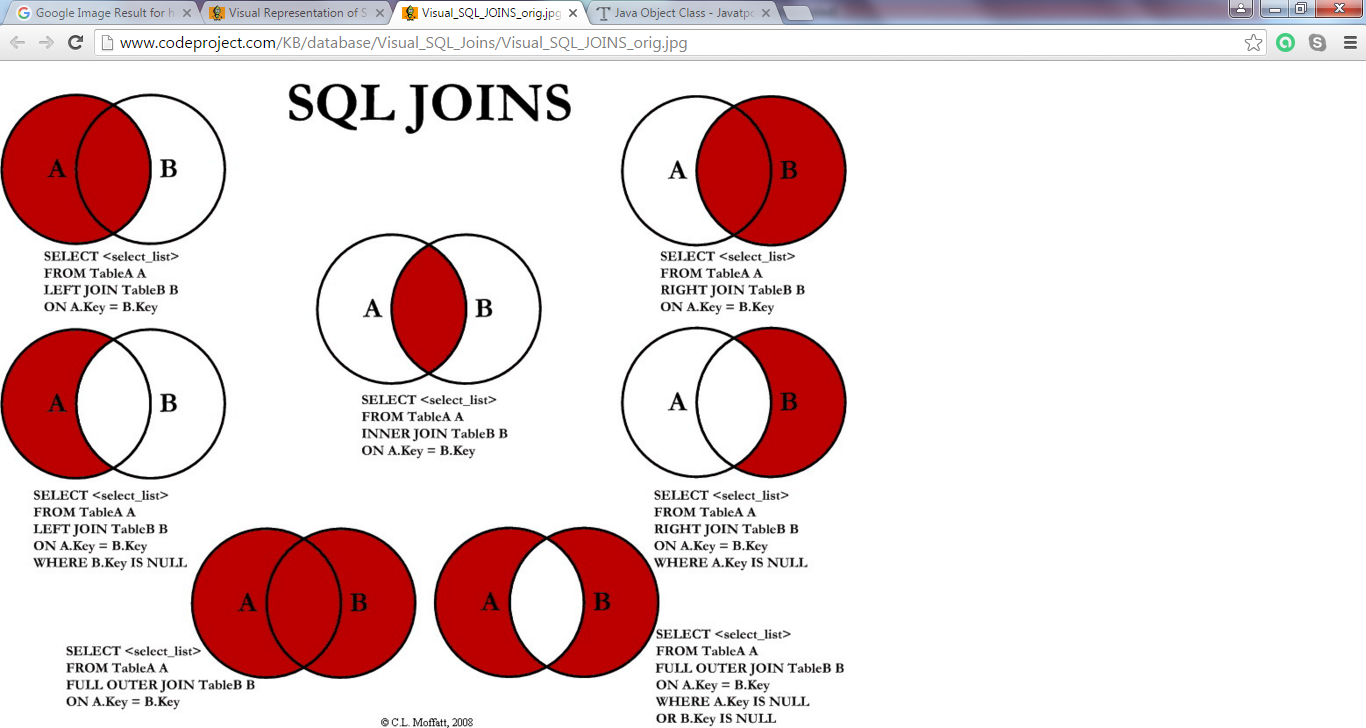
Reference: http://www.javatpoint.com/sql-interview-questions

|  |  |  |
| --- | --- | --- |
| **No.** | **DELETE** | **TRUNCATE** |
| 1) | DELETE is a **DML command**. | TRUNCATE is a **DDL command**. |
| 2) | We **can use WHERE** clause in DELETE command. | We **cannot use WHERE** clause with TRUNCATE |
| 3) | DELETE statement is used **to delete a row** from a table | TRUNCATE statement is used **to remove all the rows** from a table. |
| 4) | DELETE is **slower** than TRUNCATE statement. | TRUNCATE statement is **faster** than DELETE statement. |
| 5) | You **can rollback** data after using DELETE statement. | It is **not possible to rollback** after using TRUNCATE statement. |

|  |  |  |
| --- | --- | --- |
|  | **PRIMARY KEY** | **UNIQUE KEY** |
| **NULL** | It doesn’t allow Null values. Because of this we refer PRIMARY KEY = UNIQUE KEY + Not Null CONSTRAINT | Allows Null value. But only one Null value. |
| **INDEX** | By default it adds a clustered index | By default it adds a UNIQUE non-clustered index |
| **LIMIT** | A table can have only one PRIMARY KEY Column[s] | A table can have more than one UNIQUE Key Column[s] |
| **CREATE SYNTAX** | Below is the sample example for defining a single column as a PRIMARY KEY column while creating a table:CREATE TABLE dbo.Customer ( Id INT NOT NULL PRIMARY KEY, FirstName VARCHAR(100), LastName VARCHAR(100), City VARCHAR(50) )  Below is the Sample example for defining multiple columns as PRIMARY KEY. It also shows how we can give name for the PRIMARY KEY:  CREATE TABLE dbo.Customer ( Id INT NOT NULL, FirstName VARCHAR(100) NOT NULL, LastName VARCHAR(100), City VARCHAR(50), CONSTRAINT PK\_CUSTOMER PRIMARY KEY(Id,FirstName) ) | Below is the sample example for defining a single column as a UNIQUE KEY column while creating a table:CREATE TABLE dbo.Customer ( Id INT NOT NULL UNIQUE, FirstName VARCHAR(100), LastName VARCHAR(100), City VARCHAR(50) )  Below is the Sample example for defining multiple columns as UNIQUE KEY. It also shows how we can give name for the UNIQUE KEY:  CREATE TABLE dbo.Customer ( Id INT NOT NULL, FirstName VARCHAR(100) NOT NULL, LastName VARCHAR(100), City VARCHAR(50), CONSTRAINT UK\_CUSTOMER UNIQUE(Id,FirstName) ) |
| **ALTER SYNTAX** | Below is the Syntax for adding PRIMARY KEY CONSTRAINT on a column when the table is already created and doesn’t have any primary key:ALTER TABLE dbo.Customer ADD CONSTRAINT PK\_CUSTOMER PRIMARY KEY(Id) | Below is the Syntax for adding UNIQUE KEY CONSTRAINT on a column when the table is already created:ALTER TABLE dbo.Customer ADD CONSTRAINT UK\_CUSTOMERUNIQUE (Id) |
| **DROP SYNTAX** | Below is the Syntax for dropping a PRIMARY KEY:ALTER TABLE dbo.Customer DROP CONSTRAINT PK\_CUSTOMER | Below is the Syntax for dropping a UNIQUE KEY:ALTER TABLEdbo.Customer DROP CONSTRAINT UK\_CUSTOMER |



Different SQL JOINs

Before we continue with examples, we will list the types of the different SQL JOINs you can use:

* **INNER JOIN**: Returns all rows when there is at least one match in BOTH tables
* **LEFT JOIN**: Return all rows from the left table, and the matched rows from the right table
* **RIGHT JOIN**: Return all rows from the right table, and the matched rows from the left table
* **FULL JOIN**: Return all rows when there is a match in ONE of the tables