

Hotel Database

Required Input

The hotel Database is filled with the following Data:

- floors
- rooms
- customers
- bookings
- room service usage
- restaurant orders
- types of facilities
- price bands

Required Output

The hotel Database provides the following Data:

- past, future and current bookings
- available rooms
- list of customers
- customer bill based on room, services and extras

```

    erDiagram
        room ||--o{ floor : "has"
        room ||--o{ price_band : "has"
        room ||--o{ room_configuration : "has"
        room ||--o{ facility : "has"
        floor ||--o{ price : "has"
        service ||--o{ service_bill : "has"
        service ||--o{ price : "has"
        service_bill ||--o{ restaurant_item : "has"
        price_band ||--o{ price : "has"
        price ||--o{ booking_period : "has"
        price ||--o{ restaurant_item : "has"
        room_configuration ||--o{ booking_period : "has"
        room_configuration ||--o{ facility : "has"
        facility ||--o{ booking_period : "has"
        party_composition ||--o{ party : "has"
        party ||--o{ customer : "has"
        party ||--o{ person : "has"
        party ||--o{ restaurant_order : "has"
        region ||--o{ person : "has"
        region ||--o{ contact : "has"
        person ||--o{ contact : "has"
        person ||--o{ company : "has"
        person ||--o{ customer : "has"
        person ||--o{ restaurant_order : "has"
        contact ||--o{ company : "has"
        contact ||--o{ customer : "has"
        contact ||--o{ restaurant_order : "has"
        company ||--o{ customer : "has"
        company ||--o{ restaurant_order : "has"
        customer ||--o{ restaurant_order : "has"
        restaurant_order ||--o{ restaurant_order_composition : "has"
        restaurant_order_composition ||--o{ restaurant_item : "has"
  
```

The diagram illustrates the database schema for a restaurant, featuring the following tables and their attributes:

- room**: id INT(11), area DECIMAL(10,0), floor_id INT(11) (FK), price_band_id INT(11) (FK)
- floor**: id INT(11), level INT(11), name VARCHAR(255), price_id INT(11) (FK)
- service**: id INT(11), name VARCHAR(255), price_id INT(11) (FK)
- service_bill**: id INT(11), service_id INT(11) (FK), booking_id INT(11) (FK)
- price_band**: id INT(11), name VARCHAR(255), booking_price_id INT(11) (FK), person_price_id INT(11) (FK), area_price_id INT(11) (FK)
- price**: id INT(11), name VARCHAR(255), booking DECIMAL(10,0), daily DECIMAL(10,0)
- booking_period**: id INT(11), start_date DATETIME, end_date DATETIME, room_configuration_id INT(11) (FK), party_id INT(11) (FK)
- restaurant_item**: id INT(11), name VARCHAR(255), type VARCHAR(255), price_id INT(11) (FK)
- room_configuration**: id INT(11), quantity INT(11), start_time DATETIME, end_time DATETIME, facility_id INT(11) (FK), room_id INT(11) (FK)
- facility**: id INT(11), name VARCHAR(255), price_id INT(11) (FK)
- party_composition**: id INT(11), person_id INT(11) (FK), party_id INT(11) (FK)
- party**: id INT(11), name VARCHAR(255), customer_id INT(11) (FK)
- region**: id INT(11), city VARCHAR(255), state VARCHAR(255), zip VARCHAR(255), country VARCHAR(255)
- person**: id INT(11), first_name VARCHAR(255), last_name VARCHAR(255), birth_day DATETIME, contact_id INT(11) (FK)
- contact**: id INT(11), address VARCHAR(255), telephone VARCHAR(255), email VARCHAR(255), region_id INT(11) (FK)
- company**: id INT(11), name VARCHAR(255), contact_id INT(11) (FK)
- customer**: id INT(11), person_id INT(11) (FK), company_id INT(11) (FK)
- restaurant_order**: id INT(11), type VARCHAR(255), time DATETIME, booking_period_id INT(11) (FK), customer_id INT(11) (FK)
- restaurant_order_composition**: id INT(11), restaurant_order_id INT(11) (FK), restaurant_item_id INT(11) (FK)

Relationships are indicated by dashed lines with crow's foot notation, showing one-to-many, one-to-one, and many-to-many connections between the tables.