incremented every time we create table object. We reserve table by his id. To do it we appeal to table list index. For example table one is stored at position 0 inside list. To reserve this table we use [id-1] beacuse table one has id 1, so we need to get access to position 0. Moreover, to reserve a table client’s budget has to be higher than table price. If reservation is successfull, table is deleted from table list and transferred to reservation list.



2.2.3 Function view_reservations()

Function „view_reservations()” written inside „Reservation” module allows us to view all reserved tables. If length of reveration list is 0, it means that there are no reservations at the time. Otherwise, it lists all reservations from reversation list stored inside „Reservation” class.

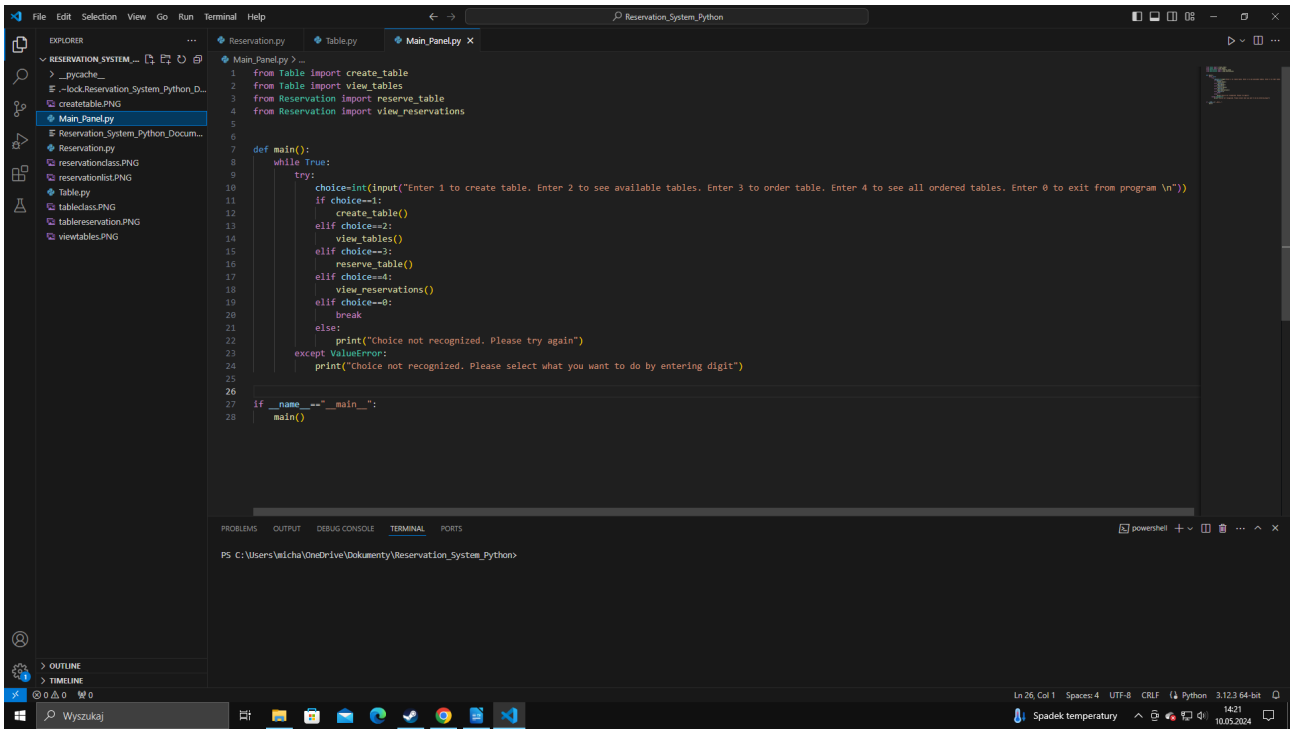


```
def view_reservations():
    if len(Reservation.list_of_revervations)==0:
        print("No reservations at current time")
    else:
        for reservation in Reservation.list_of_revervations:
            print(reservation)
```

2.3 Main_Panel module

2.3.1 Function main()

Function „main()” written inside „Main_Panel” module maps control panel. It allows us to either create new table, view all available tables, create new reservation, and view all reservations. We achieve that by implementing basic if and elif conditions.



3. Resources used for program implementation

1) Python: Python version 3.12.3 - https://www.python.org/downloads/
2) IDE: Visual Studio Code - https://code.visualstudio.com/

List of contents