

```
!pip install -q google-generativeai
```

```
import os
import random
import uuid

try:
    import google.generativeai as genai
    GEMINI_AVAILABLE = True
except:
    GEMINI_AVAILABLE = False

# 🔑 OPTIONAL: Add your Gemini API Key
# os.environ["GEMINI_API_KEY"] = "YOUR_API_KEY_HERE"

if "GEMINI_API_KEY" in os.environ:
    genai.configure(api_key=os.environ["GEMINI_API_KEY"])
    model = genai.GenerativeModel("gemini-pro")
else:
    GEMINI_AVAILABLE = False
```

```
def llm_response(text):
    """
    Generates Gen-AI styled responses.
    Falls back to normal text if LLM is unavailable.
    """
    if GEMINI_AVAILABLE:
        try:
            response = model.generate_content(text)
            return response.text
        except:
            return text
    return text
```

```
RESTAURANT_OPEN_TIME = 17 # 5 PM
RESTAURANT_CLOSE_TIME = 23 # 11 PM
```

```
TABLE_TYPES = {
    "single": "1 Seat",
    "couple": "2 Seats",
    "family": "4 to 8 Seats"
}
```

```
RESERVATION_FEE = 100
CURRENT_OFFER = "FREE"
```

```
bookings = {}
```

```
def welcome_message():
    text = (
        "紫色  Welcome to our Restaurant!\n"
        "粉色  I'm your table booking chatbot.\n"
        "I can help you with bookings, availability, and cancellations."
    )
    return llm_response(text)
```

```
def show_table_options():
    text = (
        "紫色 Available Table Types:\n"
        "1 Single Table - 1 Seat\n"
        "2 Couple Table - 2 Seats\n"
        "3 Family Table - 4 to 8 Seats"
    )
    return llm_response(text)
```

```
def check_date_availability(date):
    text = f"绿色勾 Tables are available on {date}. Bookings are for night dining."
    return llm_response(text)
```

```
def validate_time(hour):
    return RESTAURANT_OPEN_TIME <= hour <= RESTAURANT_CLOSE_TIME
```

```
def get_branch_contact():
    phone = "+91 " + str(random.randint(7000000000, 9999999999))
    return f"粉色电话 If you are late, please inform the branch: {phone}"
```

```
def generate_booking_id():
    return str(uuid.uuid4())[:8].upper()
```

```
def booking_summary(name, table_type, date, time):
    booking_id = generate_booking_id()

    bookings[booking_id] = {
        "name": name,
        "table_type": table_type,
        "date": date,
        "time": time
    }
```

```

text = (
    f"🎉 Booking Confirmed!\n"
    f"-----\n"
    f">ID Booking ID : {booking_id}\n"
    f"👤 Name : {name}\n"
    f"🍽 Table Type : {table_type.capitalize()} ({TABLE_TYPES[table_t]}\n"
    f"📅 Date : {date}\n"
    f"🕒 Time : {time}:00 PM\n"
    f"💰 Reservation : {RESERVATION_FEE} INR (Currently {CURRENT_OFFER})\n"
    f"⚠️ Cancellation is NON-REFUNDABLE\n\n"
    f"{get_branch_contact()}"
)

return llm_response(text)

```

```

def cancel_booking(booking_id):
    if booking_id in bookings:
        booking = bookings.pop(booking_id)

        text = (
            f"🔴 Booking Cancelled Successfully\n"
            f"-----\n"
            f">ID Booking ID : {booking_id}\n"
            f"👤 Name : {booking['name']}\n"
            f"📅 Date : {booking['date']}\n"
            f"🕒 Time : {booking['time']}:00 PM\n"
            f"💰 Reservation fee is NON-REFUNDABLE\n"
            f"🙏 Thank you for informing us."
        )
        return llm_response(text)
    else:
        return llm_response("🔴 Invalid Booking ID. Please check and try again")

```

```

print(welcome_message())
print(show_table_options())
print("\nOptions: book | cancel\n")

action = input("What would you like to do? ").lower().strip()

# ----- BOOKING FLOW -----
if action == "book":
    table_choice = input("Select table type (single / couple / family) : ")

    if table_choice not in TABLE_TYPES:
        print(llm_response("🔴 Invalid table type selected."))
    else:
        booking_date = input("Enter booking date (YYYY-MM-DD): ")
        print(check_date_availability(booking_date))

        booking_time = int(input("Enter booking time (5 PM to 11 PM): "))

        if not validate_time(booking_time):

```

```

        print(llm_response("X Restaurant is open only from
else:
    booking_name = input("Enter booking name: ").strip()
    print(booking_summary(
        booking_name,
        table_choice,
        booking_date,
        booking_time
    ))
# ----- CANCELLATION FLOW -----
elif action == "cancel":
    booking_id = input("Enter Booking ID to cancel: ").upper().strip()
    print(cancel_booking(booking_id))

else:
    print(llm_response("X Invalid option. Please restart and c

```

Welcome to our Restaurant!
 I'm your table booking chatbot.
I can help you with bookings, availability, and cancellations.
Available Table Types:

- 1 Single Table - 1 Seat
- 2 Couple Table - 2 Seats
- 3 Family Table - 4 to 8 Seats

Options: book | cancel

What would you like to do? book
Select table type (single / couple / family): single
Enter booking date (YYYY-MM-DD): 2026-02-02
 Tables are available on 2026-02-02. Bookings are for night dining only.
Enter booking time (5 PM to 11 PM): 20
Enter booking name: Tejoduuu
 Booking Confirmed!

ID Booking ID : 9157EBF2
👤 Name : Tejoduuu
🪑 Table Type : Single (1 Seat)
📅 Date : 2026-02-02
🕒 Time : 20:00 PM
💰 Reservation : 100 INR (Currently FREE)
⚠️ Cancellation is NON-REFUNDABLE

If you are late, please inform the branch: +91 8649867593

