# Practical 8

**Prepare a data dictionary for Software project definition.**

**Data Dictionary**

The data dictionary consists of two words, data, which represents data collected from several sources, and dictionary, which represents where this data is available. The data dictionary is an important part of the relational database because it provides additional information about the relationship between several tables in the database. A data dictionary in a DBMS helps users manage data in an orderly and orderly manner, thereby preventing data redundancy.

**- Importance of Data Dictionary**

* **Standardization:** Provides a consistent definition for each data element, ensuring that all team members use the same terminology and formats.
* **Documentation:** Acts as a reference guide during development, maintenance, and future enhancements of the system.
* **Data Integrity:** Helps enforce constraints (like primary keys and unique values) and validates data types, ensuring the quality of the data stored.
* **Communication:** Bridges the gap between technical and non-technical stakeholders by providing clear and concise definitions for each data field.

**- Components of a Data Dictionary**

* **Field Name:** The unique identifier for the data element.
* **Data Type:** Specifies the type of data (e.g., VARCHAR, INTEGER, DATETIME, DECIMAL).
* **Size/Format:** Indicates the maximum size or specific format requirements (e.g., 50 characters, (8,2) for decimals).
* **Description:** A brief explanation of what the data element represents.
* **Constraints:** Details any rules such as primary key (PK), foreign key (FK), not null, unique, or enum values.

**Data Dictionary for Movie Booking Management System**

The Movie Booking Management System manages movie ticket bookings, user profiles, payments, reviews, and administrative tasks. The following entities and their attributes represent the stored data elements in the system, along with their properties and relationships.

1. **Entity: User**

**Description**: Represents a user of the system, who can be a customer booking tickets or an administrator managing the system.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute Name | Data Type | Size | Constraints | Description |
| UserID | Integer |  | Primary Key | Unique identifier for each user |
| Name | String | 100 | Not Null | Full name of the user |
| Email | String | 100 | Unique, Not Null | Email address for login and communication |
| Password | String | 255 | Not Null | Encrypted password for authentication |
| Role | String | 10 | Not Null | Role of the user (Customer or Admin) |

1. **Entity: Profile**

**Description**: Stores additional personal information for each user, linked to the User entity.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute Name | Data Type | Size | Constraints | Description |
| ProfileID | Integer |  | Primary Key | Unique identifier for each profile |
| UserID | Integer |  | Foreign Key (User.UserID) | References the User entity |
| contact\_no | String | 15 |  | Phone number of the user |
| profile\_pic | String | 255 |  | URL or path to the user's profile picture |

1. **Entity: Movie**

**Description**: Represents a movie available for booking in the system.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute Name | Data Type | Size | Constraints | Description |
| MovieID | Integer |  | Primary Key | Unique identifier for each movie |
| Title | String | 255 | Not Null | Title of the movie |
| Genre | String | 50 |  | Genre of the movie (e.g., Action, Comedy) |
| Language | String | 50 |  | Language of the movie |
| Duration | Integer |  |  | Duration of the movie in minutes |

1. **Entity: Theatre**

**Description**: Represents a physical theatre where movies are screened.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute Name | Data Type | Size | Constraints | Description |
| TheatreID | Integer |  | Primary Key | Unique identifier for each theatre |
| Name | String | 100 | Not Null | Name of the theatre |
| Location | String | 255 | Not Null | Address or location of the theatre |
| Capacity | Integer |  |  | Total seating capacity of the theatre |

1. **Entity: Showtime**

**Description**: Represents the scheduled screening times for a movie in a specific theatre.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute Name | Data Type | Size | Constraints | Description |
| ShowtimeID | Integer |  | Primary Key | Unique identifier for each showtime |
| MovieID | Integer |  | Foreign Key (Movie.MovieID) | References the Movie entity |
| TheatreID | Integer |  | Foreign Key (Theatre.TheatreID) | References the Theatre entity |
| StartTime | DateTime |  | Not Null | Start time of the movie show |
| EndTime | DateTime |  | Not Null | End time of the movie show |

1. **Entity: Seat**

**Description**: Represents individual seats available for a specific showtime.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute Name | Data Type | Size | Constraints | Description |
| SeatID | Integer |  | Primary Key | Unique identifier for each seat |
| ShowtimeID | Integer |  | Foreign Key (Showtime.ShowtimeID) | References the Showtime entity |
| SeatNumber | String | 10 | Not Null | Seat number (e.g., A1, B2) |
| Status | String | 20 | Not Null | Status of the seat (e.g., Available, Booked) |

1. **Entity: Booking**

**Description**: Represents a reservation made by a user for a specific showtime.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute Name | Data Type | Size | Constraints | Description |
| BookingID | Integer |  | Primary Key | Unique identifier for each booking |
| UserID | Integer |  | Foreign Key (User.UserID) | References the User entity |
| ShowtimeID | Integer |  | Foreign Key (Showtime.ShowtimeID) | References the Showtime entity |
| PaymentStatus | String | 20 | Not Null | Status of the payment (e.g., Pending, Completed) |

1. **Entity: BookedSeat**

**Description**: Junction table linking bookings to specific seats, allowing multiple seats per booking.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute Name | Data Type | Size | Constraints | Description |
| BookedSeatID | Integer |  | Primary Key | Unique identifier for each booked seat |
| BookingID | Integer |  | Foreign Key (Booking.BookingID) | References the Booking entity |
| SeatID | Integer |  | Foreign Key (Seat.SeatID) | References the Seat entity |

1. **Entity: Payment**

**Description**: Represents financial transactions associated with bookings.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute Name | Data Type | Size | Constraints | Description |
| PaymentID | Integer |  | Primary Key | Unique identifier for each payment |
| BookingID | Integer |  | Foreign Key (Booking.BookingID) | References the Booking entity |
| Amount | Decimal | 10,2 | Not Null | Total amount paid for the booking |
| PaymentMethod | String | 50 | Not Null | Method of payment (e.g., Credit Card, Cash) |
| Status | String | 20 | Not Null | Status of the payment (e.g., Success, Failed) |

1. **Entity: Reviews**

**Description**: Stores user-generated feedback and ratings for movies.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute Name | Data Type | Size | Constraints | Description |
| ReviewID | Integer |  | Primary Key | Unique identifier for each review |
| MovieID | Integer |  | Foreign Key (Movie.MovieID) | References the Movie entity |
| UserID | Integer |  | Foreign Key (User.UserID) | References the User entity |
| ratings | Integer |  |  | Rating given by the user (e.g., 1 to 5) |
| description | Text |  |  | Textual review content |
| type | String | 20 |  | Type of review (e.g., User, Critic) |