CS3343 Software Design
Bug Report
Group 12

Name	Student ID	Title
CHENG Yin		Project Manager
SONG Tao		Assistant Project Manager
LUO Peiyuan		Programmer
ZHOU Junchen		Programmer
DU Wenxi		Tester
FENG Yong		Tester

Table of contents

1 Introduction

When the program is abnormal while testing, the testers will log the information in the Dropbox Paper, and after all, all the bug fixes and issues raised will be included in this Bug Report.

2 Bug Repository

Dropbox Paper is the platform we used for logging bugs or abnormalities found.

In the Dropbox Paper, we created a paper named Bug Fix Record. Inside the paper, we created a table with categories highlighted. All the important items are marked inside the Bug Fix Record, and after the bug fixing period, the finished items will be summarized in the Bug Report.

Bug Fix Record

Assigned Track Number	Report Date	Module	Title	Description + Steps to reproduce	Severity Level	Assigned to	Issue Staus	Comments from Developer
C1	14 Nov	Customer	☑ Wrong Numbering at printing official orders	After confirming the order, the official orders printed out has a constant numbering. It should be incremented by 1 every time. Official Orders: [1] Turkey [1] Filet-O-Fish [1] Beef [1] Pork [1] Beef-Egg-Burger Steps to reproduce bug: 1. Login 2. Choose Dine-in 3. Number of people: 11 4. Choose Walk in 5. Choose Restaurant 1 (PepperLunch) 6. Order dish: 1,2,3,1 7. Confirm order: true	Critical	CHENG Yin	Resolved	Problem due to the hashmap used does not accept duplicated key and value. An alternative hashmap from Google Guava called Multimap is used to replace Hashmap.
G1	13 Nov	Reservation, Admin	☑ Table Sorting should be sorted by capacity then table ID.	Table for tomorrow reservation and available time slots: 2-Seats Table with ID of 1 is available tmr for the timeslots: 00:00-23:59 2-Seats Table with ID of 2 is available tmr for the timeslots: 00:00-23:59 ... Currently, If the admin adds new tables in random orders with random capacities, the tables are not sorted. Tables should be sorted by capacity and then table ID. Steps to reproduce 1. log in to the Admin account 2. Add new tables randomly	Minor	LUO Peiyuan	Resolved	Revised algorithm.

3 Guidelines for Bug Reporting

There are important fields that testers must include in the Bug Fix Record; they are categories, titles, descriptions, steps to reproduce, and severity levels.

After the programmers receive the bugs raised in the Bug Fix Record, they will check if it is their responsibility for the classes. If they are responsible for that operation, they will write their names in the "Assigned" column and start working on it.

After checking and working on the problems raised by the testers, the developers will jot down the issue's status. And when finished, the developers should comment on what is leading to the bug and how they fix it to aid the bug fix if the problem happens again in the future.

Important Components of a bug report to be inputted by the tester:
- Title
- Report Date
- Module
- Description
- Steps to reproduce
- Severity Level

Categories of Issues:

Category	Description
General	Bugs or abnormalities that do not belong to any of the admin-related, merchant-related, or customer-related issues.
Admin-related	Bugs or abnormalities that belong to the admin module.
Merchant-related	Bugs or abnormalities that belong to the merchant module.
Customer-related	Bugs or abnormalities that belong to the customer module.

Severity Levels:

Priority Level	Description
Minor	Typos etc., that do not affect the system
Moderate	Wrong output, will not interrupt the system running
Major	Works, but it is not the favorable behavior of the system
Critical	Unable to work and will interrupt system running

Important Components of a bug report to be inputted by the developer:
- Assigned Developer
- Comment from Developer
- Issue Status

Issue Status	Description
Open	After the bug has been raised from testers.
In Progress	After the bug has been assigned to a developer.
Resolved	After the developer has finished fixing the bug.
Closed	After checking the fixed bug by the tester.
Reopened	If the same problem remains from the resolved issue.

4 Bug Reports

4.1 General

Issue #G1

Title:	Table Sorting should be sorted by capacity, then table ID.
Reported Date:	1 November 2022
Severity Level	**Moderate**
Module:	Admin, Reservation
Assigned To:	**LUO Peiyuan**
Status:	Resolved

Problem Description:
| ```
Table for tomorrow reservation and available time slots:
2-Seats Table with ID of 1 is available tmr for the timeslots: 00:00-23:59
2-Seats Table with ID of 2 is available tmr for the timeslots: 00:00-23:59
2-Seats Table with ID of 3 is available tmr for the timeslots: 00:00-23:59
2-Seats Table with ID of 4 is available tmr for the timeslots: 00:00-23:59
2-Seats Table with ID of 5 is available tmr for the timeslots: 00:00-23:59
4-Seats Table with ID of 6 is available tmr for the timeslots: 00:00-23:59
4-Seats Table with ID of 7 is available tmr for the timeslots: 00:00-23:59
4-Seats Table with ID of 8 is available tmr for the timeslots: 00:00-23:59
8-Seats Table with ID of 9 is available tmr for the timeslots: 00:00-23:59
8-Seats Table with ID of 10 is available tmr for the timeslots: 00:00-23:59
2-Seats Table with ID of 11 is available tmr for the timeslots: 00:00-23:59
8-Seats Table with ID of 12 is available tmr for the timeslots: 00:00-23:59
4-Seats Table with ID of 13 is available tmr for the timeslots: 00:00-23:59
``` |
| Currently, If the admin adds new tables in random orders with random capacities, the tables are not sorted. Tables should be sorted by capacity and then table ID. |

Steps to reproduce the bug:
1. log in to the Admin account
2. Add new tables randomly
3. Logout
4. Log in to the Customer account
5. Choose reservation

Comment By Developer:
Revised with collections sorting, changed from Collections.sort(copyOfAvailableTables); **to** Collections.sort(all, Collections.reverseOrder());

Issue #G2

Title:	Table ID does not exist but is still being printed
Reported Date:	7 November 2022
Severity Level:	**Minor**
Module:	Customer, Table
Assigned To:	**SONG Tao**
Status:	Resolved

Problem Description:

Table ID 343 does not exist, but I get this printed: "Can't delete the table with id of 343. Maybe due to its unavailability, or it doesn't exist."

However, it should not output table id 343 since the table id does not exist.

Steps to reproduce the bug:
1. Log in to the Customer account
2. Choose reservation
3. Input random time slot
4. Input table ID that does not exist

Comment By Developer:
Revised the wording of the output.

Issue #G3

Title:	Need to add confirmation check in registering account
Reported Date:	7 November 2022
Severity Level:	**Major**
Module:	Registration
Assigned To:	**CHENG Yin**
Status:	Resolved

Problem Description:

Require a confirm function in register account
Else random input can create an account.

Current register process:

```
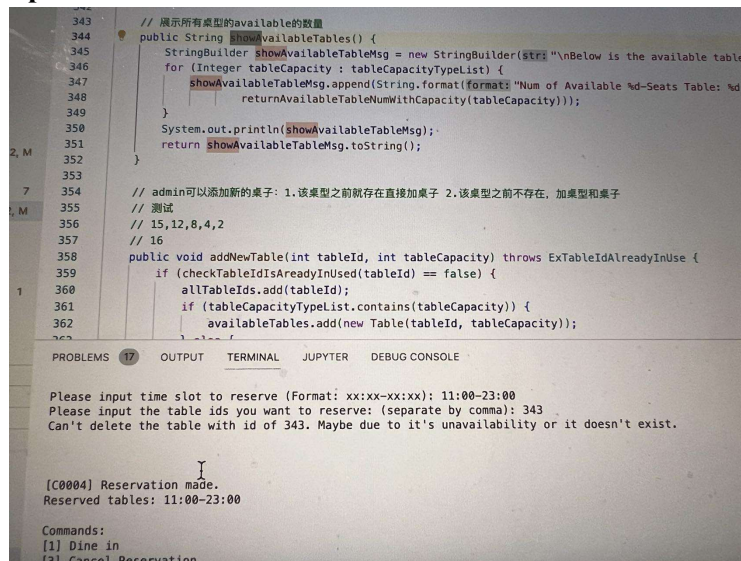List of Active Accounts:
    PepperLunchWorker | M0004
          KFCWorker | M0001
      McDonaldWorker | M0002
        TamJaiWorker | M0003
             yinch33 | C0001
               admin | A0001
               ta123 | C0002
               wedu2 | C0003
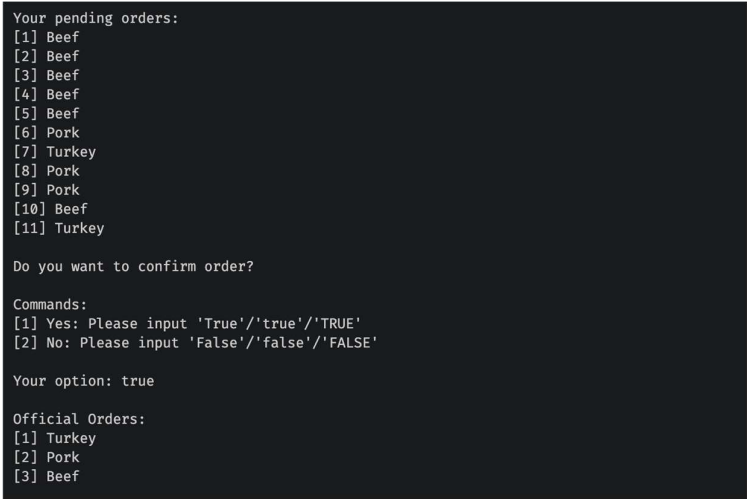

Commands:
[1] Login
[2] Register
[3] Delete Account

Please select your operation: 2

Please choose the type of account to register [1 Customer | 2 Merchant | 3 Cancel]: 1

Please input the username: test

Please input the password: t123

Registration Completed. Please return to login.
```

Comment By Developer:
A *confirmToRegister* function in *CommandAccountManagementRegister* is added when registering a customer or merchant account. Only when the user confirms to register, the respective accounts will be registered.

Issue #G4

Title:	Error! Official Order did not accept duplicate dish
Reported Date:	8 November 2022
Severity Level:	**Critical**
Module:	Customer, Dish
Assigned To:	**CHENG Yin**
Status:	Resolved

Problem Description:
Official Order should accept duplicate dish

```
Your pending orders:
[1] Beef
[2] Beef
[3] Beef
[4] Beef
[5] Beef
[6] Pork
[7] Turkey
[8] Pork
[9] Pork
[10] Beef
[11] Turkey

Do you want to confirm order?

Commands:
[1] Yes: Please input 'True'/'true'/'TRUE'
[2] No: Please input 'False'/'false'/'FALSE'

Your option: true

Official Orders:
[1] Turkey
[2] Pork
[3] Beef
```

Steps to reproduce the bug:
Enter the system as a Customer, use dine in operation and take orders in a restaurant. Then order the same dish more than one time.

Comment By Developer:
Hashmap used is replaced by Multimap to support duplicate keys.

Issue #G5

Title:	Table allocation that does not meet the requirements
Reported Date:	9 November 2022
Severity Level:	**Critical**
Module:	Customer, Table
Assigned To:	**LUO Peiyuan**
Status:	Resolved

Problem Description:

```
Table(s) for tomorrow reservation and available time slots:
4-Seats Table with ID of 0 is available tmr for the timeslots: 00:00-23:59
4-Seats Table with ID of 1 is available tmr for the timeslots: 00:00-23:59

Successfully add table with ID of 3, capacity of 5

Your arranged tables are:
[1] [5-Seats] Tables
[1] [4-Seats] Tables
```

When eight customers are inputted as the number of customers to dine in, the system gives out Two 4-seat tables and One 5-seat table for the table arrangement.

When eight customers come to dining, it is better to allocate two 4-seat tables.

Steps to reproduce the bug:

```
@Test
public void test05() throws ExTableIdAlreadyInUse, ExPeopleNumExceedTotalCapacity {

    tb.addNewTable(3, 5);
    int result = tb.arrangeTableAccordingToNumOfPeople(8).get(0);

    assertEquals(1, result);
}
```

Comment By Developer:
In TableManagement, *arrangeTableAccordingToNumOfPeople* is revised to *getTableArrangementResult.*

Issue #G6

Title:	The table id is not unique
Reported Date:	10 November 2022
Severity Level:	**Major**
Module:	Admin, Table
Assigned To:	**LUO Peiyuan**
Status:	Resolved

Problem Description:

```
<terminated> testing1 [JUnit] G:\eclipse\64bit\202203R\plugins\org.eclipse.justj.o
Successfully add table with ID of 0, capacity of 4
Successfully add table with ID of 0, capacity of 4

Below is the available tables:
Num of Available 4-Seats Table: 2
```

Same table id being added twice. It should only be added once.

Steps to reproduce the bug:

```
public class testing1 {
    TablesManagement tb = TablesManagement.getInstance();
    @Test
    public void test01() throws ExTableIdAlreadyInUse {

        tb.addNewTable(0, 4);
        tb.addNewTable(0, 4);
        String correct = "\nBelow is the available tables: \n"+ "Num of Available 4-Seats Table: 1 \n";
        String result = tb.showAvailableTables();
        assertEquals(correct, result);
    }
```

Comment By Developer:

Added *checkTableIdIsAreadyInUsed* function in *addNewTable* to prevent the same table ID from being added twice.

Issue #G7

Title:	Reservation Time Slot length does not meet the requirements
Reported Date:	11 November 2022
Severity Level:	**Moderate**
Module:	Customer, Reservation
Assigned To:	**SONG Tao**
Status:	Resolved

Problem Description:
Reservation should have time restriction:

Cannot reserve time slot less than half an hour or more than 2 hours

New Update: need to print the error message: cannot reserve timeslot less than half hour or more than 2 hours

Steps to reproduce the bug:

Enter the system as a Customer and go to the reservation page. Enter a time slot that the range is lesser than 30 mins or more than 2 hours

Comment By Developer:

Added restriction to the time limit for reservation.

Issue #G8

Title:	Table capacity cannot be removed when removing table
Reported Date:	16 November 2022
Severity Level:	**Critical**
Module:	TablesManagement
Assigned To:	**LUO Peiyuan**
Status:	Resolved

Problem Description:

```
// æµ<e *1×s1.æ|Œa-▢æœ¬e«available 2.æ|Œa-▢a,▢available 3.æ|Œa-▢æ ¨æœ¬a¨Iæ¨|æœ
public void removeTable(int tableId) throws ExTableNotExist {
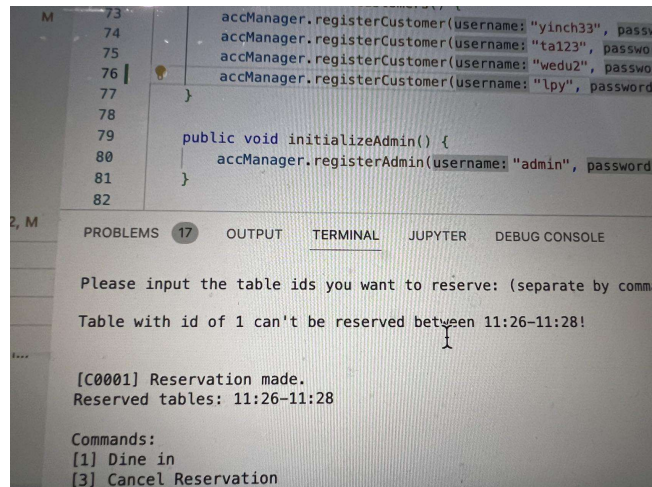
    Table t = returnTableAccordingToTableId(tableId);
    if (t != null) {

        if(allTableIds.contains(tableId)) {
            allTableIds.remove(tableId);
        }
        System.out.printf("kkk");

        availableTables.remove(t);
        System.out.printf("kkk");
        if (returnTableNumWithTableCapacity(t.getTableCapacity()) == 0) {
            removeTableCapacity(t.getTableCapacity());
        }
        System.out.printf("bbb");
        System.out.printf("Successfully delete the table with id of %d \n", tableId);
        return;
    }
    throw new ExTableNotExist(tableId);

}
```

Integer ArrayList can not directly remove the value.

Steps to reproduce the bug:

```
@Test
public void test04() throws ExTableNotExist {


    tb.removeTable(2);
    //this result format is wrong
    String result = "\nTable for tommorrow reservation and available time slots: \n"  +
            "4-Seats Table with ID of 1 is available tmr for the timeslots: 00:00-23:59 \n";

    assertEquals(result, tb.showReservationTable());

}
```

Comment By Developer:
Fixed.

Issue #G9

Title:	**Wrong return value for table reservation**
Reported Date:	19 November 2022
Severity Level:	**Moderate**
Module:	Table Management
Assigned To:	**LUO Peiyuan**
Status:	Resolved

Problem Description:
The function reserveTableAccordingToTimeslot() should return a boolean value when the timeslot has already been booked.

Steps to reproduce the bug:

```java
@Test
public void test12() throws ExTableNotExist, ExTimeSlotAlreadyBeReserved {

    cs3343_project.TimeSlot ts = new cs3343_project.TimeSlot("12:30:00","13:00:00","0");
    boolean result = tb.reserveTableAccordingToTimeslot(0,ts);

    assertEquals(false, result);

}
```

Comment By Developer:
Fixed.

Issue #G10

Title:	Exception handling in Admin setOpenCloseHour
Reported Date:	20 November 2022
Severity Level:	**Moderate**
Module:	Admin, Time management
Assigned To:	**SONG Tao**
Status:	Resolved

Problem Description:
In admin's set opening hours operation, when input the wrong time format to change time, the system outputs the null error and should be handled with exception handling.

Steps to reproduce the bug:

```
[2] Check Customer's Orders
[3] Check Customer's Reservation
[4] Add Restaurant
[5] Remove Restaurant
[6] Add Table
[7] Remove Table
[8] Logout

Please select your operations: 1

Please input new opening hour: 9:00
Please input new closing hour: 6:00
Cannot load from object array because "temp" is null

------------------------------------------------
Commands:
[1] Set Food Court's Opening and Closing Time
[2] Check Customer's Orders
[3] Check Customer's Reservation
[4] Add Restaurant
```

Comment By Developer:
Exception handling is added.

4.2 Customer-related issues
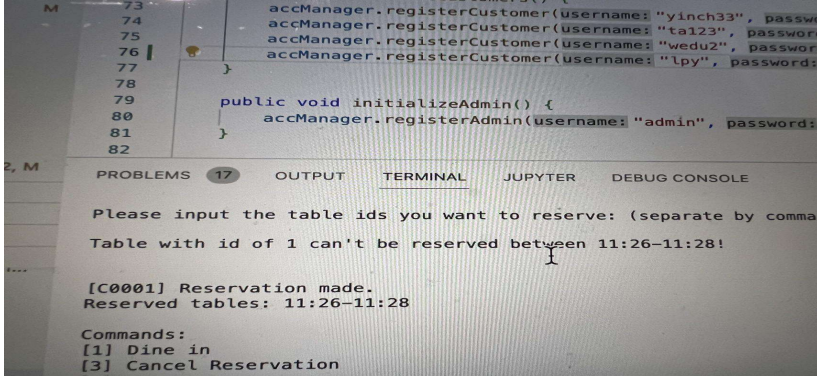
Issue #C1

Title:	Wrong Numbering at printing official orders
Reported Date:	1 November 2022
Severity Level:	**Critical**
Module:	Customer
Assigned To:	**CHENG Yin**
Status:	Resolved

Problem Description:
After confirming the order, the official orders printed out has a constant numbering. It should be incremented by one every time.

```
Official Orders:
[1] Turkey
[1] Filet-O-Fish
[1] Beef
[1] Pork
[1] Beef-Egg-Burger
```

Steps to reproduce the bug:
1. Login
2. Choose Dine-in
3. Number of people: 11
4. Choose Walk in
5. Choose Restaurant 1 (PepperLunch)
6. Order dish: 1,2,3,1
7. Confirm order: true

Comment By Developer:
Corrected the counting inside loop.

Issue #C2

Title:	No option to exit reservation once executed
Reported Date:	8 November 2022
Severity Level:	**Moderate**
Module:	Customer, Reservation
Assigned To:	**CHENG Yin**
Status:	Closed

Problem Description:
There is no option to exit the reservation prompt when I executed it. It keeps prompting me to input the time string if my format is wrong. It should exit when I input "cancel".

Steps to reproduce the bug:
1. Login
2. Choose reservation

Comment By Developer:
Not count into consideration

Issue #C3

Title:	**Wrong output for reservation**
Reported Date:	11 November 2022
Severity Level:	**Major**
Module:	**Customer, Reservation**
Assigned To:	**SONG Tao**
Status:	Resolved

Problem Description:
During every loop in running the customer's module, if there exists reservation for customers, the output should be a reminder, not "Reservation made". Only when the reservation is finished for the customer, it will output "Reservation made".

Cannot reserve between 11:26-11:28, but still, output "reservation made"

Steps to reproduce the bug:
1. Log in as customer
2. Select to reserve
3. Input 11:26-11:28 as time slot
4. Input table id 1

Comment By Developer:
Corrected output phrase set for finishing reservation.

Issue #C4

Title:	Infinite Loop in exiting queue
Reported Date:	11 November 2022
Severity Level:	**Critical**
Module:	Customer, Queue
Assigned To:	**CHENG Yin**
Status:	Resolved

Problem Description:

Infinite loop when trying to select to leave when needing to queue

```
Below is the available tables:
Num of Available 8-Seats Table: 0
Num of Available 4-Seats Table: 0
Num of Available 2-Seats Table: 4

Please input the number of people: 8

Your arranged tables are:
[1] [8-Seats] Tables

For default arrangements, you still need to wait for:
[1] [8-Seats] Table(s)
The Optimized Recommended Arrangements are:
[4] [2-seats]

Commands:
[1] Queue
[2] Walk in with recommended arrangement
[3] Leave

Please choose your operation: 3

Commands:
[1] Queue
[2] Walk in with recommended arrangement
[3] Leave

Please choose your operation: 3

Commands:
[1] Queue
[2] Walk in with recommended arrangement
[3] Leave

Please choose your operation:
```

Steps to reproduce the bug:
1. Login as customer
2. Select to dine in
3. Input number greater than the max capacity of all tables, i.e., 32.
4. Choose to leave when prompted to queue or leave.

Comment By Developer:

Corrected the output phrase set for *recommendedArrangement*.

Issue #C5

Title:	Payment method cannot be shown
Reported Date:	15 November 2022
Severity Level:	**Critical**
Module:	Customer, Payment
Assigned To:	**ZHOU Junchen**
Status:	Resolved

Problem Description:

Payment method name error
When payment is complete, the payment method is a string of gibberish.

Steps to reproduce the bug:

```
List of Merchants of this restaurant:
[1] M0003 | TamJaiWorker
[2] M0004 | TamJaiWorker2
[3] M0005 | TamJaiWorker1

Input staff MId: M0003

Customer Name: yinch33
Customer ID: C0001
Bill no. is: B0001

Ordered Dishes:
[1] Lettuce-Mixian
[2] Pork-Mixian

You have completed payment with Payment@6df97b55. Thank you!

Your bill number is: B0001

Commands:
[1] Alipay
[2] WeChat Pay
[3] Cash
```

Comment By Developer:
Used toString() method.

Issue #C6

Title:	**No need for a reminder for reservation if reservation not made**
Reported Date:	15 November 2022
Severity Level:	**Major**
Module:	Reservation
Assigned To:	**ZHOU Junchen**
Status:	Resolved

Problem Description:

If there is no reservation made, the program should not output the reservation reminder.

Steps to reproduce the bug:

Customer Reservation:

```
Please select your operation: 2

Table(s) for tomorrow reservation and available time slots:
2-Seats Table with ID of 1 is available tmr for the timeslots: 00:00-23:59
2-Seats Table with ID of 2 is available tmr for the timeslots: 00:00-23:59
2-Seats Table with ID of 3 is available tmr for the timeslots: 00:00-23:59
2-Seats Table with ID of 4 is available tmr for the timeslots: 00:00-23:59
2-Seats Table with ID of 5 is available tmr for the timeslots: 00:00-23:59
2-Seats Table with ID of 17 is available tmr for the timeslots: 00:00-23:59
4-Seats Table with ID of 6 is available tmr for the timeslots: 00:00-23:59
4-Seats Table with ID of 7 is available tmr for the timeslots: 00:00-23:59
4-Seats Table with ID of 8 is available tmr for the timeslots: 00:00-23:59
4-Seats Table with ID of 15 is available tmr for the timeslots: 00:00-23:59
8-Seats Table with ID of 9 is available tmr for the timeslots: 00:00-23:59
8-Seats Table with ID of 10 is available tmr for the timeslots: 00:00-23:59
8-Seats Table with ID of 11 is available tmr for the timeslots: 00:00-23:59


Please input time slot to reserve (Format: xx:xx-xx:xx): 00:00-00:15
Please input the table ids you want to reserve: (separate by comma): 1,2,3

Table with id of 1 can't be reserved between 00:00-00:15!

Table with id of 2 can't be reserved between 00:00-00:15!

Table with id of 3 can't be reserved between 00:00-00:15!


[C0001] Error: Reservation not made.


-----------------------------------------------
Reminder: You have a reserved for tomorrow:

Commands:
[1] Dine in
[3] Cancel Reservation
[5] Logout
```

Comment By Developer:

Revised.

4.3 Merchant-related issues

Issue #M1

Title:	Merchant should not check customers' orders that do not belong to his/her restaurant
Reported Date:	2 November 2022
Severity Level:	**Critical**
Module:	Merchant
Assigned To:	**CHENG Yin**
Status:	Resolved

Problem Description:
In the merchant module, when choosing to check customers' orders, it printed a lot of dishes from orders that do not belong to the restaurant owned by the merchant. It should print the dish from orders that belong to the restaurant owned by the merchant only.

Steps to reproduce the bug:

```
Commands:
[1] Login
[2] Register
[3] Delete Account

Please select your operation: 1

Please input the username: KFCWorker
Please input the password: t123

You have succeeded in logging in the system.
UserId: M0001

Commands:
[1] Modify Menu
[2] Check Order
[3] Logout

Please select your operations: 2

Please input customer's id: C0001

Customer Name: yinch33
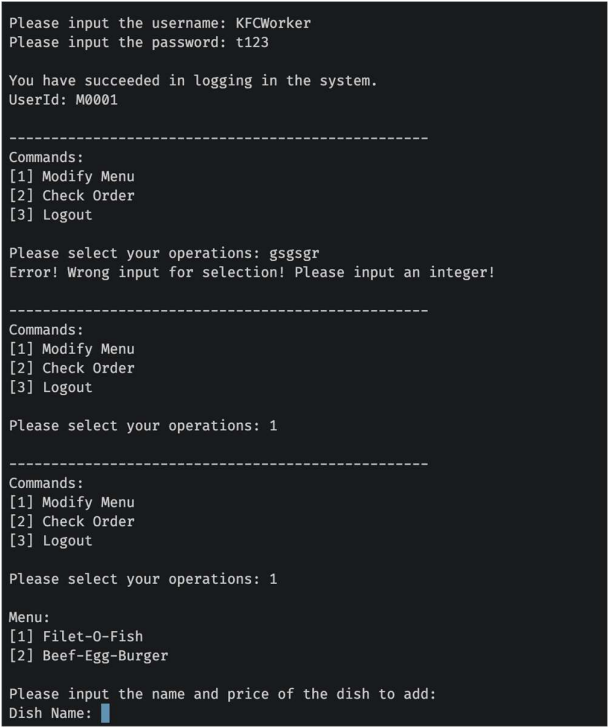Customer ID: C0001
Bill no. is: B0001

Ordered Dishes:
[1] Hot-Chocolate
[2] Latte
[3] Americano
[4] Crunchy-Ovaltine
[5] Hot-Chocolate

Commands:
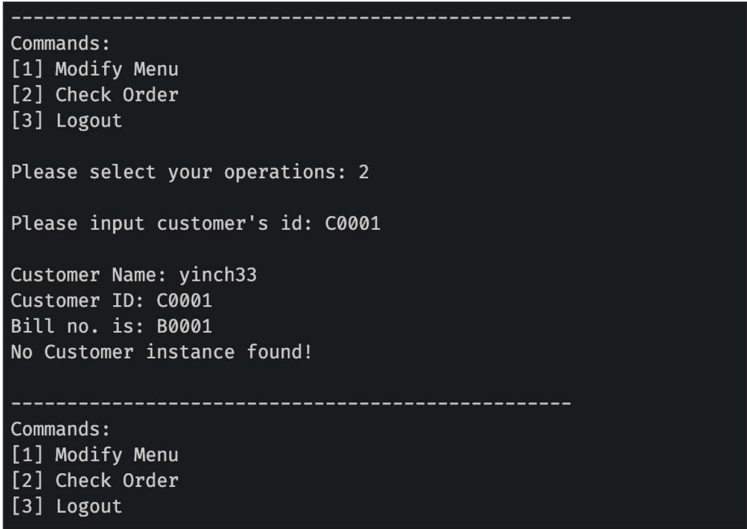[1] Modify Menu
[2] Check Order
[3] Logout
```

Comment By Developer:
A multimap is added to store the dish corresponding to the restaurant. A function *customerOrdersAccordingToRestaurant* is used to return the corresponding dish of the merchant.

Issue #M2
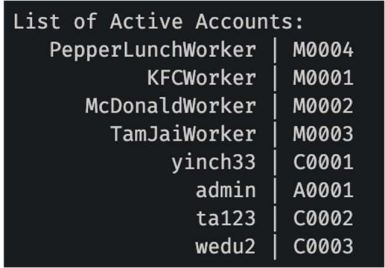
Title:	**Wrong output for selection while choosing to modify menu.**
Reported Date:	4 November 2022
Severity Level:	**Major**
Module:	**Merchant**
Assigned To:	**CHENG Yin**
Status:	Resolved

Problem Description:
When going into the merchant's modify menu, should output choice to add dish, remove dish, or edit dish, instead of outputting modifying menu again.

```
Please input the username: KFCWorker
Please input the password: t123

You have succeeded in logging in the system.
UserId: M0001

------------------------------------------------
Commands:
[1] Modify Menu
[2] Check Order
[3] Logout

Please select your operations: gsgsgr
Error! Wrong input for selection! Please input an integer!

------------------------------------------------
Commands:
[1] Modify Menu
[2] Check Order
[3] Logout

Please select your operations: 1

------------------------------------------------
Commands:
[1] Modify Menu
[2] Check Order
[3] Logout

Please select your operations: 1

Menu:
[1] Filet-O-Fish
[2] Beef-Egg-Burger

Please input the name and price of the dish to add:
Dish Name: ▮
```

Steps to reproduce the bug:
1. Login as Merchant
2. Choose Modify Menu

Comment By Developer:
Corrected the output phrase set for modifying the menu.

Issue #M3

Title:	**Wrong Output for checking orders**
Reported Date:	7 November 2022
Severity Level:	**Minor**
Module:	Merchant
Assigned To:	**CHENG Yin**
Status:	Resolved

Problem Description:

Should be "This customer has no orders" not "No customer instance found"

```
-------------------------------------------------
Commands:
[1] Modify Menu
[2] Check Order
[3] Logout

Please select your operations: 2

Please input customer's id: C0001

Customer Name: yinch33
Customer ID: C0001
Bill no. is: B0001
No Customer instance found!

-------------------------------------------------
Commands:
[1] Modify Menu
[2] Check Order
[3] Logout
```

Steps to reproduce the bug:
1. Log in to the merchant's account
2. Choose the operation [2] to check customers' orders
3. Input the customer ID of whom do not have orders made
4. "No customer instance found" wordings are printed even when the customer id is printed out

Comment By Developer:
Wrong exception handling, revised to print out no orders found.

4.4 Admin-related Issues

Issue #A1

Title:	Account List printed is not sorted
Reported Date:	28 October 2022
Severity Level:	**Minor**
Module:	Admin
Assigned To:	**CHENG Yin**
Status:	Resolved

Problem Description:
As shown in the below screenshot, the active account list retrieved from admin module is not sorted by user ID. And the list's formatting is a bit chaotic. Please align to the left with the numbering, user ID, and username.

```
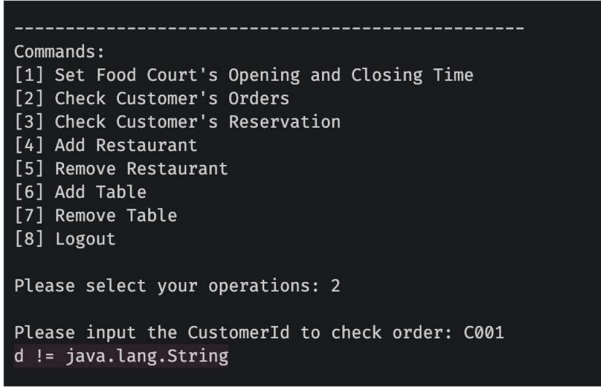List of Active Accounts:
 PepperLunchWorker | M0004
        KFCWorker | M0001
   McDonaldWorker | M0002
      TamJaiWorker | M0003
          yinch33 | C0001
            admin | A0001
            ta123 | C0002
            wedu2 | C0003
```

Steps to reproduce the bug:
At the start of the program.

Comment By Developer:
A sorting algorithm in *AccountManagement* is added.
The sorting algorithm will first sort the prefix of the user ID in numerical order; then it will sort the id based on the last four digits.

Issue #A2

Title:	Wrong output in admin module when checking customers' orders
Reported Date:	6 November 2022
Severity Level:	**Minor**
Module:	Admin
Assigned To:	**CHENG Yin**
Status:	Resolved

Problem Description:
When running admin's operation to check customers' orders, there is the "Official orders" wording printed out even if there are no customers' orders returned.

It should not print "official orders" when there are no customers' orders.

Steps to reproduce the bug:
1. Log in to the admin's account
2. Choose the operation to check customers' orders
3. Input the customer ID of whom do not have orders made
4. Only "Official Orders" wordings are printed

Comment By Developer:
Revised the output when there are no orders made. The revised version will print out "No orders made".

Issue #A3

Title:	**Wrong output for Exception Handling in Checking order**
Reported Date:	13 November 2022
Severity Level:	**Minor**
Module:	Admin, Customer
Assigned To:	**CHENG Yin**
Status:	Resolved

Problem Description:

```
-------------------------------------------------
Commands:
[1] Set Food Court's Opening and Closing Time
[2] Check Customer's Orders
[3] Check Customer's Reservation
[4] Add Restaurant
[5] Remove Restaurant
[6] Add Table
[7] Remove Table
[8] Logout

Please select your operations: 2

Please input the CustomerId to check order: C001
d != java.lang.String
```

Should catch customer ID not found, not "d != java.lang.String"

Steps to reproduce the bug:
1. Login as Admin
2. Choose to check customers' orders
3. Input customer id that do not exist.

Comment By Developer:
Added Exception handling to correct output if customer instance not found.

End of Report