Database Keys and Integrity Constraints

In this Hotel Management System, **Keys** are the fundamental tools used to ensure data accuracy, prevent duplicate records, and establish reliable links between different business entities (Guests, Rooms, and Payments).

1. Primary Keys (PK)

The Primary Key is a unique identifier for every record in a table. In this system, we use **Surrogate Keys** (unique alphanumeric strings) to ensure that even if two guests have the same name, the system can distinguish between them.

Table	Primary Key	Purpose
Customer	`customer_id`	Ensures every guest has a unique profile.
Room	`room_num`	Unique identifier for physical rooms.
Reservation	`res_id`	Distinguishes between different stays by the same guest.
Invoice	`invoice_id`	Unique reference for a specific billing event.
Services	`service_id`	Identifies specific products/amenities in the catalog.

2. Foreign Keys (FK)

Foreign Keys create the "Relational" aspect of the database. They act as pointers that link a record in one table to a record in another.

Key Operational Links:

- `customer_id` **(in Reservation table)**: This FK links a specific stay to a specific guest. It ensures that the hotel knows exactly who is staying in which room.
- `res_id` **(in Room table)**: This FK is used to track "State." If a room has a `res_id` assigned, it is "Occupied." If the field is NULL, the room is "Available."
- `invoice_id` **(in Line table)**: This links individual service charges (like a $20 breakfast) to the master bill.

3. Composite Keys & Associative Entities

In the **Line** table (which handles the Many-to-Many relationship between Invoices and Services), we utilize a **Composite Logic**.

Instead of a single ID, the uniqueness of a charge is often defined by the combination of: `invoice_id + service_id`

This ensures that a single invoice can list multiple different services, but prevents the exact same service from being double-recorded as a duplicate entry without intent.

4. Referential Integrity Rules

To prevent "Orphaned Records" (data that has lost its parent), the system enforces strict integrity rules:

1. **RESTRICT DELETE**: You cannot delete a `Customer` record if they have an active `Reservation`. The system forces you to resolve the reservation first.
2. **MANDATORY PARENT**: An `Invoice` cannot be created unless it is linked to a valid `res_id`. This prevents the creation of "ghost bills" that don't belong to any guest stay.
3. **VALIDATION**: The `payment_method` in the `Transactions` table is often constrained to specific values (Cash, Credit, Debit) to ensure data cleanlines for the Revenue Reports.

5. Why These Keys Matter

- **Data Consistency**: If a guest changes their email in the `Customer` table, the `Reservation` table (which only holds the `customer_id`) automatically reflects that change without needing manual updates.
- **Search Optimization**: Primary Keys are automatically indexed by the database engine, making searches for specific invoices or rooms near-instantaneous.
- **Accounting Accuracy**: By linking `Satisfaction` to `transaction_id`, we ensure that feedback is only collected for completed, paid transactions, preventing fraudulent reviews.