HSBC Technology Graduate Training

Databases: MySQL & SQL

Day 3 (Afternoon)

Wednesday 28 October 2020 | 2pm

Contents

Databases

Databases

- **Database Management System (DBMS) is software that manages databases.**
- **A record is a collection of information.**
- **A record contains fields.**
- **The following is a record:**

User ID	47893475
Email	hello@john.com
Age	43

- **The record above contains 3 fields.**
- **A collection of records is stored in a table.**

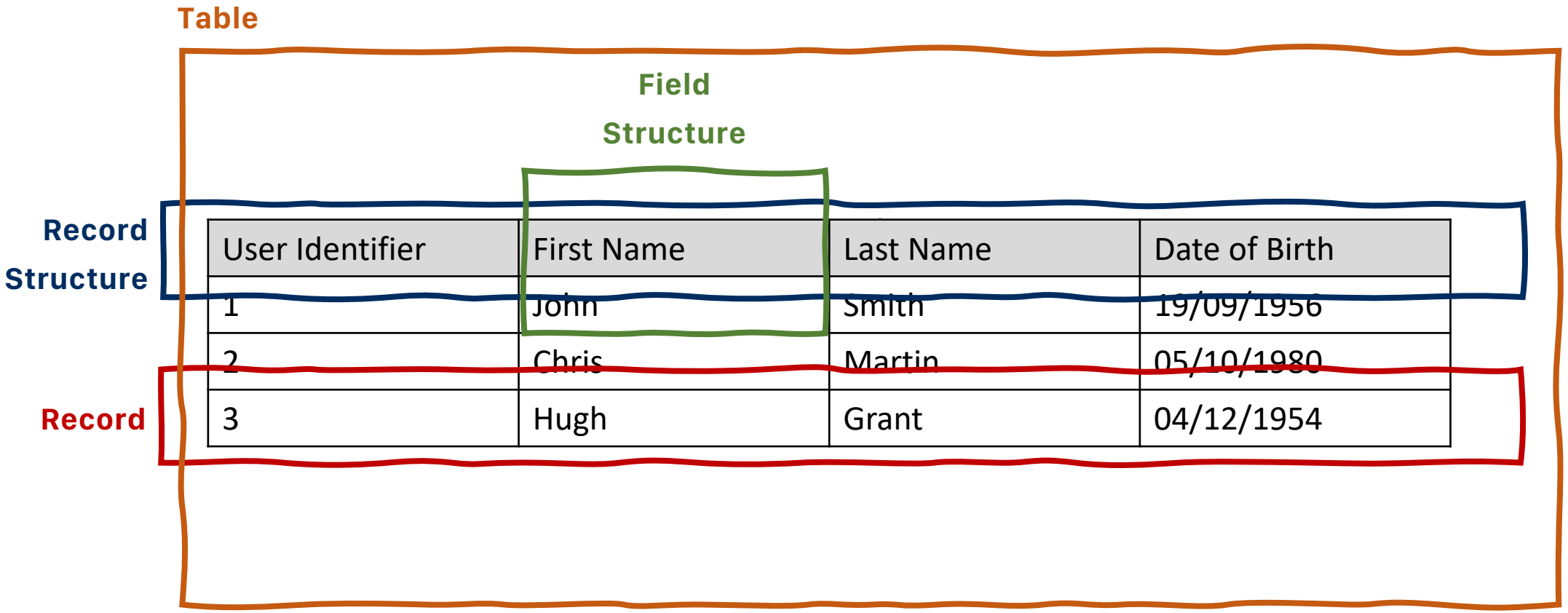


Table
Field
Structure
Record
Structure
Record
User Identifier
First Name
Last Name
Date of Birth
1
John
Smith
19/09/1956
2
Chris
Martin
05/10/1980
3
Hugh
Grant
04/12/1954

Database

User Identifier	First Name	Last Name	Date of Birth
1	John	Smith	19/09/1956
2	Chris	Martin	05/10/1980
3	Hugh	Grant	04/12/1954

User Identifier	First Name	Last Name	Date of Birth
1	John	Smith	19/09/1956
2	Chris	Martin	05/10/1980
3	Hugh	Grant	04/12/1954

User Identifier	First Name	Last Name	Date of Birth
1	John	Smith	19/09/1956
2	Chris	Martin	05/10/1980
3	Hugh	Grant	04/12/1954

RECORD STRUCTURES

- **A record structure is created through the creation of field structures.**
- **For instance, if creating a table to store Student information, we can create the following field structures:**

Name of field	Data type	Capacity
name	varchar	255
age	int	
subject	varchar	255

SQL

SQL

- **SQL – Structured Query Language**
- **Collection of commands to communicate with a Database Management System**

SQL Command: CREATE DATABASE

SQL Command: CREATE DATABASE

- The CREATE DATABASE command is used to create a database.
- The example below creates a database named HSBC.

SQL Command: SHOW DATABASES

SQL Command: SHOW DATABASES

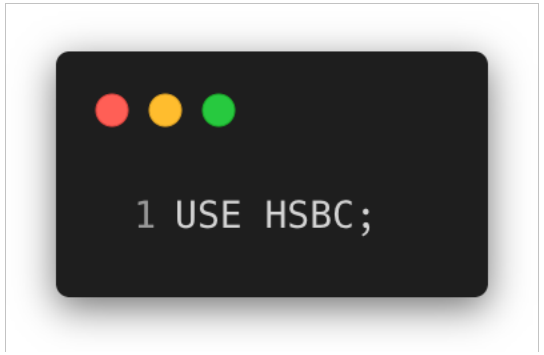
- **The SHOW DATABASES command shows a list of databases.**

```
mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| HSBC               |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)
```

SQL Command: USE

SQL Command: USE

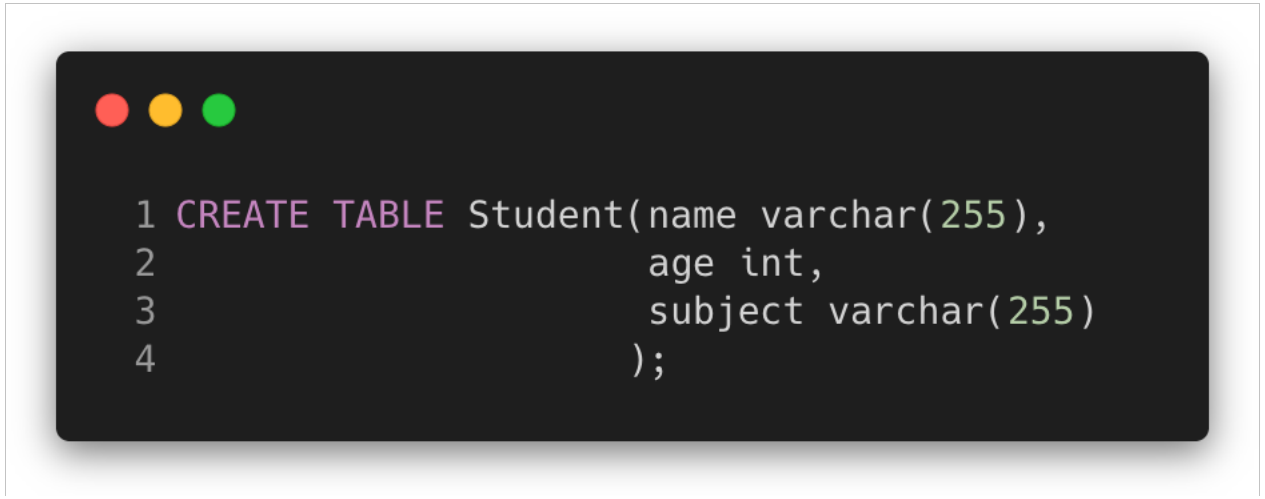
- The USE command is used to use a database.
- The example below shows us using the database HSBC.



SQL Command: CREATE TABLE

SQL Command: CREATE TABLE

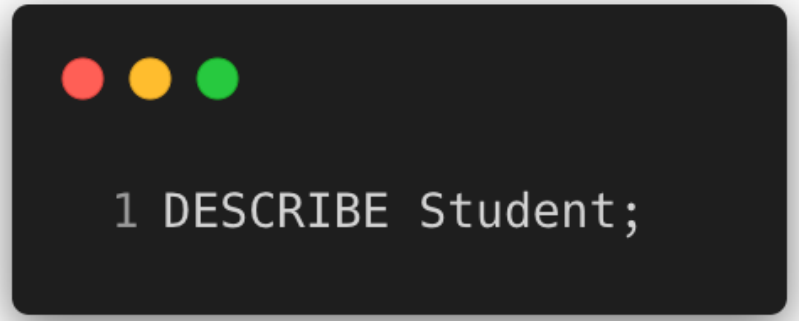
- We use the CREATE TABLE command to create a table in a database.
- For example, if we want to create a table named Student, we use the following command.
- Part of this command requires us to specify the record structure of the table.
- In other words, we must specify what columns will be added to the table, including its data type and capacity (if applicable).



SQL Command: DESCRIBE

SQL Command: DESCRIBE

- **We can use the DESCRIBE command to show to structure of a given table**




```
mysql> DESCRIBE Student;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| name    | varchar(255) | YES  |     | NULL    |       |
| age     | int         | YES  |     | NULL    |       |
| subject | varchar(255) | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

Output

SQL Command: INSERT INTO

SQL Command: INSERT INTO

- We use the INSERT INTO command to insert records into a table.
- Non-numeric values must be enclosed in single quotes.

```
1 INSERT INTO Student VALUES ('David Attenborough', 80, 'Geology');
```

SQL Command: SELECT

SQL Command: SELECT

- We use the SELECT command to retrieve and view records from a table.
- * means show all columns.
- The command below means “Select all records from the table Student”.

```
1 SELECT * FROM Student;
```

```
mysql> SELECT * FROM Student;
+-------+------+----------+
| name  | age  | subject  |
+-------+------+----------+
| John  |   23 | Comp Sci |
| James |   32 | History  |
+-------+------+----------+
2 rows in set (0.00 sec)
```

Output

SQL Command: SELECT

- We can specify which columns we want to see by replacing * with a list of columns.
- For instance, we want to see name and age from the Student table.

```
1 SELECT name, age FROM Student;
```

```
mysql> SELECT name, age FROM Student;
+-------+------+
| name  | age  |
+-------+------+
| John  |   23 |
| James |   32 |
+-------+------+
2 rows in set (0.00 sec)
```

Output

SQL Command: SELECT

- We can apply filters to the records returned from a SELECT query.
- For instance, if we only wanted to see records in the Student table with an age of greater than 25, we can use the WHERE keyword to filter the query.

```
1 SELECT * FROM Student WHERE age > 25;
```

```
mysql> SELECT * FROM Student WHERE age > 25;
+-------+------+---------+
| name  | age  | subject |
+-------+------+---------+
| James |   32 | History |
+-------+------+---------+
1 row in set (0.00 sec)
```

Output

SQL Command: SELECT

- **If we want to filter on a column with a string value, we need to use single quotes.**

```
1 SELECT * FROM Student WHERE subject='Comp Sci';
```

```
mysql> SELECT * FROM Student WHERE subject='Comp Sci';
+------+------+----------+
| name | age  | subject  |
+------+------+----------+
| John |   23 | Comp Sci |
+------+------+----------+
1 row in set (0.00 sec)
```

Output

SQL Command: UPDATE

SQL Command: UPDATE

- The UPDATE Command is used to edit records in a table.
- For instance, say we want to change the subject of John to “Physics”.

```
1 UPDATE Student SET subject='Physics' WHERE name='John';
```

SQL Command: UPDATE

- **We can update more than one column at a time.**

```
1 UPDATE Student SET subject='Physics', age=43 WHERE name='John';
```

SQL Command: ORDER BY

SQL Command: ORDER BY

- We can ask SQL to return a list of records in a specific order.
- The order can be dependent on a column.
- If the column is numeric, we can order in ASC (ascending) or DESC (descending) order.
- If the column is a string, we can order in ASC or DESC alphabetical order.
- If we don't specify a direction, the default direction is ASC (Ascending order).

```
1 SELECT * FROM Student ORDER BY age;
```

```
mysql> SELECT * FROM Student ORDER BY age;
+-------+-----+---------+
| name  | age | subject |
+-------+-----+---------+
| John  |  23 | Physics |
| James |  32 | History |
+-------+-----+---------+
2 rows in set (0.00 sec)
```

```
1 SELECT * FROM Student ORDER BY age DESC;
```

```
mysql> SELECT * FROM Student ORDER BY age DESC;
+-------+-----+---------+
| name  | age | subject |
+-------+-----+---------+
| James |  32 | History |
| John  |  23 | Physics |
+-------+-----+---------+
2 rows in set (0.00 sec)
```

SQL Command: DELETE

SQL Command: DELETE

- **DELETE command is used to delete records from the table.**
- **If no WHERE clause is specified, DELETE will remove all records in the table.**

```
1 DELETE FROM Student WHERE name='James';
```