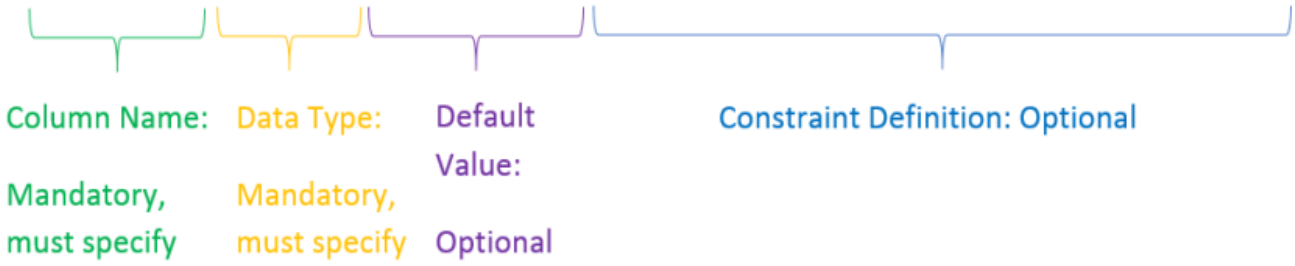
DDL Hands-on Demo Document

1. CREATE TABLE Statement

Oracle CREATE TABLE statement is used to create new tables in the database.

```
CREATE TABLE table_name
(column_name data_type DEFAULT value CONSTRAINT constraint_name constraint_type,
```



Data Types

Column Type	Description	Example
varchar2 (size**)**	String column. The value within the brackets indicates the maximum size of each field in the column (in characters)	varchar2(3) → 'ABC' varchar2(3) → 'AB'
NUMBER (p,s)	Numeric column. **P**recision – number of digits, **S**cale – how	number(5,2) → 476.29 number(5,2) → 6.29

	many of the digits are located after the number point	
DATE	Date format column	‘DD-MON-YYYY’ ’15-MAY-2015′

Oracle Default Value

A column can be given a default value using the DEFAULT keyword. The DEFAULT keyword provides a default value to a column when the Oracle INSERT INTO statement does not provide a specific value. The default value can be literal value, an expression, or a SQL Function, such as SYSDATE.

To define a Default value, use this syntax:

```
DEFAULT default_value
```

For example:

```
CREATE TABLE demo_tbl

(

salary number(8,2) DEFAULT 9500,

hire_date DATE DEFAULT '01-JAN-2011' ,

birthdate DATE DEFAULT SYSDATE

)
```

2. Creating Oracle Constraints

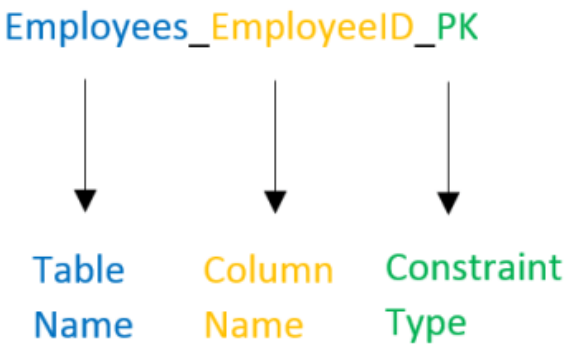
```
CONSTRAINT constraint_name constraint_type
```

- Constraint_type – the type of the constraint to be enforced on the column (for example, Unique or Not Null)
- Constraint_name – although not mandatory, it is always advisable to give the constraint a name, thereby allowing you to easily identify it.

The following naming convention is commonly used by many database developers:

<table name>_<column_name>_<constraint abbreviation>

For example:



Primary Key (PK)

In Oracle, the Primary Key constraint is a column (or a set of columns) that uniquely identifies each row in the table, this constraint enforces uniqueness and ensures that no column that is part of the Primary Key can hold a NULL value. Only one Primary Key can be created for each table.

The syntax used for defining a Primary Key Constraint is as follows:

```
column_name column_DataType [DEFAULT value] [CONSTRAINT constraint_name]
PRIMARY KEY,
```

For example:

```
CREATE TABLE products
(product_id number(3) CONSTRAINT products_id_pk PRIMARY KEY,
 product_name varchar2(25))
```

Please note – the square brackets in this demonstration (and in those that follow) indicate that what enclosed within them is optional, the square brackets are not part of the CREATE TABLE statement.

Not Null (NN)

In Oracle, the Not Null constraint ensures that the column contains no NULL values. The syntax used for defining a Not Null constraint is as follows:

```
column_name column_DataType [DEFAULT value] [CONSTRAINT constraint_name]
NOT NULL,
```

For example:

```
CREATE TABLE products
```

```
(product_id number(3)       CONSTRAINT products_id_pk PRIMARY KEY,
 product_name varchar2(25) CONSTRAINT products_name_nn NOT NULL)
```

This constraint can only be defined at the column level

UNIQUE (UQ)

In Oracle, the Unique constraint requires that every value in a column (or set of columns) be unique. The syntax used for defining a UNIQUE Constraint is as follows:

```
column_name column_DataType [DEFAULT value] [CONSTRAINT constraint_name]
UNIQUE,
```

For example:

```
CREATE TABLE products
(product_id number(3)                CONSTRAINT products_id_pk PRIMARY
KEY,
 product_name varchar2(25)           CONSTRAINT products_name_nn NOT
NULL,
 product_description varchar2(25)    CONSTRAINT products_description_uq
UNIQUE)
```

CHECK (CK)

In Oracle, the Check constraint defines a condition that each row must satisfy. The syntax used for defining a Check Constraint is as follows:

```
column_name column_DataType [DEFAULT value] [CONSTRAINT constraint_name]
CHECK (Condition),
```

- The condition written in the CHECK is quite similar in its structure to each of the conditions written in a WHERE sentence.
- The condition in the CHECK part must not include:
    - Values that are returned as a result of using SEQUENCES
    - Functions such as SYSDATE, ROWNUM
    - Subqueries

For Example:

```
CREATE TABLE products
(product_id number(3)                CONSTRAINT products_id_pk PRIMARY
KEY,
 product_name varchar2(25)           CONSTRAINT products_name_nn NOT NULL,
 product_description varchar2(25)    CONSTRAINT products_description_uq
UNIQUE,
 product_price       number(8,2)     CONSTRAINT products_productprice_ck
CHECK (product_price > 50))
```

FOREIGN KEY (FK)

In Oracle, the Foreign Key constraint designates a column (or a set of columns) as a Foreign Key and establishes a relationship between a Primary Key (or Unique) in different table (or in the same table) The syntax used for defining a Check Constraint is as follows:

```
column_name  … [CONSTRAINT constraint_name] REFERENCES
table_name(column_name) [ON DELETE CASCADE] [ON DELETE SET NULL]
```

Example:

The Parent Table

```
CREATE TABLE categories
(category_id number(3) CONSTRAINT categories_id_pk PRIMARY KEY ,
 category_name varchar2(25))
```

The Child Table

```
CREATE TABLE products
(product_id number(3)              CONSTRAINT products_id_pk PRIMARY KEY,
 product_name varchar2(25)         CONSTRAINT products_name_nn NOT NULL,
 product_description varchar2(25)  CONSTRAINT products_description_uq UNIQUE,
 product_price     number(8,2)     CONSTRAINT product_price_ck CHECK (product_price >
50),
```

```
 category_id number(3)          CONSTRAINT emp_categoryid_fk REFERENCES
categories(category_id))
```

Table Level Constraints

- Created after defining the various column.
- Can refer to more than one column (a constraint that comprises two columns together).
- Allows creating several constraints on the same column.
- It is not possible to create a NOT NULL constraint by using this method.

For example:

```
CREATE TABLE products
(product_id number(3),
 product_name varchar2(25)                    CONSTRAINT products_name_nn
CONSTRAINT NOT NULL,
 product_description varchar2(25),
 product_price     number(8,2) ,
 product_discount  number(8,2)                CONSTRAINT
products_discount_nn CONSTRAINT NOT NULL,
 category_id       number(3),
 CONSTRAINT products_id_pk                    PRIMARY KEY (product_id),
 CONSTRAINT products_description_name_uq      UNIQUE(product_name,
product_description),
 CONSTRAINT product_price_ck                  CHECK (product_price >
product_discount),
```

```
 CONSTRAINT emp_categoryid_fk                  FOREIGN KEY
(category_id)REFERENCES categories(category_id) )
```

3. Drop an Existing Table

The syntax used for deleting an existing table in Oracle is as follows:

```
DROP TABLE table_name
```

For example

```
DROP TABLE employees
```