GRAB-N-Go

FINAL PROJECT

Course: ITMD 510

Anusha Satish A20401921

Sivaranjani Prabasankar A20436206

Abstract

This report gives us a detailed insight on how the project is built. The project will be explained right from the design with the diagrams such as Class diagram and ER diagram. You will know how the CRUD functionalities are executed in the system.

Introduction

Project Description:

A Cake shop where you can place your cake orders online and pick it from the stores located in different parts of the city.

A customer can order cakes from available types and can choose the toppings from the given set. It can be ordered online and need to pick it up from the nearby store which you can choose while ordering. Customer can check the status of their orders as well as all orders from their house for the day and can pick it up when ready.

The system Admin tells when the next order is to be done. He also decides when all the orders are done and all the orders for the day has fulfilled. The Admin can select the number of toppings for the day and setup the shop from the scratch. The Admin can also check the status of all the orders for the day.

Cake and the orders placed in cake_orders_table will have three status.

PREPARING: Just after the order has been placed.

BAKED: A while later, after the cake is ready for pickup.

Scope

- There are two actors in the system: ADMIN and CUSTOMER.
- The Admin has to login into the system.
- The admin can decide the menu for the day, review the progress of the orders and declare the end of the day when all the orders are complete.
- Customers can order from the available menu, check the progress and number of orders.

User Characteristics in the system

CUSTOMER

- ⇒ Order a cake giving their apartment number as reference with their choice of toppings and type of cake.
- ⇒ Check the status of their order using their apartment number.

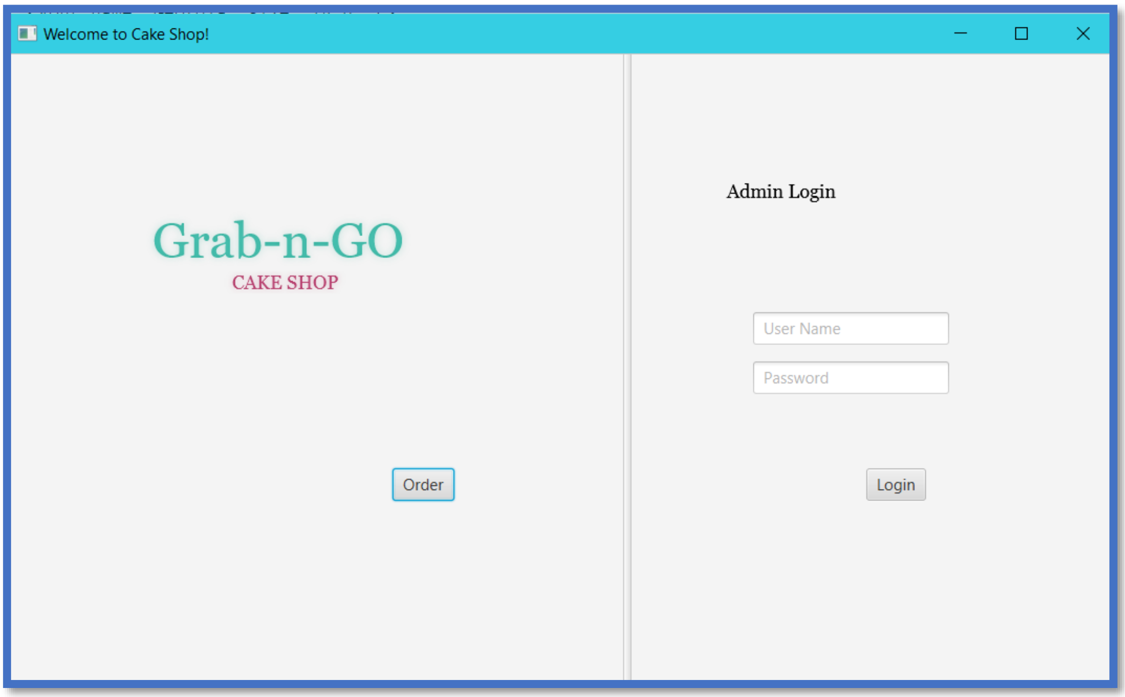
ADMIN

- ⇒ Reinitialize the database: drops all orders, topping, and type of cake
- ⇒ List toppings
- ⇒ Add a topping
- ⇒ Delete a topping
- ⇒ List cake types
- ⇒ Add a cake type
- ⇒ Delete a cake type
- ⇒ Mark the oldest PREPARING cake order as BAKED
- ⇒ Fulfill all orders for the day and start new for the next day. All orders for the day are marked FINISHED.
- ⇒ Output a list of cake orders in preparation (the ones that are not FINISHED).

List of classes

- ⇒ AdminService
- ⇒ Login
- ⇒ CakeOrderData
- ⇒ ServiceException
- ⇒ Service

Home Page



Admin Services

The admin has to log in into the system with his credentials (**Username: admin, Password: 1234**) and manage the store through various functionalities. He can initialize the database, manage the available orders and manage toppings and cake types for the day. He can also tell when all the orders for a day is fulfilled.

The actions performed by Admin can include:

initializedDb()

addTopping()

removeTopping()

addCakeType()

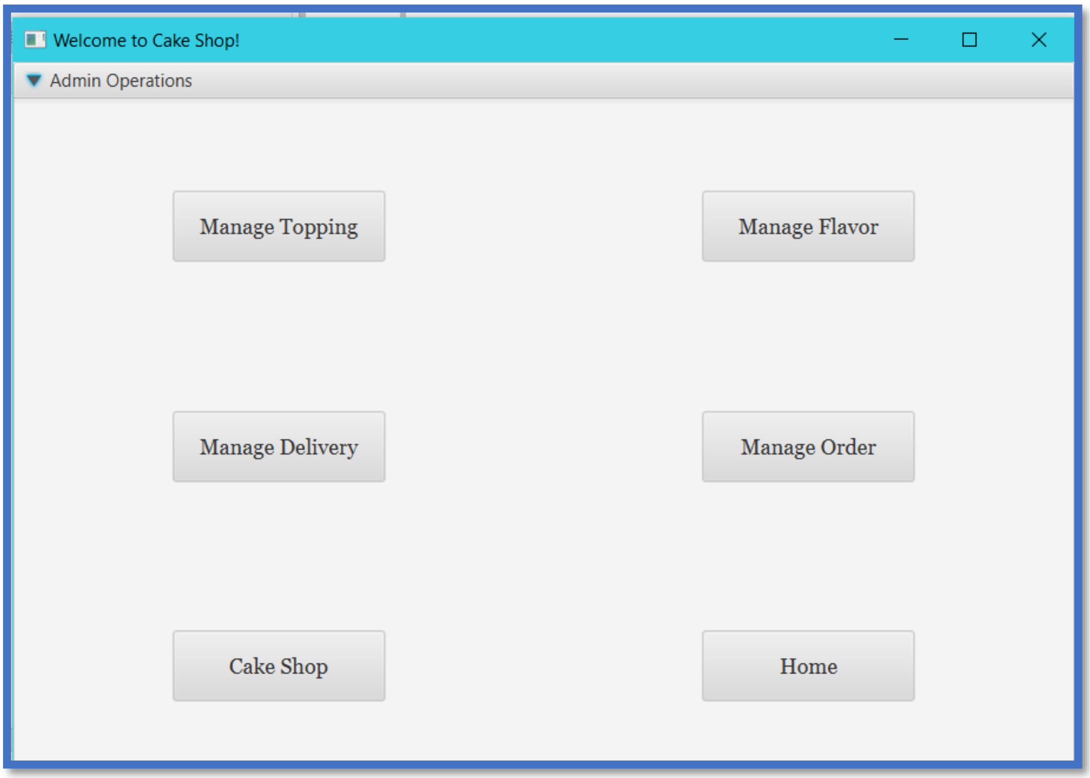
removeCakeType()

markNextOrderReady()

getCurrentDay()

advanceDay()

The window shows the functionalities available for the admin:



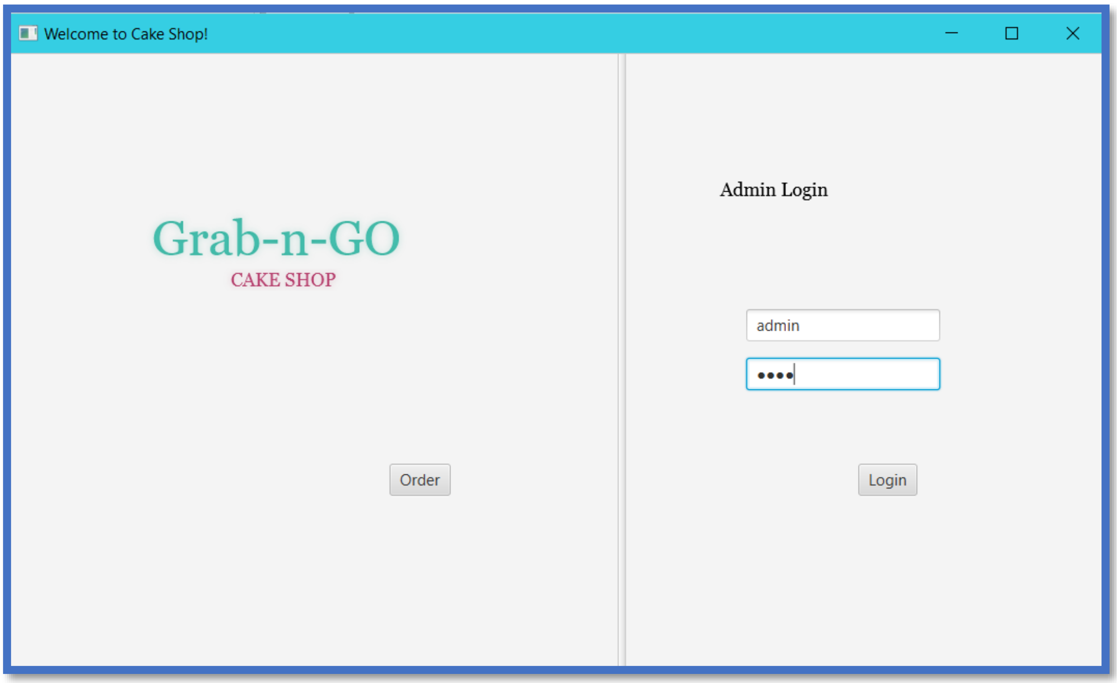
Login

The admin logs in into the system with his credentials:

Username: admin

Password: 1234

Customers need not login into the system for placing an order. Both users have their specific role, but Admin dominates most of the operation.



Service

The customers need not log into the system. They can just place an order. A customer will enter the details such as location, toppings and type of cake while placing an order. This class hold the functionalities by the customers.

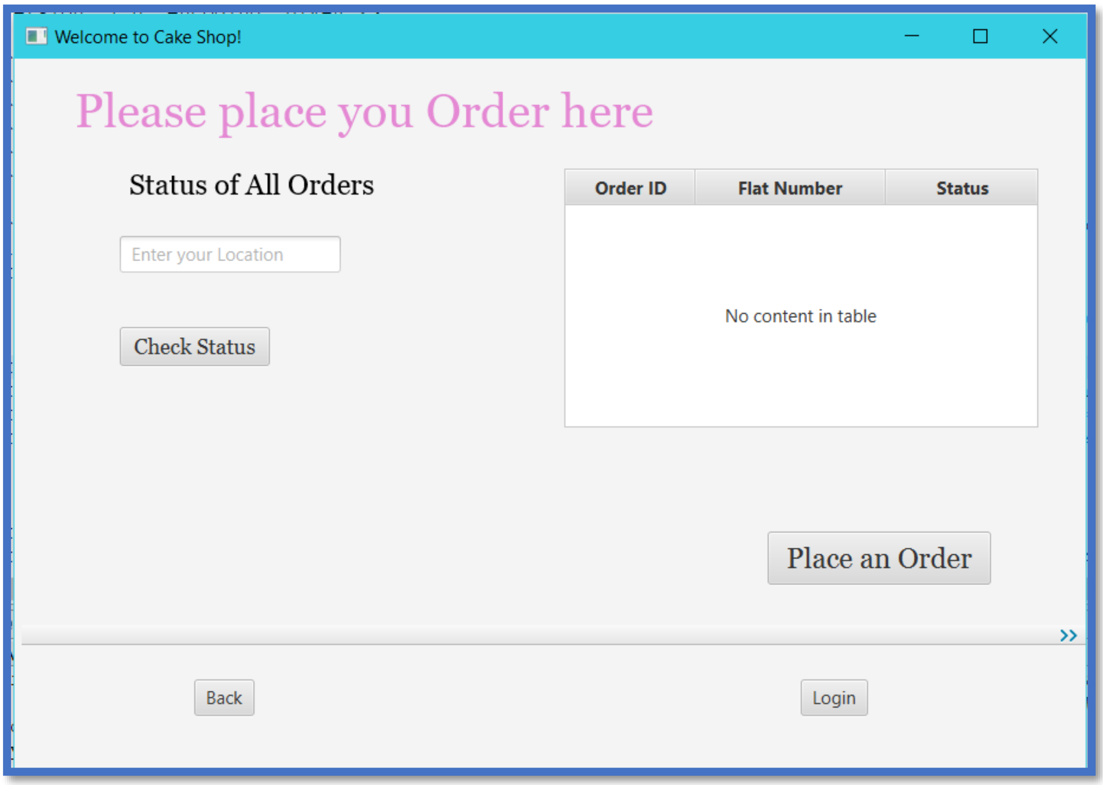
studentService()

makeOrder()

getOrderStatus()

receiveOrders()

The student can select a topping, a flavor and order a cake to one in the list of room numbers.

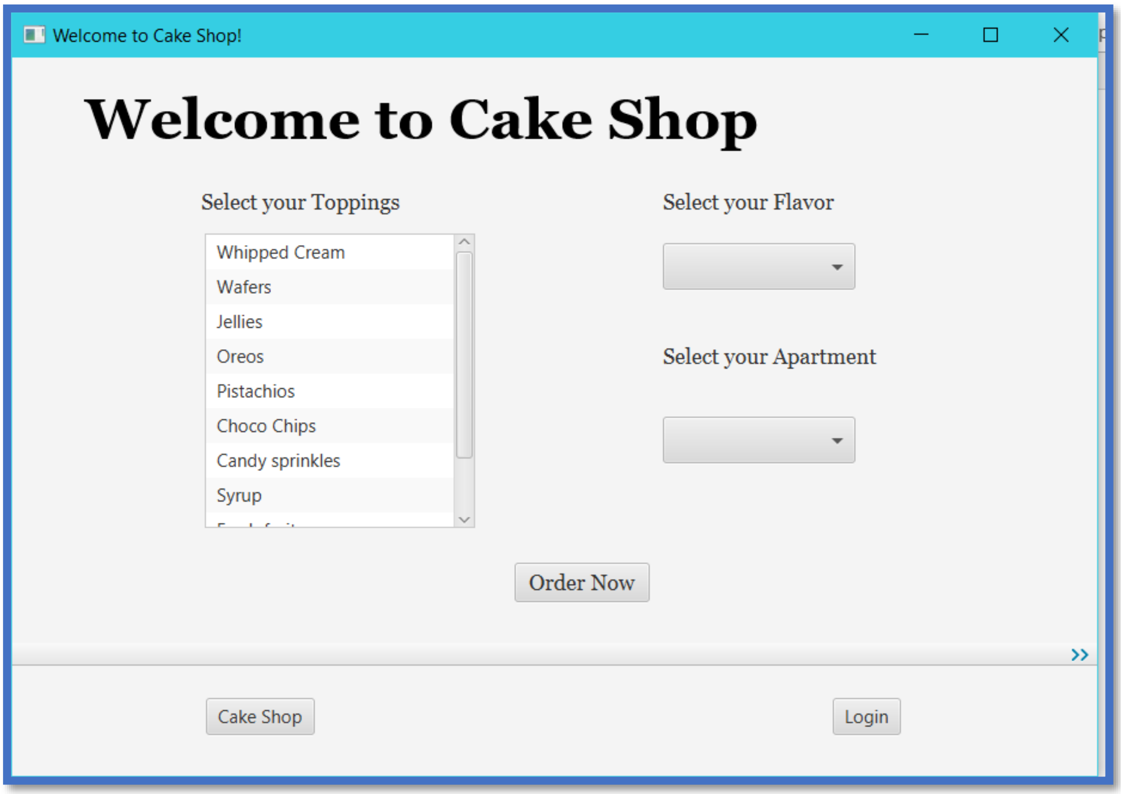


Please place you Order here

Status of All Orders

Order ID	Flat Number	Status

No content in table



Welcome to Cake Shop

Select your Toppings

Select your Flavor

Select your Apartment

CakeOrderData

Once customer places an order this class gets and holds all the details from the order such as, the order id, location, day, cake type and status of the order. The Location, flavor, and toppings for the order will be entered by the customer.

CakeOrderData()

getToppings()

getDay()

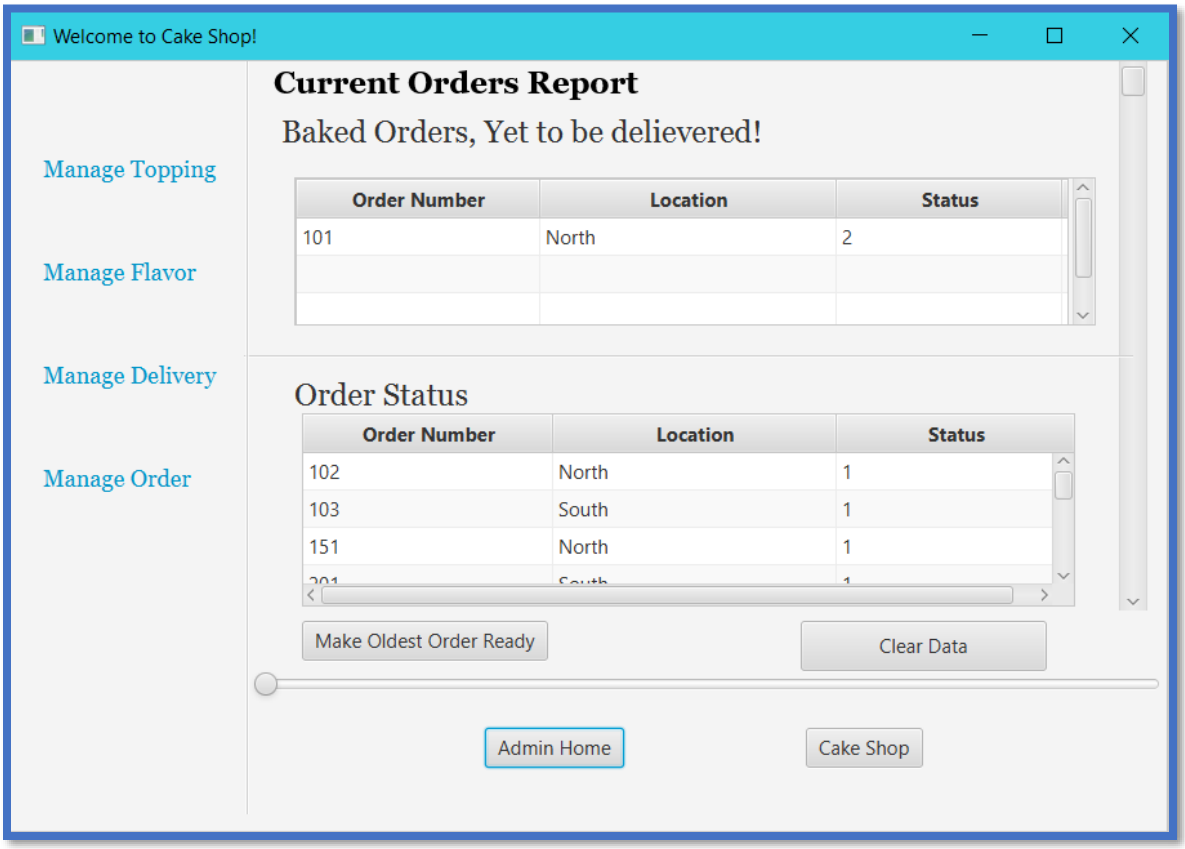
getRoomNumber()

getCakeType()

getId()

getStatus()

All the order details can be accessed from here



Current Orders Report

Baked Orders, Yet to be delievered!

Order Number	Location	Status
101	North	2

Order Status

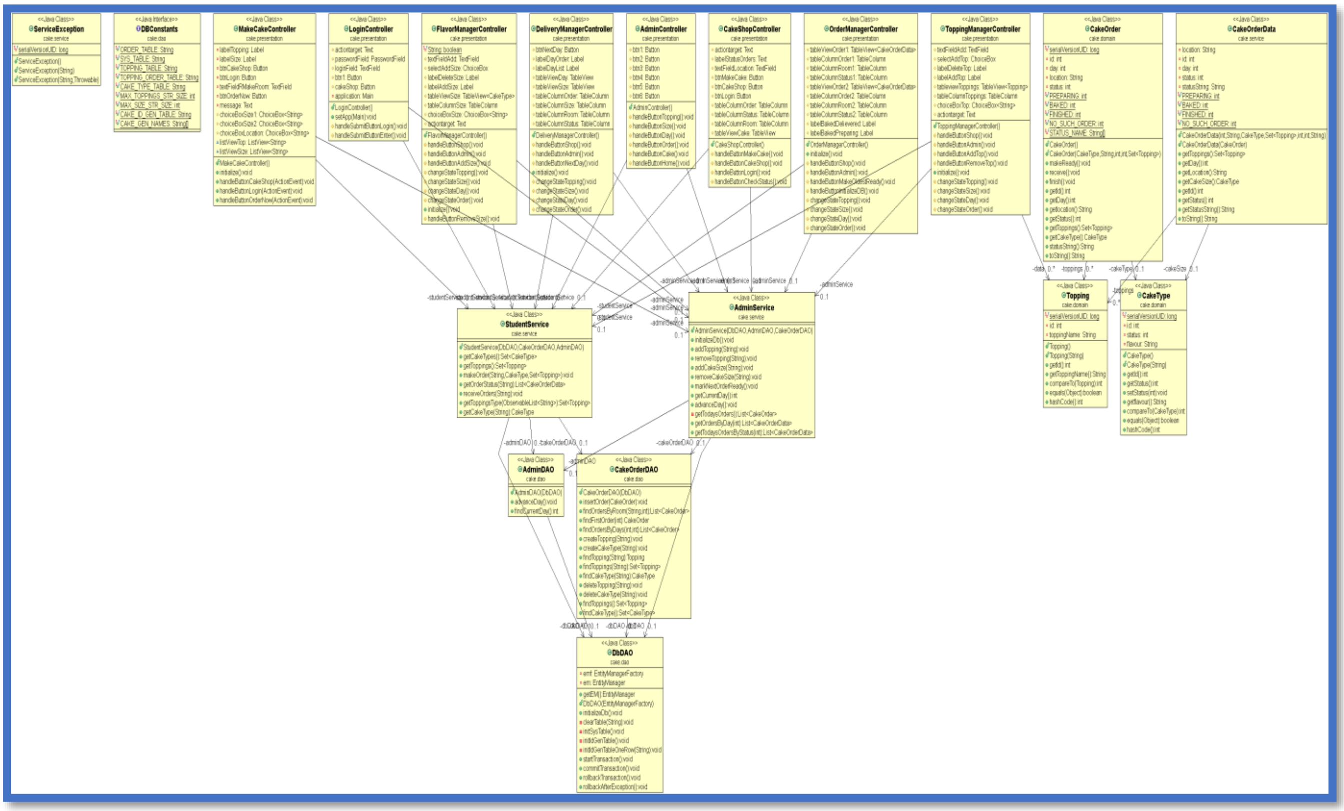
Order Number	Location	Status
102	North	1
103	South	1
151	North	1
201	South	1

Database tables

The following is the list of database tables used:

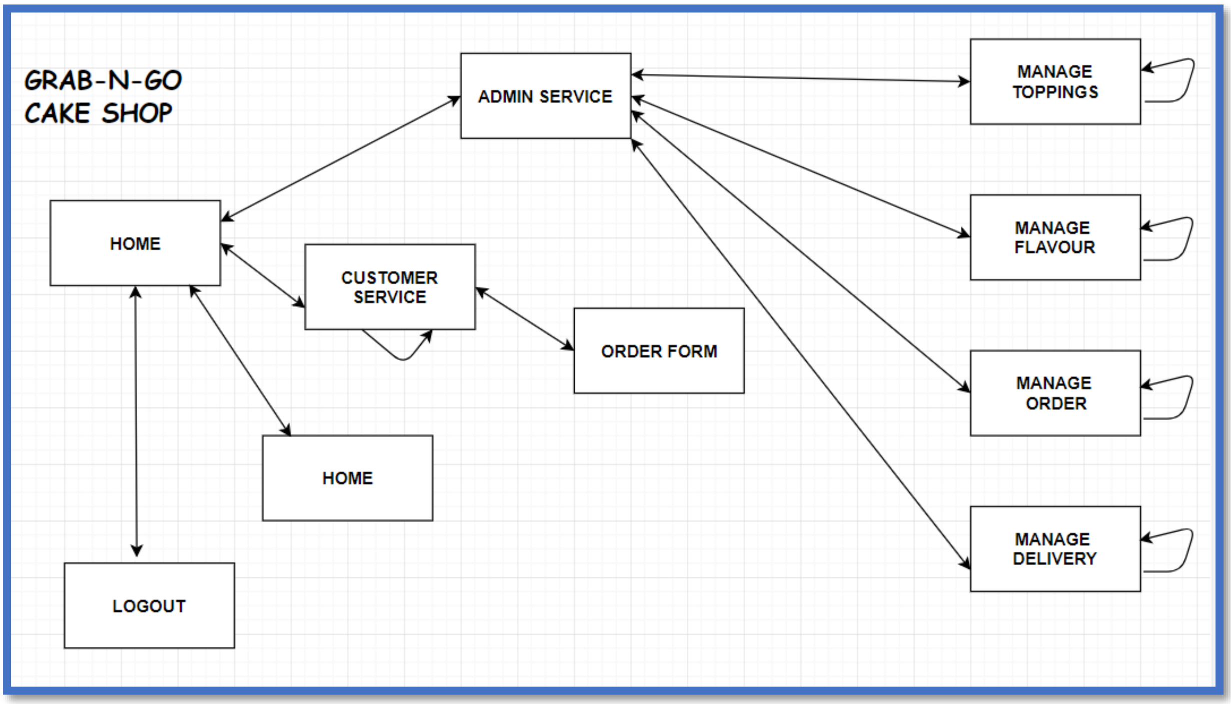
- cake_types
- cake_orders_table
- cake_toppings
- order_toppings
- cake_system_tab
- cake_id_gen

UML Diagram



Flow of pages

The below diagrams gives us an abstract of the flow of pages in the project:



Concepts Implemented

Collections

Collections have been implemented while developing the code for adding cake toppings, adding cake type and maintaining the orders.

CRUD operations in multiple roles

The CRUD option is being implemented for Admin and Student Roles. Then student role has been limited according to his needs.

For Extra Credit, in the Admin Role, CRUD is being implemented for more than one entity, such as updating toppings for the day, updating types of cakes available for the day, initializing the database, updating orders and fulfilling the day's orders.

Meaningful packages and MVC pattern

MVC pattern is implemented under meaningfully named packages and the data flows from the front end in the View package and then from getters and setters under the Model package to Controllers under the Controller package to DAO and then to the database.

Polymorphism

Polymorphism and Overriding is implemented by having class with same name and then passing different arguments for different operations. For instance, in the class CakeOrderDAO, there are two methods called findToppings in the same class.

FXML

The pages in fxml navigates through different entities by coding in the file cake.presentation and navigating through them.

DAO package

The DAO classes are implemented in the DAO package. All the various database operations are performed in the DAO. The methods are passed from the controllers.

GUI with combo boxes

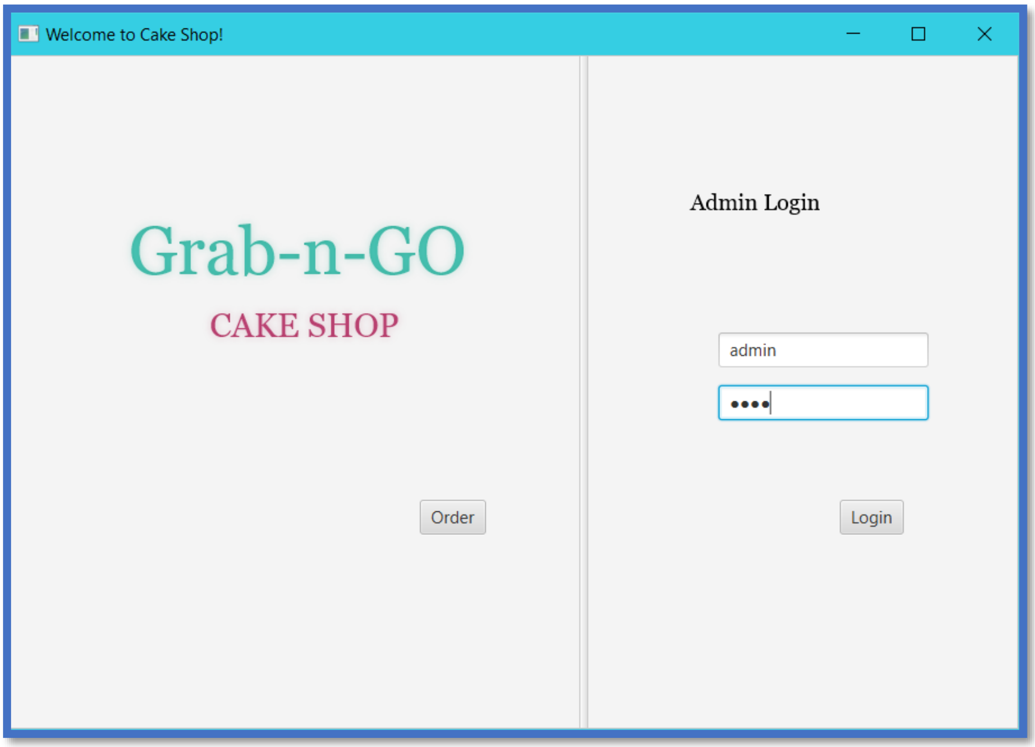
For Extra credit, the GUI has been designed with the CSS styling and components such as ComboBoxes are implemented as shown in the below screenshots.

General flow of the project

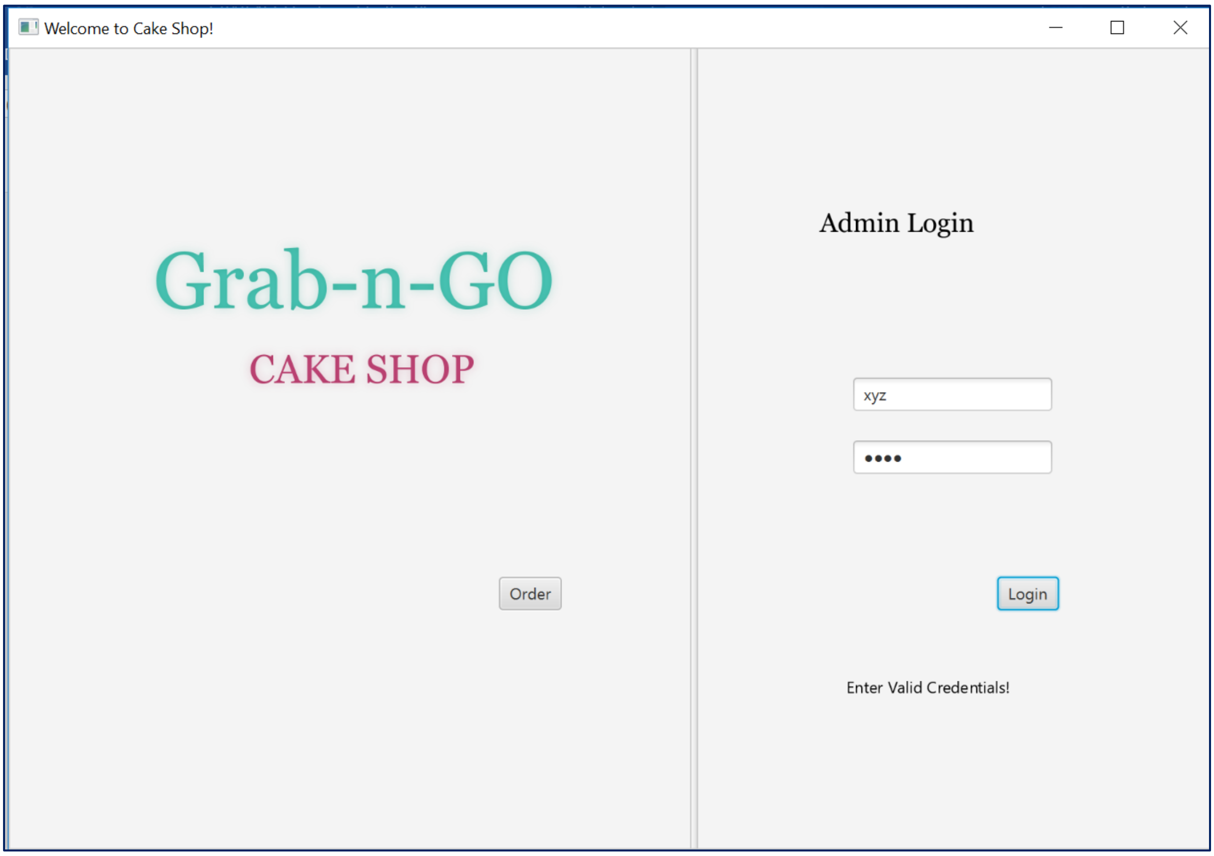
The admin logs into the system:

Username: admin

Password: 1234

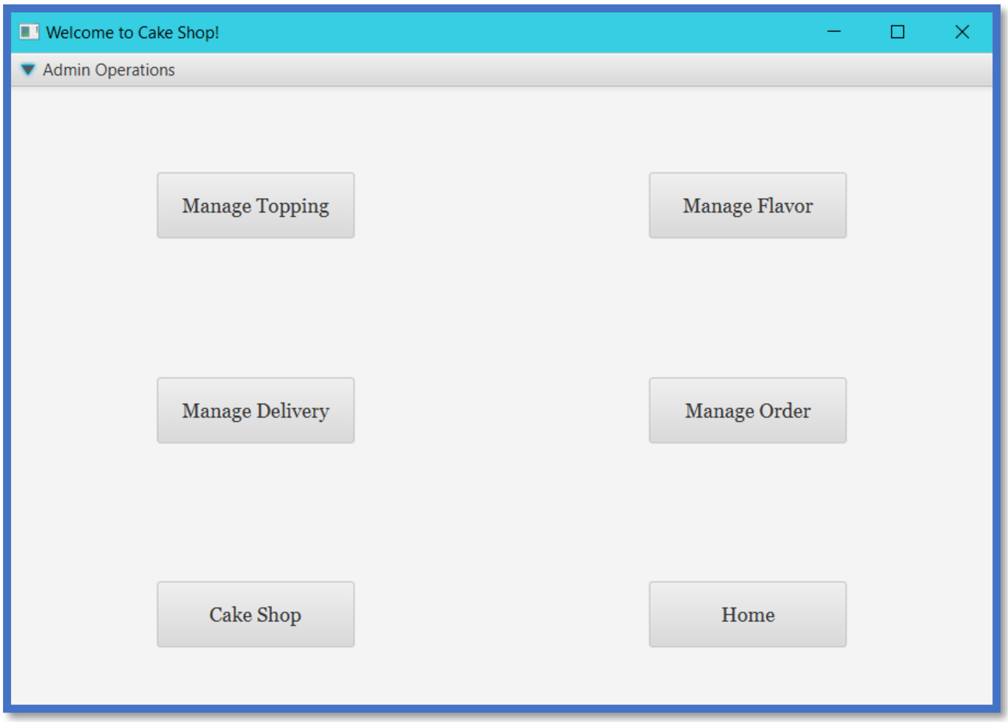


Invalid credentials

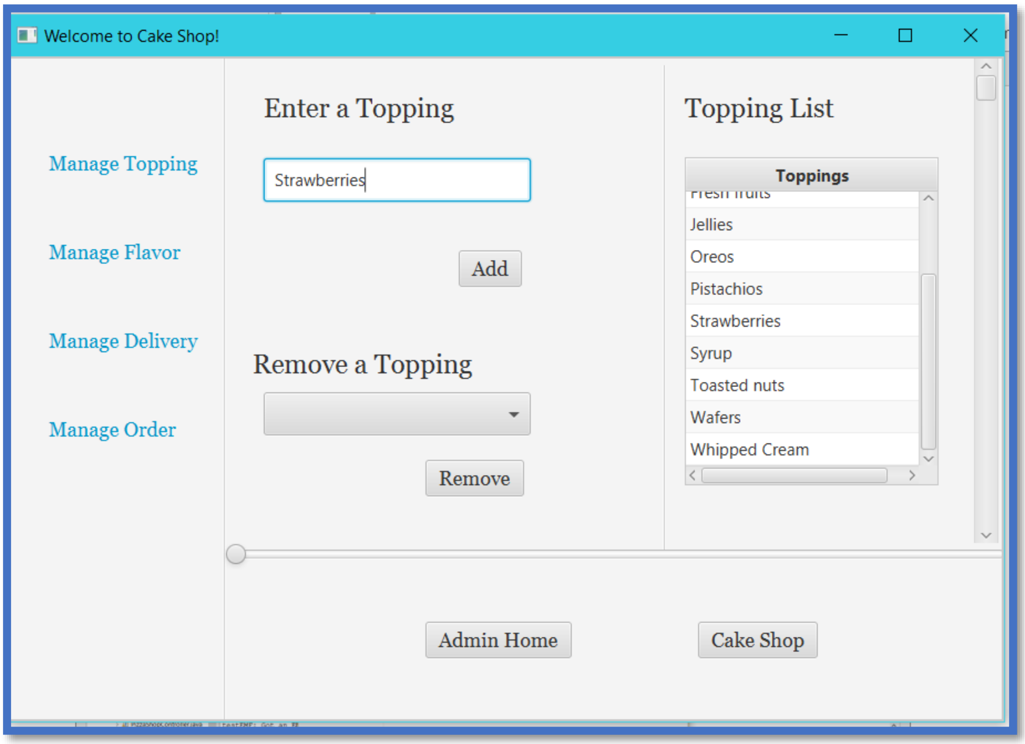


Enter Valid Credentials!

The set of operations the admin can do:



Pick a topping and add:

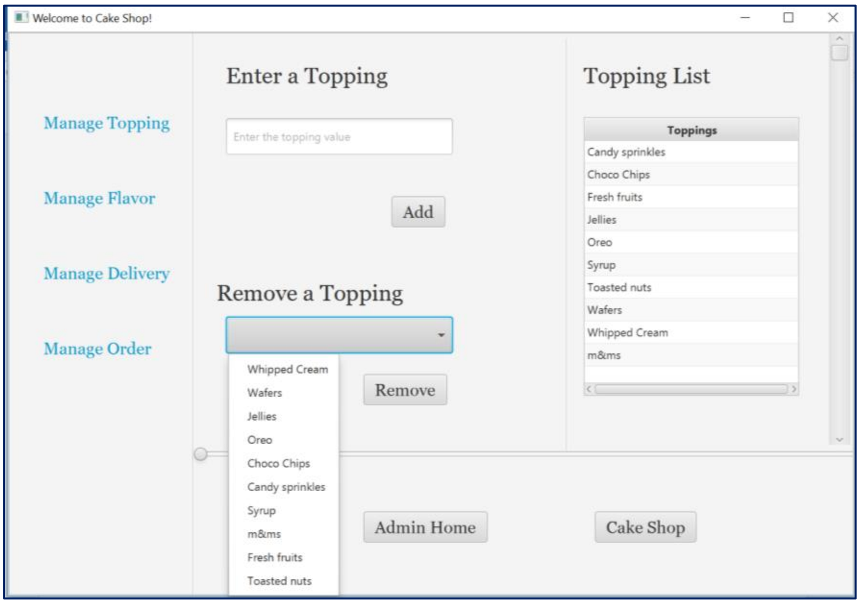


Strawberries

Toppings

Fresh fruits
Jellies
Oreos
Pistachios
Strawberries
Syrup
Toasted nuts
Wafers
Whipped Cream

Delete a topping



Welcome to Cake Shop!

Manage Topping

Manage Flavor

Manage Delivery

Manage Order

Enter a Topping

Enter the topping value

Add

Remove a Topping

- Whipped Cream
- Wafers
- Jellies
- Oreo
- Choco Chips
- Candy sprinkles
- Syrup
- m&ms
- Fresh fruits
- Toasted nuts

Remove

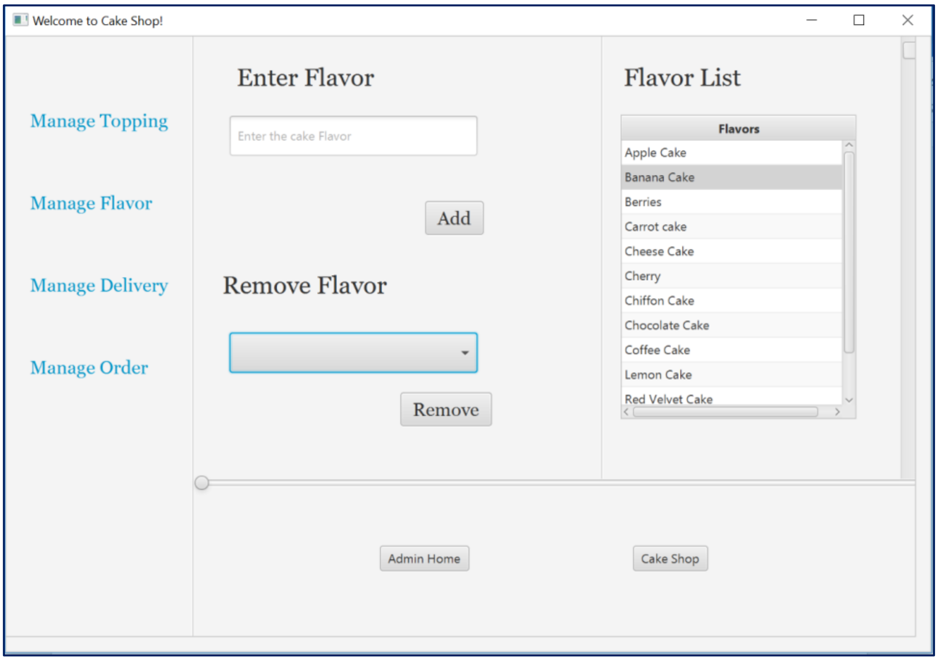
Topping List

Toppings
Candy sprinkles
Choco Chips
Fresh fruits
Jellies
Oreo
Syrup
Toasted nuts
Wafers
Whipped Cream
m&ms

Admin Home

Cake Shop

Pick a flavor / Cake type:



Welcome to Cake Shop!

Manage Topping

Manage Flavor

Manage Delivery

Manage Order

Enter Flavor

Enter the cake Flavor

Add

Remove Flavor

Remove

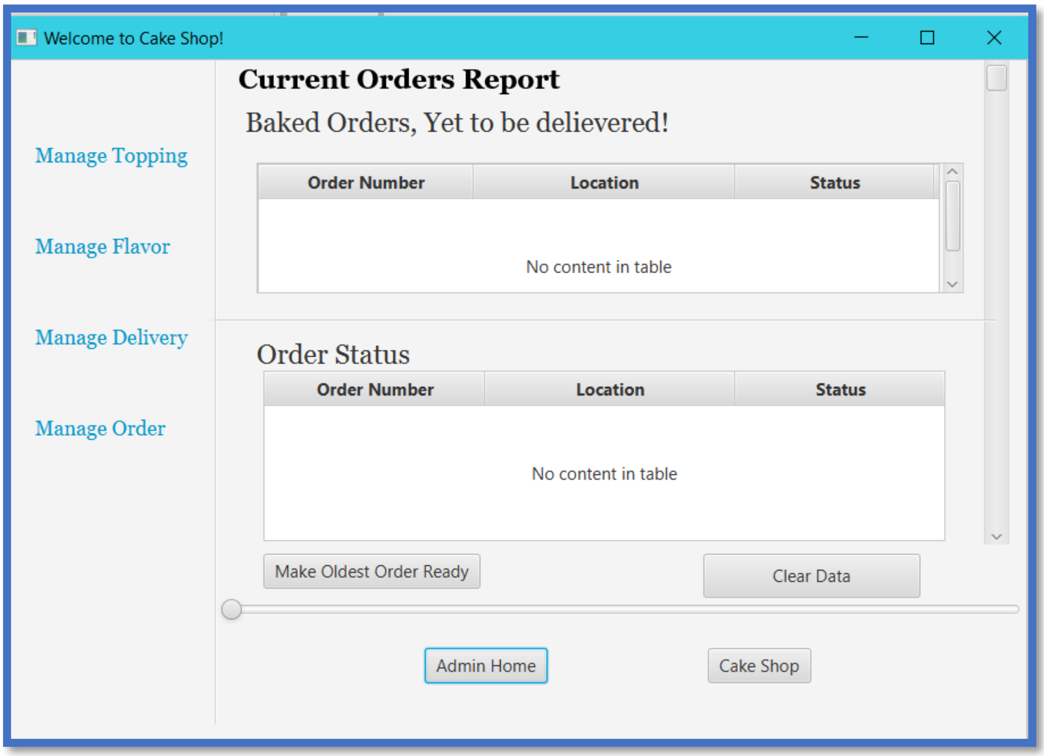
Flavor List

Flavors
Apple Cake
Banana Cake
Berries
Carrot cake
Cheese Cake
Cherry
Chiffon Cake
Chocolate Cake
Coffee Cake
Lemon Cake
Red Velvet Cake

Admin Home

Cake Shop

No orders placed yet:



Welcome to Cake Shop!

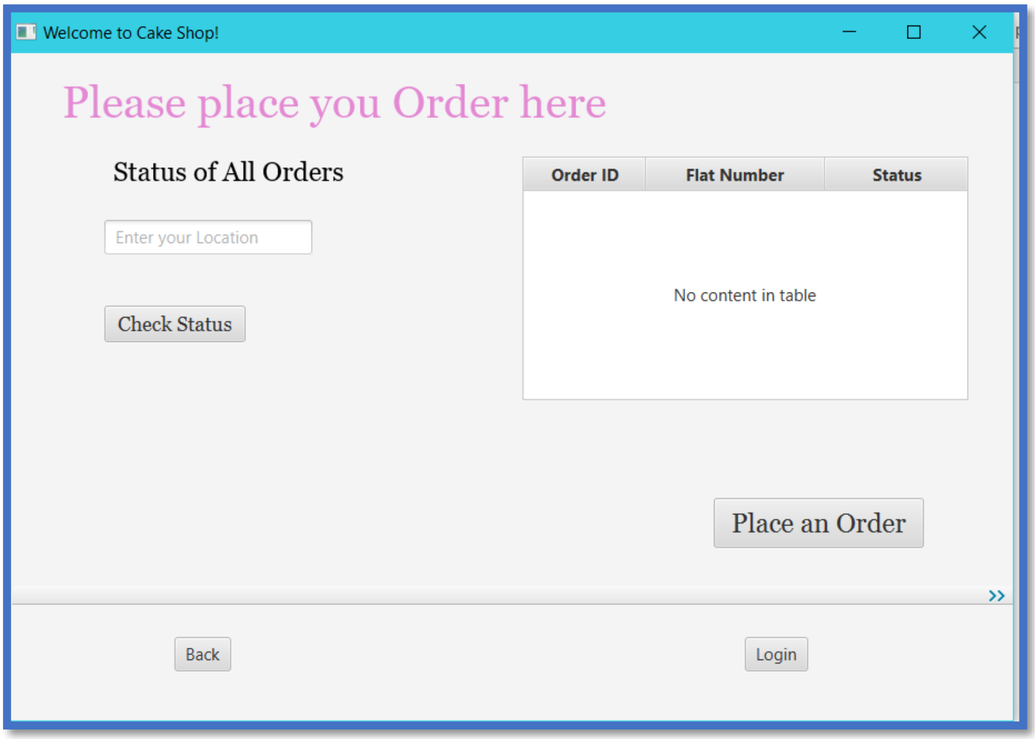
Current Orders Report

Baked Orders, Yet to be delievered!

Order Number	Location	Status
No content in table		

Order Status

Order Number	Location	Status
No content in table		



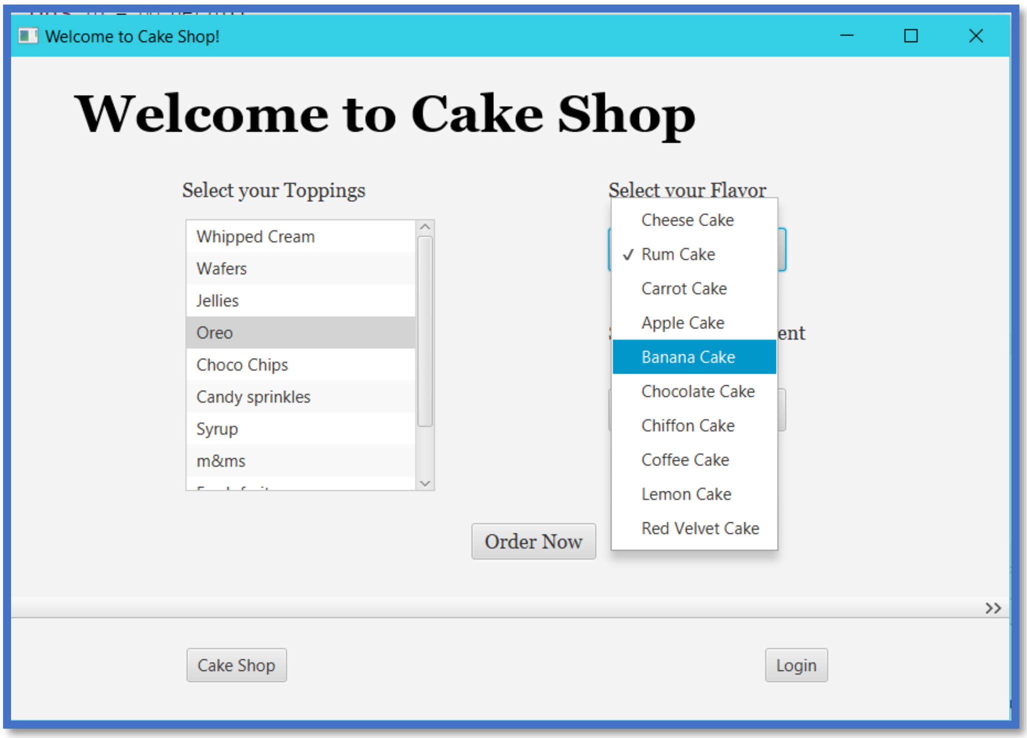
Welcome to Cake Shop!

Please place you Order here

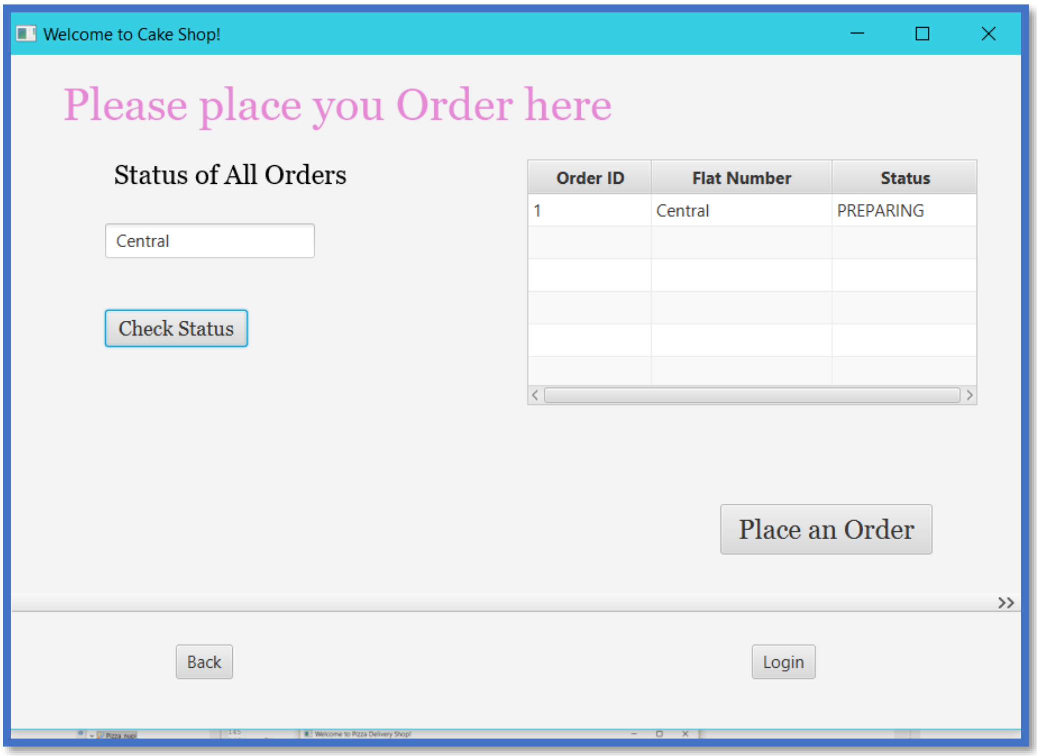
Status of All Orders

Order ID	Flat Number	Status
No content in table		

Customer places an order:



Status of the order:

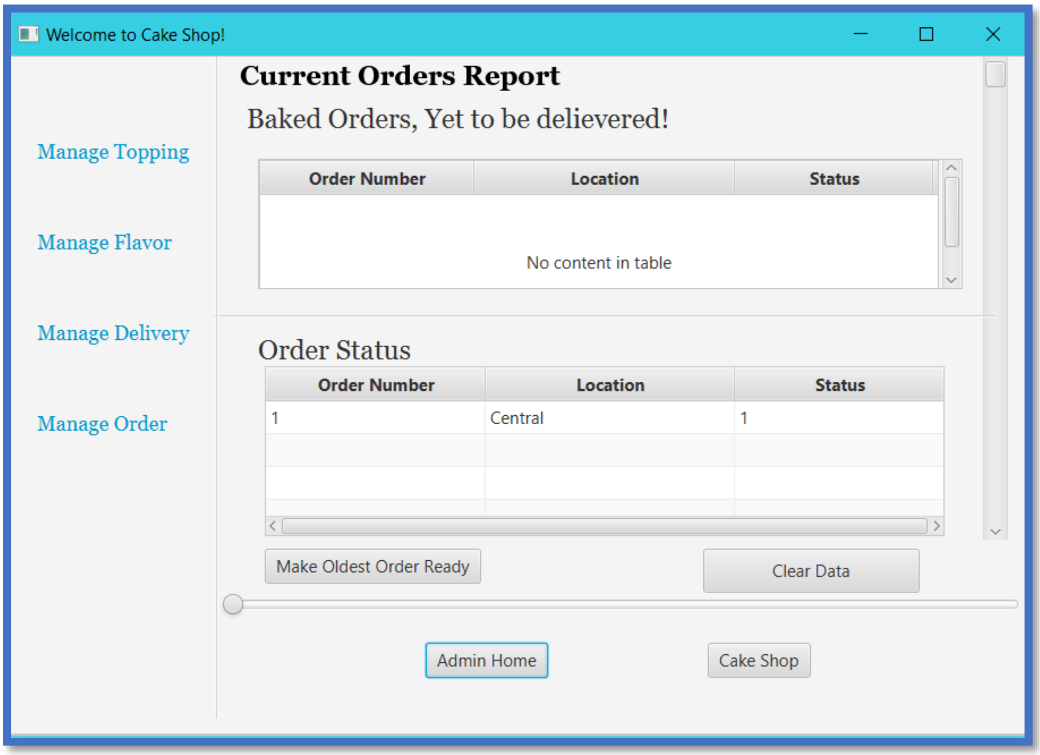


Please place you Order here

Status of All Orders

Central

Order ID	Flat Number	Status
1	Central	PREPARING



Current Orders Report

Baked Orders, Yet to be delievered!

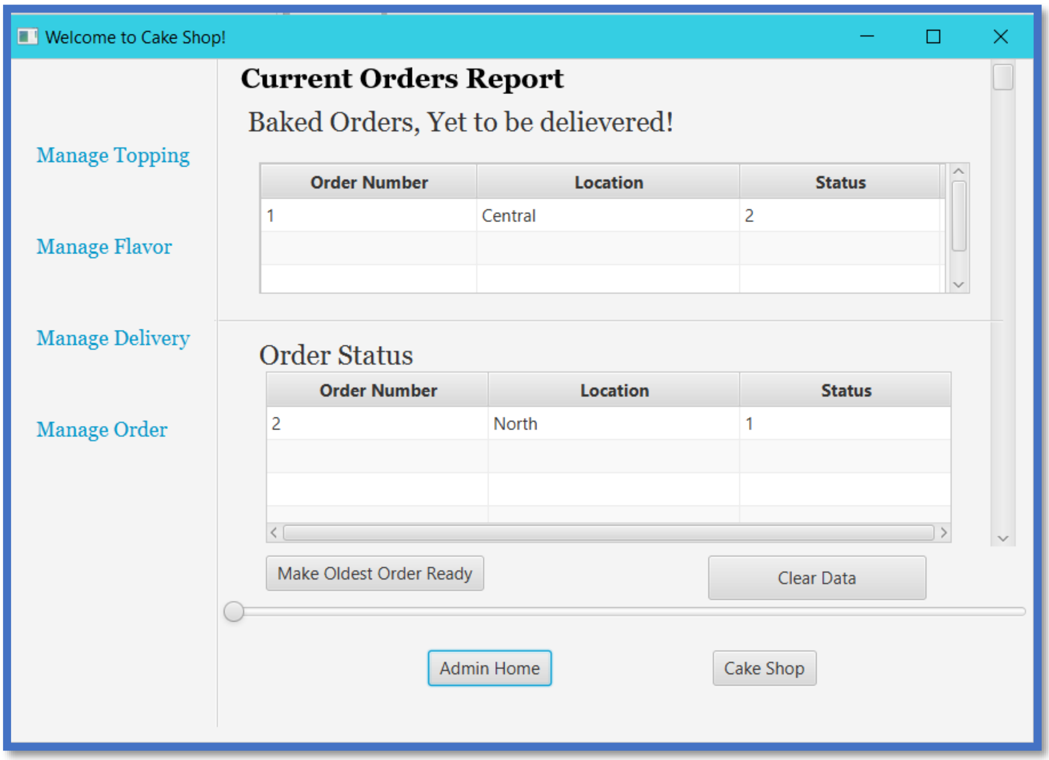
Order Number	Location	Status

No content in table

Order Status

Order Number	Location	Status
1	Central	1

Admin updates the order:



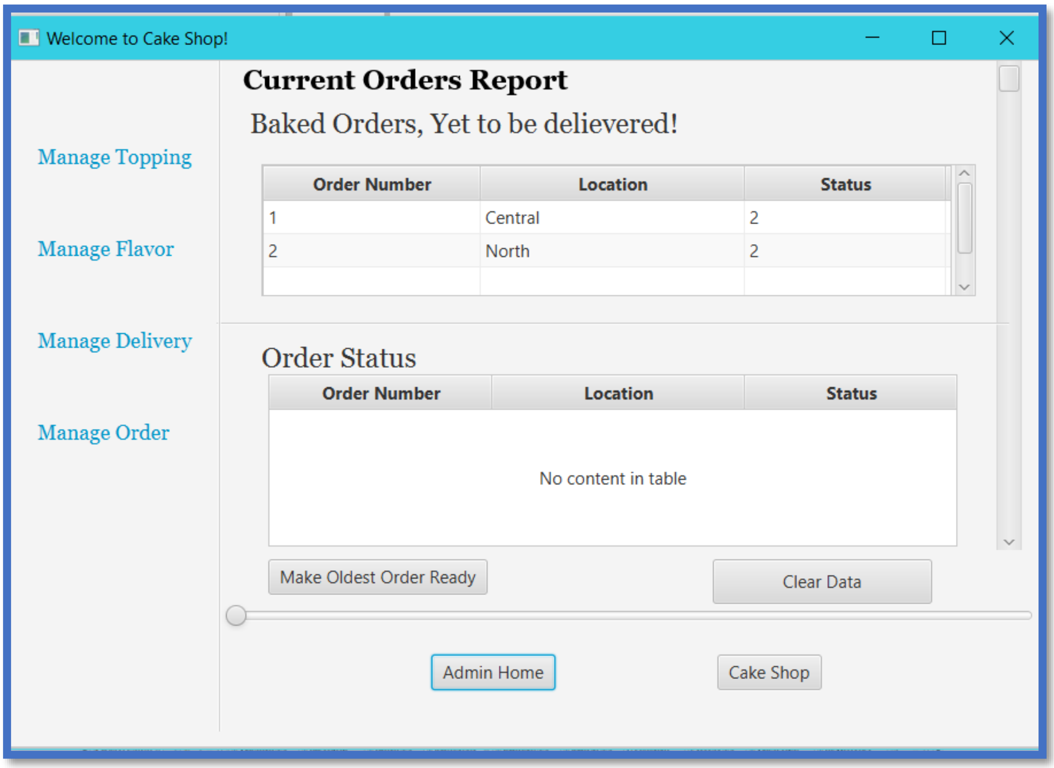
Current Orders Report

Baked Orders, Yet to be delievered!

Order Number	Location	Status
1	Central	2

Order Status

Order Number	Location	Status
2	North	1



Current Orders Report

Baked Orders, Yet to be delievered!

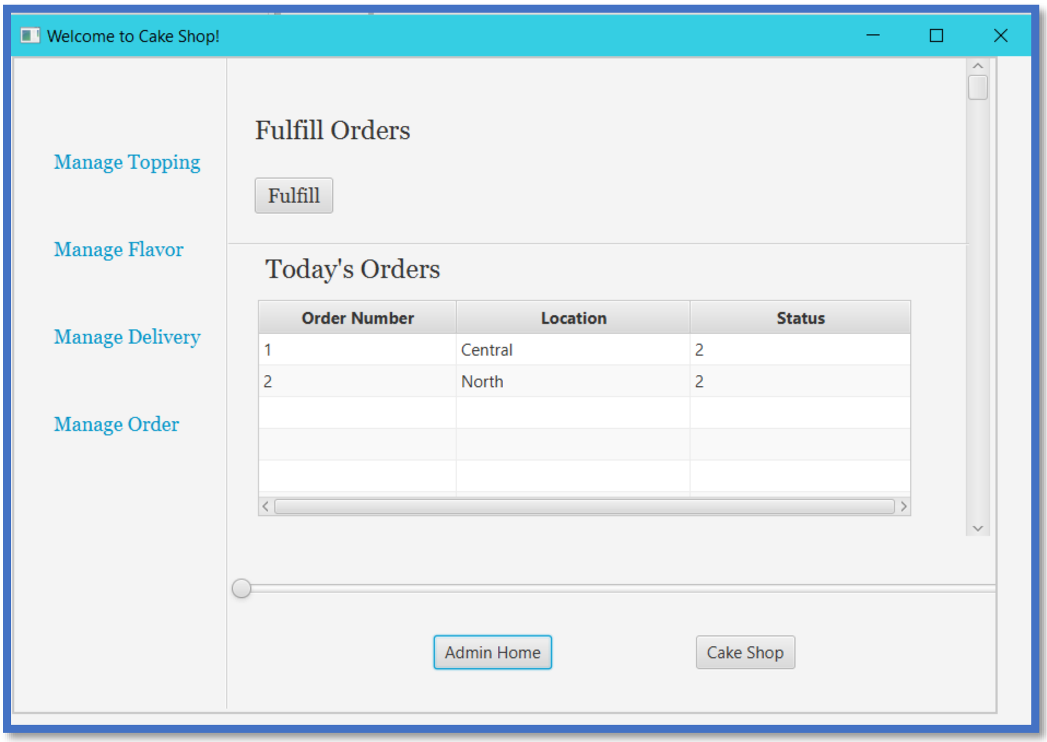
Order Number	Location	Status
1	Central	2
2	North	2

Order Status

Order Number	Location	Status

No content in table

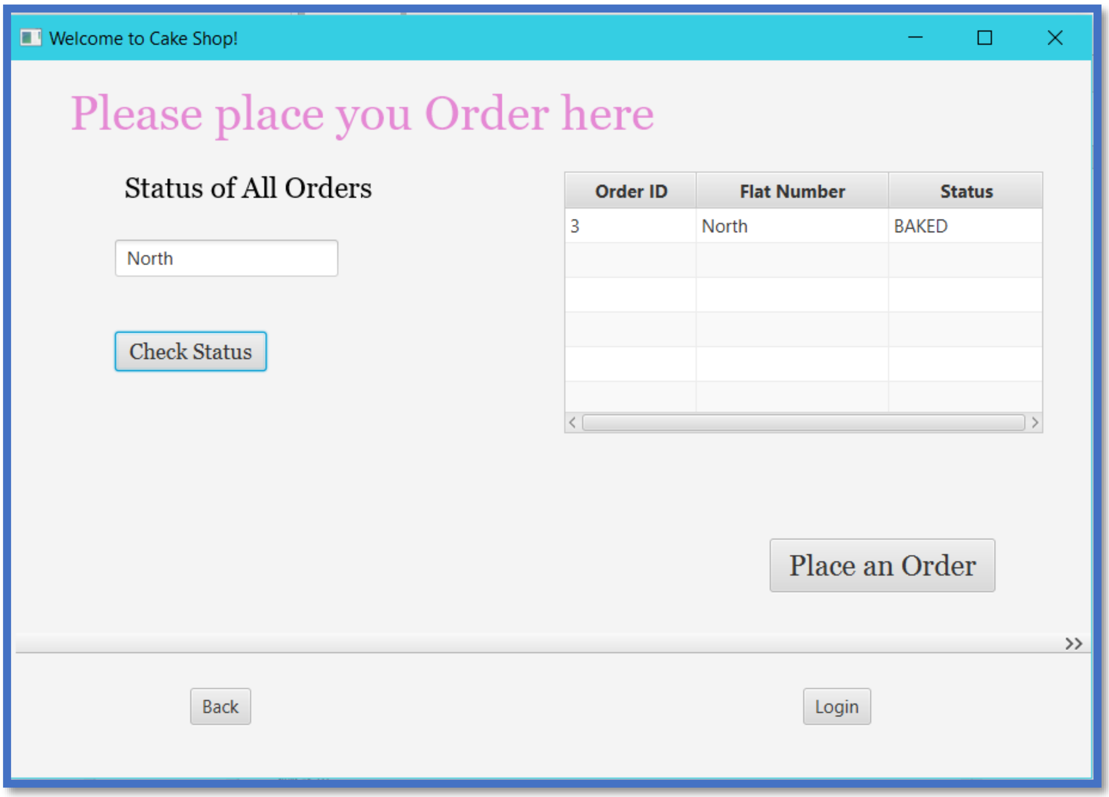
The order shows baked:



Fulfill Orders

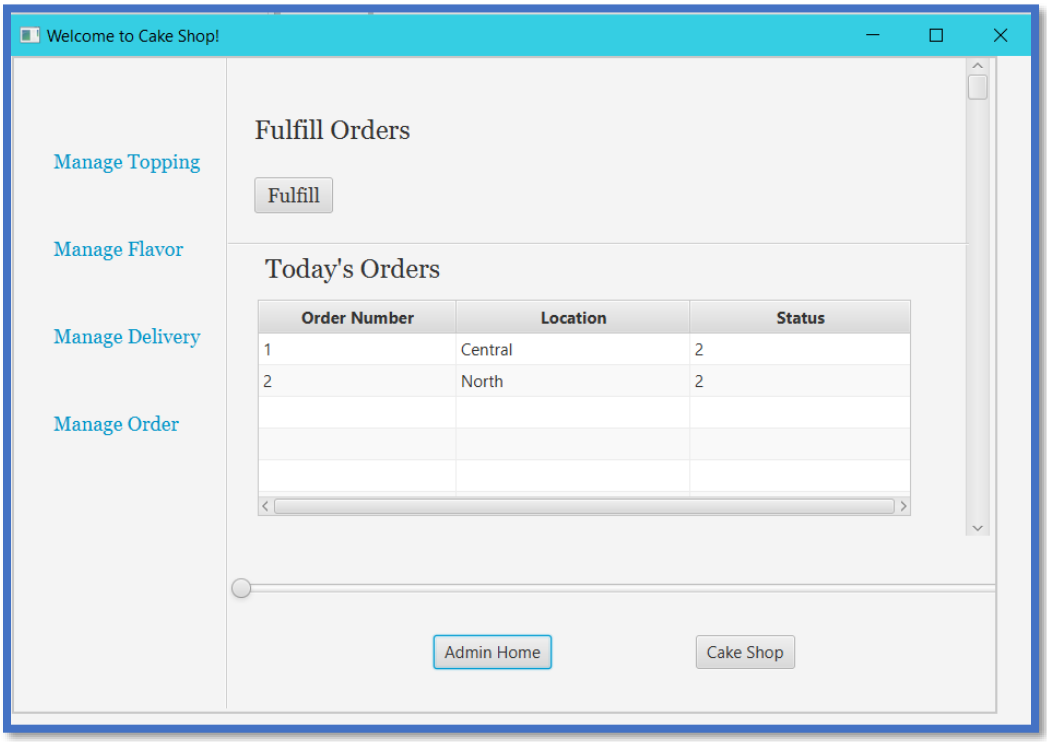
Today's Orders

Order Number	Location	Status
1	Central	2
2	North	2

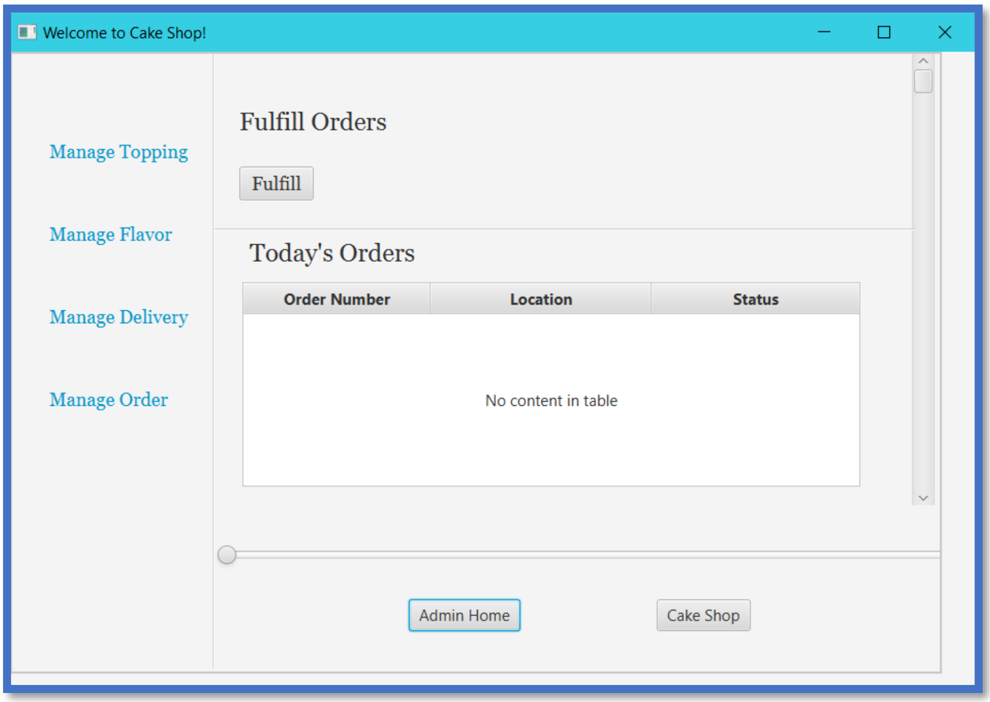


Order ID	Flat Number	Status
3	North	BAKED

To fulfill current day’s orders



Order Number	Location	Status
1	Central	2
2	North	2



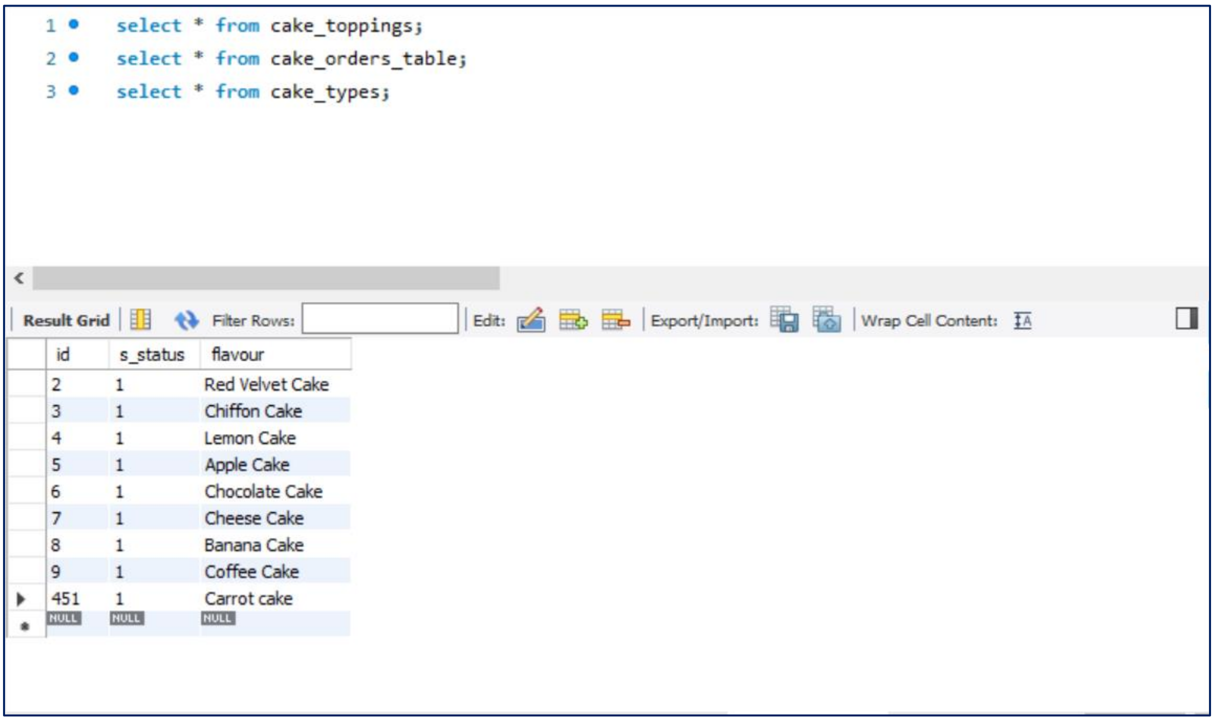
Fulfill Orders

Today's Orders

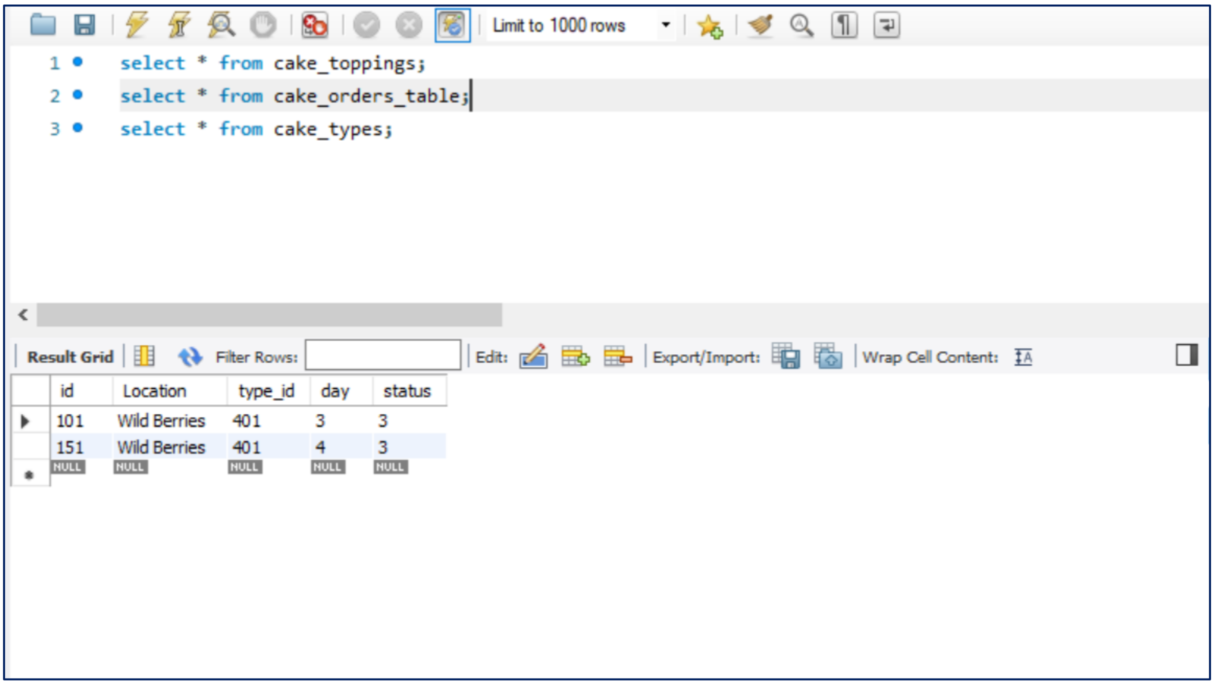
Order Number	Location	Status

No content in table

Database

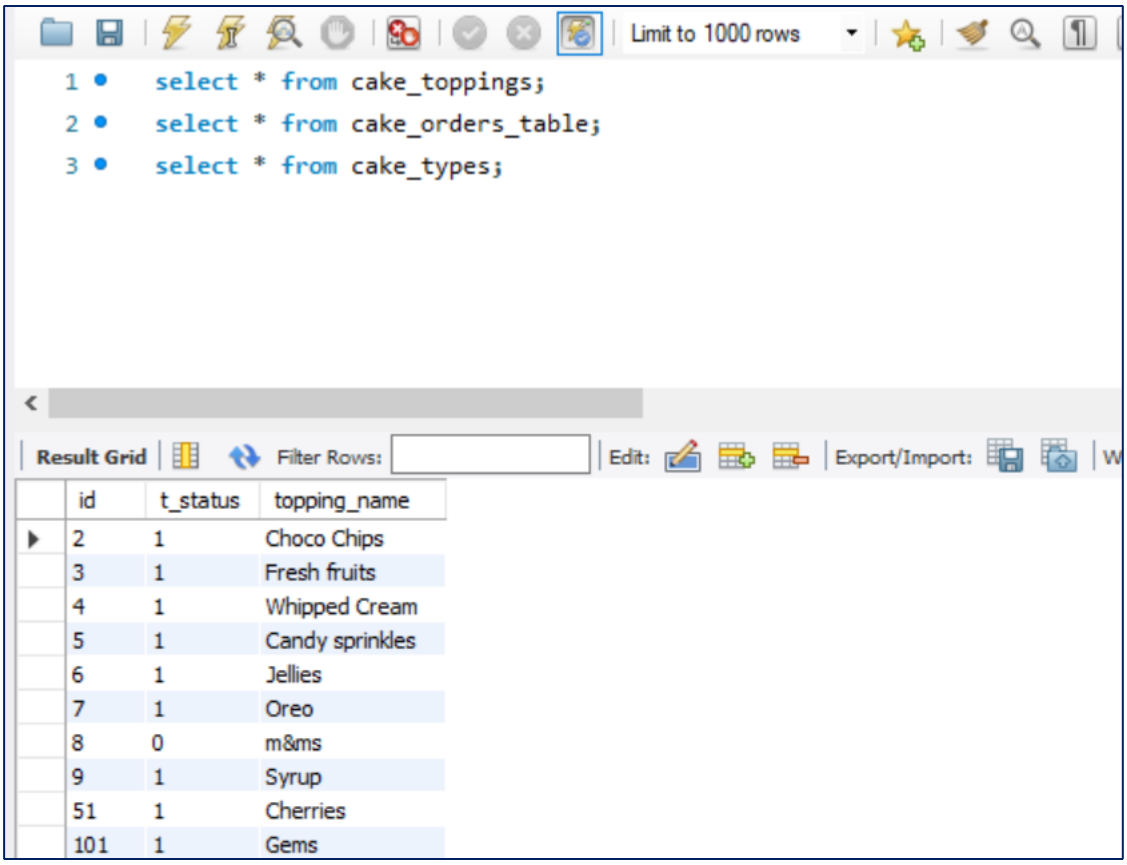

```
select * from cake_toppings;
select * from cake_orders_table;
select * from cake_types;
```

id	s_status	flavour
2	1	Red Velvet Cake
3	1	Chiffon Cake
4	1	Lemon Cake
5	1	Apple Cake
6	1	Chocolate Cake
7	1	Cheese Cake
8	1	Banana Cake
9	1	Coffee Cake
451	1	Carrot cake
NULL	NULL	NULL



```sql
select * from cake_toppings;
select * from cake_orders_table;
select * from cake_types;
```

id	Location	type_id	day	status
101	Wild Berries	401	3	3
151	Wild Berries	401	4	3
NULL	NULL	NULL	NULL	NULL



```sql
select * from cake_toppings;
select * from cake_orders_table;
select * from cake_types;
```

id	t_status	topping_name
2	1	Choco Chips
3	1	Fresh fruits
4	1	Whipped Cream
5	1	Candy sprinkles
6	1	Jellies
7	1	Oreo
8	0	m&ms
9	1	Syrup
51	1	Cherries
101	1	Gems