


Faculty of Information Technology									
<p>I declare that I am familiar with, and will abide to the Examination rules of CTU</p>             <p>_____</p> <p><b>Signature</b></p>	<p><b>SUBJECT NAME: Programming with Python</b></p> <p><b>SUBJECT CODE: PRP 411</b></p>								
	<p><b>Formative Assessment 1</b></p> <p><b>Duration:</b></p> <p><b>Date:</b> 22-Jun-2023</p> <p><b>Total Marks:</b></p> <p><b>Total pages:</b></p>					<p><b>Examiner:</b> Mr. Junior Manganyi</p> <p><b>Moderator:</b></p>			
	<p><b>Student number</b></p>								
	2	0	2	3	1		2	9	7
	<p><b>Surname:</b> Modise</p>					<p><b>Initials:</b> RGW</p>			/

## Contents

Code: .....	3
Output: .....	8

Code:

```
class Sale:
    def __init__(self):
        self.total_sale = 0.0

class Table:
    def __init__(self, table_num):
        self.table_num = table_num
        self.server = None
        self.customers = 0
        self.orders = []

    def assign_server(self, server):
        self.server = server

    def add_customers(self, count):
        self.customers = count

    def add_order(self, order):
        self.orders.append(order)

    def prepare_bill(self):
        total = sum(order.price * order.quantity for order in self.orders)
        bill = f"Table: {self.table_num}\n"
        bill += "-----\n"
        for order in self.orders:
            bill += f"{order.name} x {order.quantity}: R{order.price *
order.quantity}\n"
        bill += "-----\n"
        bill += f"Total: R{total}\n"
        return bill, total

    def clear_table(self):
        self.server = None
        self.customers = 0
        self.orders = []

class Order:
    def __init__(self, name, price, quantity):
        self.name = name
        self.price = price
        self.quantity = quantity

def read_login_cred(file_name):
    credentials = {}
```

```

with open(file_name, 'r') as file:
    for line in file:
        username, passwd = line.strip().split(',')
        credentials[username] = passwd
    return credentials

def read_menu(file_name):
    menu = {}
    with open(file_name, 'r') as file:
        for line in file:
            item_name, item_price = line.strip().split(',')
            menu[item_name] = float(item_price)
    return menu

def display_menu():
    print("Main Menu:")
    print("1. Assign Table")
    print("2. Change Customers")
    print("3. Add to Order")
    print("4. Prepare Bill")
    print("5. Complete Sale")
    print("6. Cash Up")
    print("0. Log Out")

def assign_table(current_server, table_list):
    print("Available tables:")
    available_tables = [table for table in table_list if table.server is None]
    for table in available_tables:
        print(f"Table {table.table_num}")
    table_num = int(input("Enter the table number: "))
    table = next((table for table in table_list if table.table_num ==
table_num), None)
    if table and table.server is None:
        table.assign_server(current_server)
        choice = input("Do you want to add customers to the table? (y/n): ")
        if choice.lower() == "y":
            count = int(input("Enter the number of customers: "))
            table.add_customers(count)
            print("Table assigned successfully!")
        else:
            print("Invalid table number or the table is already assigned.")

def change_customers(current_server, table_list):
    print("Tables assigned to you:")

```

```

    assigned_tables = [table for table in table_list if table.server ==
current_server]
    for table in assigned_tables:
        print(f"Table {table.table_num}")
        table_num = int(input("Enter the table number: "))
        table = next((table for table in table_list if table.table_num ==
table_num and table.server == current_server), None)
        if table:
            count = int(input("Enter the new number of customers: "))
            table.add_customers(count)
            print("Number of customers changed successfully!")
        else:
            print("Invalid table number or the table is not assigned to you.")

def add_to_order(current_server, table_list, menu_items):
    print("Tables assigned to you:")
    assigned_tables = [table for table in table_list if table.server ==
current_server]
    for table in assigned_tables:
        print(f"Table {table.table_num}")
        table_num = int(input("Enter the table number: "))
        table = next((table for table in table_list if table.table_num ==
table_num and table.server == current_server), None)
        if table:
            print("Menu:")
            for item_name, item_price in menu_items.items():
                print(f"{item_name}: R{item_price}")
            item_name = input("Enter the name of the item: ")
            if item_name in menu_items:
                item_price = menu_items[item_name]
                item_quantity = int(input("Enter the quantity: "))
                order = Order(item_name, item_price, item_quantity)
                table.add_order(order)
                print("Order added successfully!")
            else:
                print("Invalid item name.")
        else:
            print("Invalid table number or the table is not assigned to you.")

def prepare_bill(current_server, table_list):
    print("Tables assigned to you:")
    assigned_tables = [table for table in table_list if table.server ==
current_server]
    for table in assigned_tables:
        print(f"Table {table.table_num}")
        table_num = int(input("Enter the table number: "))

```

```

        table = next((table for table in table_list if table.table_num ==
table_num and table.server == current_server), None)
        if table:
            bill, total = table.prepare_bill()
            print("Bill:")
            print(bill)
            file_name = input("Enter the file name to save the bill: ")
            with open(file_name, 'w') as file:
                file.write(bill)
            print("Bill successfully prepared!")
            return total
        else:
            print("Invalid table number or the table is not assigned to you.")
            return 0.0

def complete_sale(current_server, table_list, current_sale):
    print("Tables assigned to you:")
    assigned_tables = [table for table in table_list if table.server ==
current_server]
    for table in assigned_tables:
        print(f"Table {table.table_num}")
        table_num = int(input("Enter the table number: "))
        table = next((table for table in table_list if table.table_num ==
table_num and table.server == current_server), None)
        if table:
            if len(table.orders) > 0:
                total = prepare_bill(current_server, table_list)
                current_sale.total_sale += total
                table.clear_table()
                print("Sale completed successfully!")
            else:
                print("No orders found for the table.")
        else:
            print("Invalid table number or the table is not assigned to you.")

def cash_up(current_sale):
    print("Total sales: R", current_sale.total_sale)
    choice = input("Do you want to clear the daily total? (y/n): ")
    if choice.lower() == "y":
        current_sale.total_sale = 0.0
        print("Daily total cleared.")

def point_of_sale():
    login_file = "Login.txt"
    menu_file = "Menu.txt"
    login_cred = read_login_cred(login_file)

```

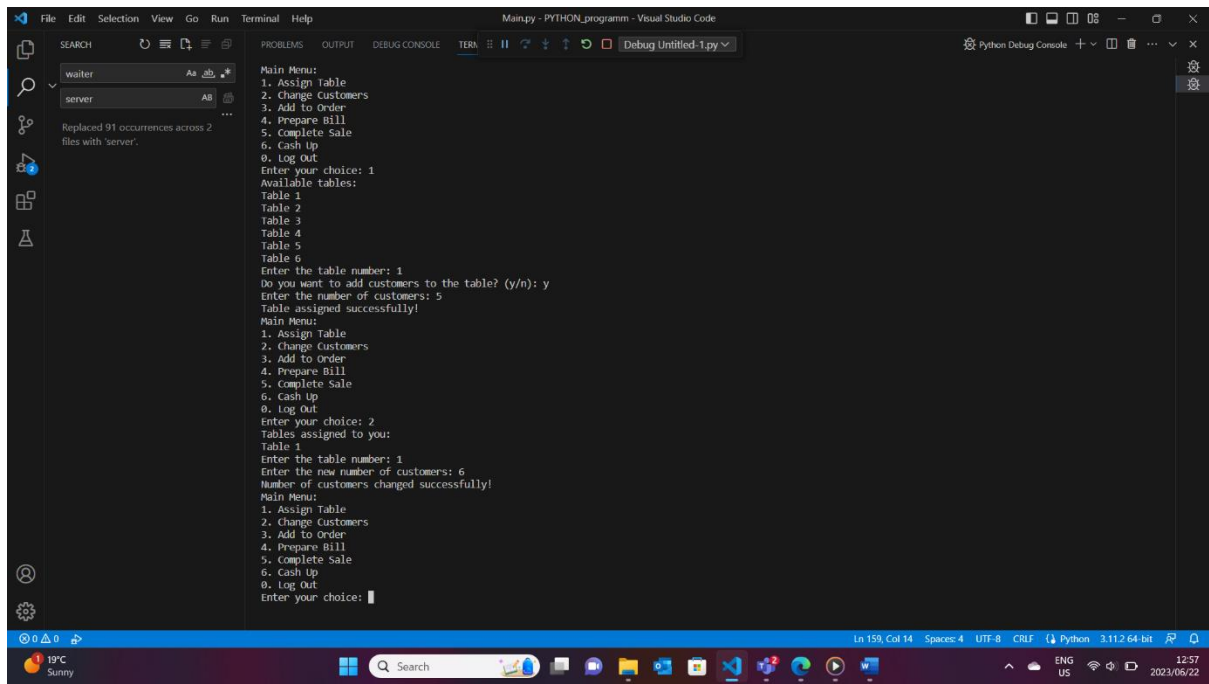
```

menu_items = read_menu(menu_file)
table_list = [Table(i + 1) for i in range(6)]
current_sale = Sale()
current_server = input("Username: ")
passwd = input("passwd: ")
if current_server in login_cred and login_cred[current_server] == passwd:
    print("Login successful!")
    while True:
        display_menu()
        choice = int(input("Enter your choice: "))
        if choice == 1:
            assign_table(current_server, table_list)
        elif choice == 2:
            change_customers(current_server, table_list)
        elif choice == 3:
            add_to_order(current_server, table_list, menu_items)
        elif choice == 4:
            prepare_bill(current_server, table_list)
        elif choice == 5:
            complete_sale(current_server, table_list, current_sale)
        elif choice == 6:
            cash_up(current_sale)
        elif choice == 0:
            print("Logged out successfully!")
            break
        else:
            print("Invalid choice. Please try again.")
    else:
        print("Invalid username or passwd. Login failed.")

point_of_sale()

```

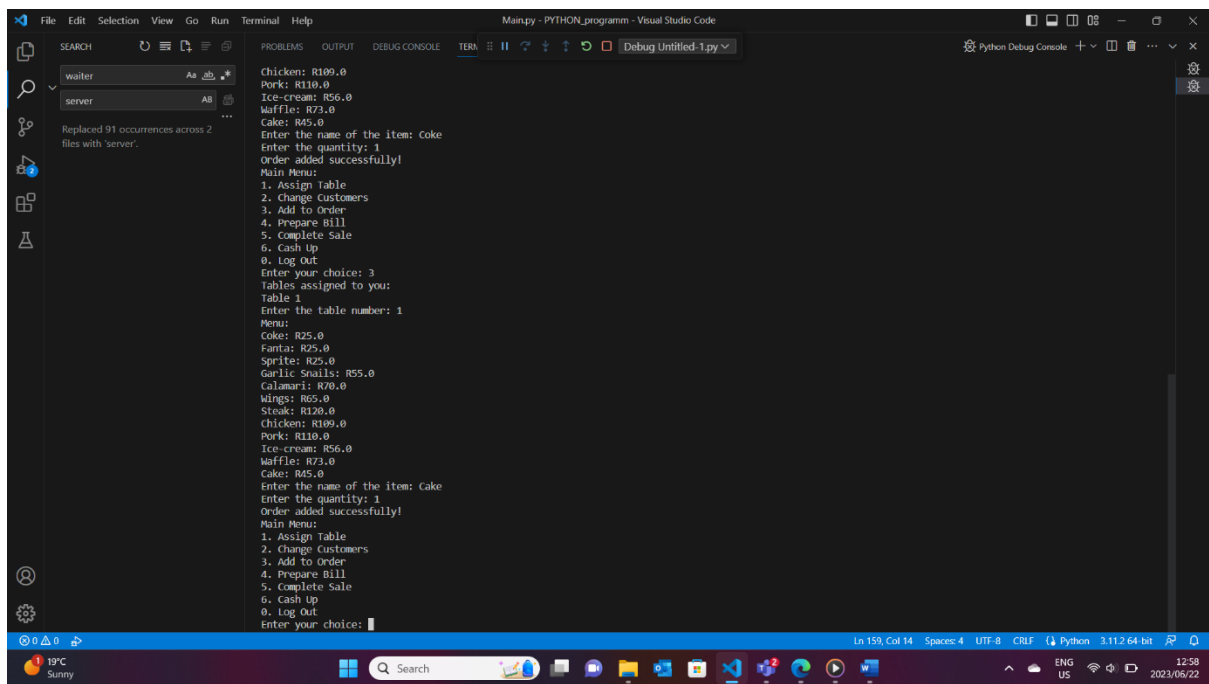
## Output:



The screenshot shows the Visual Studio Code interface with a terminal window titled 'Main.py - PYTHON\_programm - Visual Studio Code'. The terminal output is as follows:

```
Main Menu:
1. Assign Table
2. Change Customers
3. Add to Order
4. Prepare Bill
5. Complete Sale
6. Cash Up
0. Log Out
Enter your choice: 1
Available tables:
Table 1
Table 2
Table 3
Table 4
Table 5
Table 6
Enter the table number: 1
Do you want to add customers to the table? (y/n): y
Enter the number of customers: 5
Table assigned successfully!
Main Menu:
1. Assign Table
2. Change Customers
3. Add to Order
4. Prepare Bill
5. Complete Sale
6. Cash Up
0. Log Out
Enter your choice: 2
Tables assigned to you:
Table 1
Enter the table number: 1
Enter the new number of customers: 6
Number of customers changed successfully!
Main Menu:
1. Assign Table
2. Change Customers
3. Add to Order
4. Prepare Bill
5. Complete Sale
6. Cash Up
0. Log Out
Enter your choice: |
```

The status bar at the bottom indicates 'Ln 159, Col 14', 'Spaces: 4', 'UTF-8', 'CRLF', 'Python', '3.11.2 64 bit', and the system clock shows '12:57' on '2023/06/22'.



The screenshot shows the Visual Studio Code interface with a terminal window titled 'Main.py - PYTHON\_programm - Visual Studio Code'. The terminal output continues from the previous screenshot:

```
Chicken: R109.0
Pork: R110.0
Ice-cream: R56.0
Waffle: R73.0
Cake: R45.0
Enter the name of the item: Coke
Enter the quantity: 1
Order added successfully!
Main Menu:
1. Assign Table
2. Change Customers
3. Add to Order
4. Prepare Bill
5. Complete Sale
6. Cash Up
0. Log Out
Enter your choice: 3
Tables assigned to you:
Table 1
Enter the table number: 1
Menu:
Coke: R25.0
Fanta: R25.0
Sprite: R25.0
Garlic Snails: R55.0
Calamari: R70.0
Wings: R65.0
Steak: R120.0
Chicken: R109.0
Pork: R110.0
Ice-cream: R56.0
Waffle: R73.0
Cake: R45.0
Enter the name of the item: Cake
Enter the quantity: 1
Order added successfully!
Main Menu:
1. Assign Table
2. Change Customers
3. Add to Order
4. Prepare Bill
5. Complete Sale
6. Cash Up
0. Log Out
Enter your choice: |
```

The status bar at the bottom indicates 'Ln 159, Col 14', 'Spaces: 4', 'UTF-8', 'CRLF', 'Python', '3.11.2 64 bit', and the system clock shows '12:58' on '2023/06/22'.



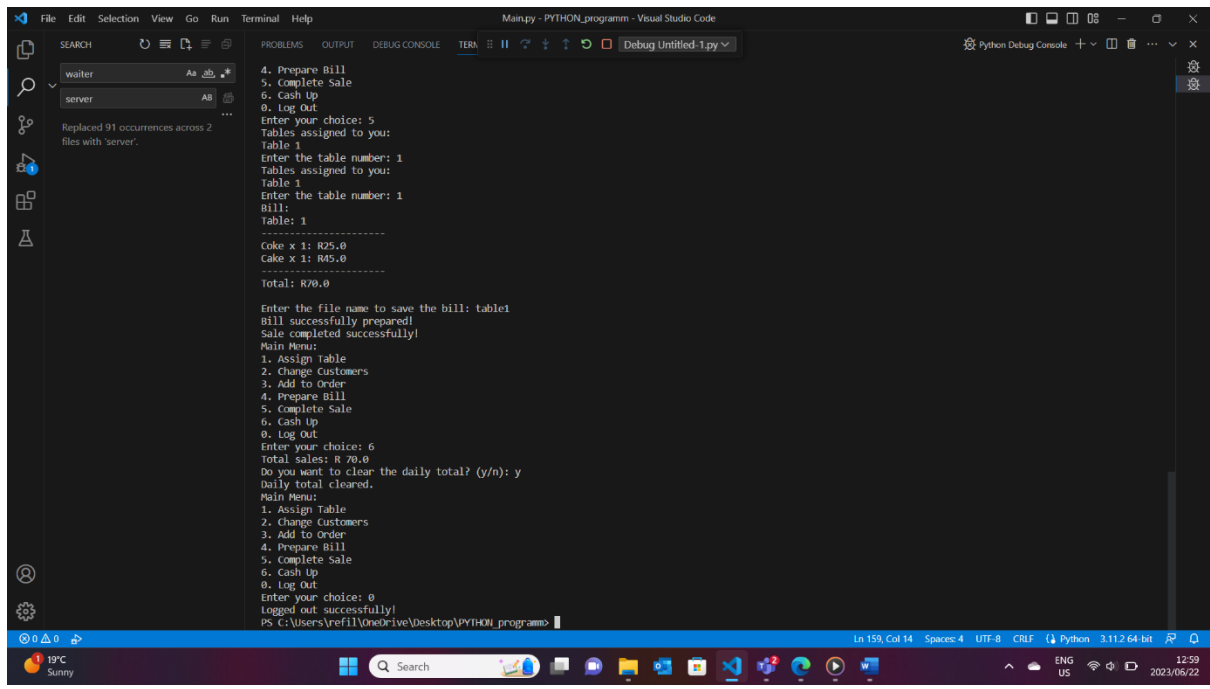
```
File Edit Selection View Go Run Terminal Help
Main.py - PYTHON_programm - Visual Studio Code
SEARCH
waiter
server
Replaced 91 occurrences across 2 files with 'server'.
PROBLEMS OUTPUT DEBUG CONSOLE TERN
Debug Untitled-1.py
Fantar: R25.0
Sprite: R25.0
Garlic Snails: R55.0
Calamari: R70.0
Wings: R65.0
Steak: R120.0
Chicken: R100.0
Pork: R110.0
Ice-cream: R50.0
Mafies: R75.0
Cake: R45.0
Enter the name of the item: Cake
Enter the quantity: 1
Order added successfully!
Main Menu:
1. Assign Table
2. Change Customers
3. Add to Order
4. Prepare Bill
5. Complete Sale
6. Cash Up
0. Log Out
Enter your choice: 4
Tables assigned to you:
Table 1
Enter the table number: 1
Bill:
Table: 1
-----
Coke x 1: R25.0
Cake x 1: R45.0
-----
Total: R70.0

Enter the file name to save the bill: table1
Bill successfully prepared!
Main Menu:
1. Assign Table
2. Change Customers
3. Add to Order
4. Prepare Bill
5. Complete Sale
6. Cash Up
0. Log Out
Enter your choice: 1
Ln 159, Col 14 Spaces: 4 UTF-8 CRLF Python 3.11.2 64-bit
19°C Sunny 12:58 2023/06/22
```

```
File Edit Selection View Go Run Terminal Help
Main.py - PYTHON_programm - Visual Studio Code
SEARCH
waiter
server
Replaced 91 occurrences across 2 files with 'server'.
PROBLEMS OUTPUT DEBUG CONSOLE TERN
Debug Untitled-1.py
-----
Coke x 1: R25.0
Cake x 1: R45.0
-----
Total: R70.0

Enter the file name to save the bill: table1
Bill successfully prepared!
Main Menu:
1. Assign Table
2. Change Customers
3. Add to Order
4. Prepare Bill
5. Complete Sale
6. Cash Up
0. Log Out
Enter your choice: 5
Tables assigned to you:
Table 1
Enter the table number: 1
Tables assigned to you:
Table 1
Enter the table number: 1
Bill:
Table: 1
-----
Coke x 1: R25.0
Cake x 1: R45.0
-----
Total: R70.0

Enter the file name to save the bill: table1
Bill successfully prepared!
Sale completed successfully!
Main Menu:
1. Assign Table
2. Change Customers
3. Add to Order
4. Prepare Bill
5. Complete Sale
6. Cash Up
0. Log Out
Enter your choice: 6
Total sales: R 70.0
do you want to clear the daily total? (y/n):
Ln 159, Col 14 Spaces: 4 UTF-8 CRLF Python 3.11.2 64-bit
19°C Sunny 12:59 2023/06/22
```



## Completed Declaration of Authenticity

I, Refilwe Modise solemnly declare that the work presented in this Summative is entirely my own. I have not plagiarized or copied the work of others without proper acknowledgment. I affirm that the content, ideas, and arguments presented herein are the result of my independent effort and intellectual contributions.

I understand the significance of academic integrity and the detrimental consequences of engaging in plagiarism or other forms of dishonesty. Therefore, I assure you the following:

0. All sources used in this work, including but not limited to books, articles, websites, and personal communications, have been appropriately cited and referenced according to the specified guidelines or referencing style.
1. Any direct quotations or paraphrased information from external sources have been identified by using quotation marks or proper citation methods.
2. I have not received any unauthorized assistance or collaboration from others in completing this work, except for instances explicitly permitted by the instructor or clearly stated in the assignment guidelines.
3. The ideas, arguments, and interpretations expressed in this work are my own and have not been submitted for assessment in any other academic setting unless explicitly mentioned and properly acknowledged.
4. I acknowledge that failure to adhere to these principles of academic honesty and integrity may result in severe penalties, including but not limited to the rejection of this work, loss of marks, academic probation, or disciplinary action as deemed appropriate by the educational institution.



Signature:

Date: 22/06/2023

