SQL Constraints

By Rahul Barve

SQL Constraints

- SQL constraints are used to specify rules for the data in a table.
- If there is any violation between the constraint and the data action, the action is aborted by the constraint.
- Constraints can be specified when the table is being created or after the table has been created.

SQL Constraints

- SQL constraints are categorized as:
  - `NOT NULL`
  - `UNIQUE`
  - `CHECK`
  - `DEFAULT`
  - `PRIMARY KEY`
  - `FOREIGN KEY`

NOT NULL

NOT NULL

- Restricts column from holding a NULL value.

NOT NULL

- Syntax (While creating table):

```
<column-name> <dimension>
[constraint <constraint-name>]
NOT NULL
```

NOT NULL

- Syntax (Using existing table):

```
alter table <table-name>
modify <column-name>
[constraint <constraint-name>]
NOT NULL
```

UNIQUE

By Rahul Barve

UNIQUE

- Restricts column from holding duplicate values.

UNIQUE

- Syntax (While creating table):

```
<column-name> <dimension>
[constraint <constraint-name>]
UNIQUE
```

UNIQUE

- Syntax (Using existing table):

```
alter table <table-name>
modify <column-name>
[constraint <constraint-name>]
UNIQUE
```

CHECK

CHECK

- Restricts column from holding a value that violates condition.

CHECK

- Syntax (While creating table):

```
<column-name> <dimension>
[constraint <constraint-name>]
CHECK (<CONDITION>)
```

CHECK

- Syntax (Using existing table):

```
alter table <table-name>
modify <column-name>
[constraint <constraint-name>]
CHECK (<CONDITION>)
```

DEFAULT

By Rahul Barve

DEFAULT

- Used to assign a default value to the column when not assigned explicitly.

DEFAULT

- Syntax (While creating table):

```
<column-name> <dimension>
DEFAULT <VALUE>
```

DEFAULT

- Syntax (Using existing table):

```
alter table <table-name>
modify <column-name>
DEFAULT <VALUE>
```

PRIMARY KEY

By Rahul Barve

PRIMARY KEY

- Restricts column from holding NULL and duplicate values.

PRIMARY KEY

- Syntax (While creating table):

```
<column-name> <dimension>
[constraint <constraint-name>]
PRIMARY KEY
```

PRIMARY KEY

- Syntax (While creating table)

```
<column-name> <dimension>,
<column-name> <dimension>,
[constraint <constraint-name>]
PRIMARY KEY (<column-name>)
```

PRIMARY KEY

- Syntax (Using existing table):

```
alter table <table-name>
modify <column-name>
[constraint <constraint-name>]
PRIMARY KEY
```

FOREIGN KEY

By Rahul Barve

FOREIGN KEY

- Used to build an association or relationship between the 2 tables.

FOREIGN KEY

- There are 3 types of relationships possible in database:
  - One-to-One
  - One-to-Many
  - Many-to-Many

FOREIGN KEY

- Syntax (While creating table)

```
<column-name> <dimension>,
<column-name> <dimension>,
[constraint <constraint-name>]
FOREIGN KEY (<child-table-column-name>)
REFERENCES
<parent-table-name>
(<parent-table-primary-column-name)
```

FOREIGN KEY

- Syntax (Using existing table)

```
alter table <table-name>
modify <column-name>
[constraint <constraint-name>]
references <parent-table-name>
(<parent-table-primary-column-
name>)
```

SQL Constraints

- It’s possible to remove the constraints.
- Syntax:

```
alter table <table-name>
drop constraint <constraint-name>
```

Database Objects

By Rahul Barve

Database Objects

- A database object in a relational database is a data structure used to either store or reference data.
- The most commonly used database object is the table.

Database Objects

- There are so many database objects available e.g.
  - Constraints
  - Sequences
  - Views
  - Synonyms
  - Indexes
  - Stored Procedures
  - Functions
  - Triggers

Sequence

By Rahul Barve

Sequence

- A sequence is a database object that generates a sequence of integers.
- Generally used especially in order to auto generate values for identity columns.

Sequence

- Syntax:

```
create sequence <sequence-name>
START WITH <VALUE>
MINVALUE <VALUE> [NOMINVALUE>
MAXVALUE <VALUE> [NOMAXVALUE>
INCREMENT BY <VALUE>
[CYCLE][NOCYCLE]
```

View

View

- In SQL, a view is a virtual table based upon the set of results of a query.
- A view contains rows and columns, just like a real table.

View

- The fields in a view are fields from one or more real tables in the database.
- Updates made to the view are reflected in the original table.

View

- Views can be created using the following syntax:

```
CREATE [OR REPLACE] VIEW <VIEW-NAME> AS
SELECT column1, column2,
FROM <TABLE-NAME>
[WHERE <condition>];
```

Lets Summarize

- What is RDBMS
- Why RDBMS
- Data Normalization
- Working with SQL Commands
- SQL Functions
- Predicates
- Constraints
- SQL Clauses
- Using Joins
- Using DB Objects