

The operation and setup of a basic restaurant reservation system are described in this study. The system is made up of a Python backend that communicates with a MySQL database to handle reservations and customer data. Python's HTTP server is used to deliver a web portal that provides access to the system. Schema for Databases:

Customers Table: Contains contact information and the name of the customer.

Reservations Table: Contains information about reservations, such as customer ID, time, number of guests, and special requests. Dining Preferences Table: Keeps track of patrons' dietary needs and preferred tables for meals.

Classes and Methods in Python:

RestaurantDatabase: Manages all database functions, such as establishing a connection, adding clients, adding, retrieving, and removing reservations.

To let the site be active:

Run the Server: Run the Python script to launch the server.

Open a web browser and go to <http://127.0.0.1:8000/> to access the server.

Add Customer: To access the customer form and submit new customers, click the "Add Customer" link.

Add Reservation: To view the reservation form and make new reservations, click the "Add Reservation" link.

View Reservations: To view every reservation, click the "View Reservations" link.

Delete Reservation: To access the form and delete reservations, click the "Delete Reservation" link.

In summary, A simple yet useful interface is offered by this restaurant reservation system for handling patrons and bookings. The system offers an intuitive web interface for user interaction and guarantees accurate data storage and retrieval from the database. Future enhancements might include improved UI/UX for a better user experience, user authentication, and more resilient error handling.

## Restaurant Portal

[Home](#) | [Add Reservation](#) | [View Reservations](#) | [Delete Reservations](#)

### All Reservations

Reservation ID	Customer Name	Contact Info	Reservation Time	Number of Guests	Special Requests
13	Ace Ronaldo	917.333.2233	2024-09-07 13:30:00	4	none
16	Ace zoro	989.087.01893	2024-06-05 16:30:00	8	none
17	Ace Messi	917.659.4402	2024-12-09 13:30:00	9	none
18	James Rodriguez	918.922.9209	2024-05-23 05:00:00	9	growth of hair
19	Green Tea	918.234.6758	2024-09-23 13:00:00	7	pride month

Delete Reservation

127.0.0.1:8000/deleteReservationForm

☆

Reservation ID:

Delete Reservation

All Reservations

Reservation ID	Customer Name	Contact Info	Reservation Time	Number of Guests	Special Requests
13	Ace Ronaldo	917.333.2233	2024-09-07 13:30:00	4	none
16	Ace zoro	989.087.01893	2024-06-05 16:30:00	8	none
17	Ace Messi	917.659.4402	2024-12-09 13:30:00	9	none
18	James Rodriguez	918.922.9209	2024-05-23 05:00:00	9	growth of hair
19	Green Tea	918.234.6758	2024-09-23 13:00:00	7	pride month

Customer ID:

Reservation Time:

Number of Guests:

Special Requests:

Add Reservation

← → ↻ github.com/Keion-12/CIS344-project/blob/main/restaurantDatabase.py

Keion-12 / CIS344-project

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Files

main

Go to file

Final SQLsql  
RestaurantServer.py  
restaurantDatabase.py

CIS344-project / restaurantDatabase.py

Keion-12 restaurantDatabase.py 133484a · 14 minutes ago History

Code Blame 218 lines (195 loc) · 9.42 KB Code 55% faster with GitHub Copilot Raw Download Edit Copy

```
1 import mysql.connector
2 from mysql.connector import Error
3
4 class RestaurantDatabase:
5     def __init__(self, host="127.0.0.1", port="3306", database="restaurant_reservations", user="root", password="PASSWORD"):
6         self.host = host
7         self.port = port
8         self.database = database
9         self.user = user
10        self.password = password
11        self.connection = None
12        self.cursor = None
13        self.connect()
14
15    def connect(self):
16        try:
17            self.connection = mysql.connector.connect(
```

← → ↻ github.com/Keion-12/CIS344-project/blob/main/RestaurantServer.py

Keion-12 / CIS344-project

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Files

main

Go to file

Final SQLsql  
RestaurantServer.py  
restaurantDatabase.py

CIS344-project / RestaurantServer.py

Keion-12 Update RestaurantServer.py 79dee9c · 15 minutes ago History

Code Blame 241 lines (216 loc) · 12.5 KB Code 55% faster with GitHub Copilot Raw Download Edit Copy

```
1 from http.server import HTTPServer, BaseHTTPRequestHandler
2 from restaurantDatabase import RestaurantDatabase
3 import cgi
4
5 class RestaurantPortalHandler(BaseHTTPRequestHandler):
6
7     def __init__(self, *args, **kwargs):
8         self.database = RestaurantDatabase(password="PASSWORD")
9         super().__init__(*args, **kwargs)
10
11    def do_POST(self):
12        try:
13            if self.path == '/addReservation':
14                self.send_response(200)
15                self.send_header('Content-type', 'text/html')
16                self.end_headers()
17                form = cgi.FieldStorage(
18                    fp=self.rfile,
```

github.com/Keion-12/CIS344-project/blob/main/Final%20SQLsql

Keion-12 / CIS344-project

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Files

main

Go to file

Final SQLsql

RestaurantServer.py

restaurantDatabase.py

CIS344-project / Final SQLsql

Keion-12 Add files via upload 15bfe4a · 19 minutes ago History

Code Blame 39 lines (35 loc) · 1.2 KB Code 55% faster with GitHub Copilot

```
1 CREATE DATABASE restaurant_reservations;
2 USE restaurant_reservations;
3 CREATE TABLE Customers (
4     customerId INT NOT NULL UNIQUE AUTO_INCREMENT,
5     customerName VARCHAR(45) NOT NULL,
6     contactInfo VARCHAR(200), -- Stores email or phone number
7     PRIMARY KEY (customerId)
8 );
9 CREATE TABLE Reservations (
10     reservationId INT NOT NULL UNIQUE AUTO_INCREMENT,
11     customerId INT NOT NULL,
12     reservationTime DATETIME NOT NULL,
13     numberOfGuests INT NOT NULL,
14     specialRequests VARCHAR(200),
15     PRIMARY KEY (reservationId),
16     FOREIGN KEY (customerId) REFERENCES Customers(customerId)
17 );
18 CREATE TABLE DiningPreferences (
```

<https://github.com/Keion-12/CIS344-project>