**Java Project**

**Concept**: Restaurant Reservation System.

**Author**: Dylan McCarthy

**Description:** This project is used for a restaurant to add/amend/remove tables/reservations/customers and for customers to pay the bill on reservation.

**Date**: 5/12/2020

**Constructor files:** Tables.java ,Reservations.java , Customers.java.

**Problem: I couldn’t create a javadoc through my intellij**

**File Description:**

* **AddTableGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Tables.java.  
     This class asks the user to input tableNo, tableSize and table status and all details are validated then the system clarifies to the user are they sure about it with a confirmation message. Then the System adds the details in the allTables ArrayList.
* **AmendTablesGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Tables.java  
     This class asks the user to input tableNo ,tableSize ,status and all details are validated then the system clarifies to the user are to enter the tableNo twice to confirm the amendments with a confirmation message.Then the System  
     chooses the details they want to update details within a switch statement and saves the new details in the allTables ArrayList
* **RemoveTableGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Tables.java. This class asks the user to input TableNo and the system receives all the detail from that specific tableNo then the system clarifies to the user are they sure about removing the object with a confirmation message. Then the System removes the details from the allTables ArrayList.
* **ViewTableGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class tables.java  
     The system loads up a button that views all the objects in allTables arraylist and the user presses the button and Then the System displays all the details from the allTables ArrayList on a JOptionPane Dialog box.
* **AddResGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Reservations.java  
     This class asks the user to input ResID,tableNo,custID,people,date,status, bill amount and all details are validated then the system clarifies to the user are they sure about it with a confirmation message. Then the System  
    adds the details in the allRes ArrayList.
* **AmendResGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Reservations.java  
     This class asks the user to input resID, tableNo, CustID,resStatus,people,date and billAmount and all details are validated then the system clarifies to the user are to enter the resID twice to confirm the amendments with a confirmation message. Then the System chooses the details they want to update details within a switch statement and saves the new details in the allRes ArrayList.
* **PBResGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Reservations.java. This class asks the user what resID would like to pay the bill and it then the system clarifies to the user are they sure about it. Then the System updates the specific resID status to 'PB'.
* **RemoveResGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Reservations.java  
     This class asks the user to input resID and the system receives all the details from that specific resID then the system clarifies to the user are they sure about removing the object with a confirmation message. Then the System removes the details from the allRes ArrayList.
* **ViewResGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Reservations.java  
     The system loads up a button that views all the objects in allRes arraylist and the user presses the button and Then the System displays all the details from the allRes ArrayList on a JOptionPane Dialog box.
* **AddCustomersGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Customers.java. This class asks the user to input custID, name and phone number and all details are validated then the system clarifies to the user are they sure about it with a confirmation message. Then the System adds the details in the allCus ArrayList.
* **AmendCustomersGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Customers.java  
     This class asks the user to input custID, name and phone number and all details are validated then the system clarifies to the user are they sure about it twice with a confirmation message. Then the System chooses the details they want to update details within a switch statement and saves the new details in the allCus ArrayList.
* **RemoveCustomersGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Customers.java  
     This class asks the user to input custID and the system receives all the details from that specific custID then the system clarifies to the user are they sure about removing the object with a confirmation message. Then the System removes the details from the allCus ArrayList.
* **ViewCustomersGUI.java**
  + This java class is accessed through the MainMenu.java. It receives its objects constructors through its constructor class Customers.java  
     The system loads up a button that views all the objects in customers and the user presses the button and Then the System displays all the details from the allCus ArrayList on a JOptionPane Dialog box.

