

```
# database.py
```

```
import sqlite3
```

```
from datetime import datetime
```

```
# Establish a connection to the SQLite database and create tables if they don't exist
```

```
class DatabaseManager:
```

```
    def __init__(self):
```

```
        self.conn = sqlite3.connect("restaurant.db")
```

```
        self.create_tables()
```

```
    def create_tables(self):
```

```
        # Create the necessary tables for the restaurant management system
```

```
        cursor = self.conn.cursor()
```

```
        cursor.execute("""
```

```
            CREATE TABLE IF NOT EXISTS foods (
```

```
                id INTEGER PRIMARY KEY AUTOINCREMENT,
```

```
                name TEXT NOT NULL,
```

```
                price REAL NOT NULL
```

```
            )
```

```
        """)
```

```
        cursor.execute("""
```

```
            CREATE TABLE IF NOT EXISTS drinks (
```

```
                id INTEGER PRIMARY KEY AUTOINCREMENT,
```

```
        name TEXT NOT NULL,  
        price REAL NOT NULL  
    )  
""")
```

```
cursor.execute("""
```

```
    CREATE TABLE IF NOT EXISTS orders (  
        id INTEGER PRIMARY KEY AUTOINCREMENT,  
        food_id INTEGER,  
        drink_id INTEGER,  
        food_qty INTEGER,  
        drink_qty INTEGER,  
        total_price REAL,  
        order_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
        FOREIGN KEY (food_id) REFERENCES foods(id),  
        FOREIGN KEY (drink_id) REFERENCES drinks(id)  
    )  
""")
```

```
cursor.execute("""
```

```
    CREATE TABLE IF NOT EXISTS customers (  
        id INTEGER PRIMARY KEY AUTOINCREMENT,  
        name TEXT NOT NULL,  
        contact TEXT  
    )
```

```
""")
```

```
self.conn.commit()
```

```
class Food:
```

```
    def __init__(self):
```

```
        self.db = DatabaseManager()
```

```
    def add_item(self, name, price):
```

```
        with self.db.conn:
```

```
            self.db.conn.execute("INSERT INTO foods (name, price) VALUES (?, ?)", (name, price))
```

```
    def list_items(self):
```

```
        with self.db.conn:
```

```
            return self.db.conn.execute("SELECT * FROM foods").fetchall()
```

```
class Drink:
```

```
    def __init__(self):
```

```
        self.db = DatabaseManager()
```

```
    def add_item(self, name, price):
```

```
        with self.db.conn:
```

```
            self.db.conn.execute("INSERT INTO drinks (name, price) VALUES (?, ?)", (name, price))
```

```
def list_items(self):  
    with self.db.conn:  
        return self.db.conn.execute("SELECT * FROM drinks").fetchall()
```

```
class OrderManager:
```

```
    def __init__(self):  
        self.db = DatabaseManager()
```

```
    def place_order(self, food_id, drink_id, food_qty, drink_qty, total_price):
```

```
        with self.db.conn:  
            self.db.conn.execute("""  
                INSERT INTO orders (food_id, drink_id, food_qty, drink_qty, total_price)  
                VALUES (?, ?, ?, ?, ?)  
            """, (food_id, drink_id, food_qty, drink_qty, total_price))
```

```
    def list_orders(self):
```

```
        with self.db.conn:  
            return self.db.conn.execute("SELECT * FROM orders ORDER BY order_date  
DESC").fetchall()
```

```
class CustomerManager:
```

```
    def __init__(self):
```

```
self.db = DatabaseManager()
```

```
def add_customer(self, name, contact):
```

```
    with self.db.conn:
```

```
        self.db.conn.execute("INSERT INTO customers (name, contact) VALUES (?, ?)", (name,  
contact))
```

```
def list_customers(self):
```

```
    with self.db.conn:
```

```
        return self.db.conn.execute("SELECT * FROM customers").fetchall()
```