Data Types

This lesson talks about the various data types available in MySQL.

Data Types

In the previous lesson we created a database and now we'll learn how to create tables within a database. But before we create a table, we'll examine the concept of a table. A database is primarily made up of tables and as explained before, a table consists of columns. Each column is an attribute of an entity that is defined by the table. For instance, the first table we'll create will consist of actors. What information do we want to hold about actors? We can be interested in the first and last names, age, date of birth, marital status, gender, etc., of an actor, all attributes that define some aspect of an actor.

Our table, **Actors**, will consist of columns that represent the age, gender, date of birth, etc., for an actor. However, note that a name will consist of alphabets, age will be a number, and date of birth will be a date. MySQL provides various data types to represent the type of columns in a table. The analogy is similar to a strongly-typed language such as Java or C# which allows a developer to define a type for each variable. This strong typing and sizing make the data in relational databases structured. The major categories of various data-types are as follows:

- **Numeric** e.g., INT, BIGINT, TINYINT, DECIMAL, etc.
- **Date and Time** e.g., DATE, TIME, TIMESTAMP, YEAR, etc.
- **String** e.g., VARCHAR, CHAR, ENUM, SET, BLOB, etc.
- **JSON** e.g., JSON

- **Spatial Data** represents the location, size, and shape of an object on planet Earth such as a building, lake, mountain, or township. MySQL also supports spatial data types, e.g., GEOMETRY, POINT, etc.

We won't exhaustively discuss all the data types offered by MySQL, as ample online documentation exists. We will mention the data types that we use in creating tables for our example **MovieIndustry** database. We can create our **Actors** table with the following columns and associated datatypes.

Column Name	Column Type
FirstName	VARCHAR(20)
SecondName	VARCHAR(20)
DoB	DATE
Gender	ENUM('Male','Female','Transgender')
MaritalStatus	ENUM('Married','Divorced','Single')
NetWorthInMillions	DECIMAL

The first column, **FirstName**, is represented by data type **VARCHAR(20)**. The data type means that the first name of an actor can be stored as

twenty or fewer characters. The number twenty within brackets specifies the maximum length of string we intend to store. The absolute maximum

for **VARCHAR** is 65,535 characters. Instead of **VARCHAR** we could have used **CHAR** but the difference between **CHAR** and **VARCHAR** is that the former is fixed length whereas the latter is variable length. If we specify **CHAR(20)**, it will imply the name is always stored as twenty characters with spaces even if the name consists of only four characters.

In the next lesson, we'll create the **Actors** table.