LO4 and LO5 – Table Creation with DDL

DDL – Data Definition Language (LO4 and LO5)

1. We are changing the structure of the database.

DML – Data Manipulation Language (LO6)

1. We are manipulating the contents of the database.

# Data Definition Language

CREATE TABLE statement

* Syntax:  
  CREATE TABLE TableName(  
   columnName datatype[constraint]  
   {, columnName datatype [constraint]}…  
   {, [constraints]}…

);

* Each column definition has 3 parts

DROP TABLE statement

* Syntax:  
  DROP TABLE[schema.]tableName[CASCADE CONSTRAINTS]

# Oracle Data Types

* Oracle does not use INT or TINYINT – instead we use NUMBER with an appropriate size. INTERGER and SMALLINT are the same as NUMBER(38).
* Oracle does not reasonably support DECIMAL (it lets you use DECIMAL, but ends up treating it as if it were a NUMBER(38) – with no decimal portion). Instead, use NUMBER(size, #decimals) or DECIMAL(size, #decimal).

# Number Data Types

* columnName NUMBER [([precision,][scale])]
* Subtypes:
  + Integer numbers
    - columnName NUMBER(precision)
  + Fixed-point numbers
    - columnName NUMBER(6,2)
  + Floating-point numbers
    - columnName NUMBER

# Character Datatypes

* CHAR
  + columnName CHAR [(max size)]
  + stores fixed-length character data up to a maximum of 2000 characters
* VARCHAR2
  + columnName VARCHAR2(maxSize)
  + Always uses VARCHAR2 and not VARCHAR
  + Stores variable length character data up to 4000 characters
* NVARCHAR2 and NCHAR
  + Only allows ASCII character set. Similar to VARCHAR2 and CHAR.

# Date and Time Datatypes

* Datatypes that store date and time values include **datetime** data subtypes, which store actual date and time values, and the **interval** data subtypes, which store an elapsed time interval between two date/time values. The main datetime subtypes are DATE and TIMESTAMP. The interval subtypes include INTERVAL YEAR TO MONTH and INTERVAL DAY TO SECOND.
  + DATE
    - columnName DATE
    - Stores dates from December 31, 4712 BC to December 31, 4712 AD.
    - Default time format is HH:MI:SS AM – using the 12 hour clock
  + TIMESTAMP
    - columnName TIMESTAMP (fractional seconds precision)
    - stores the date values similar to DATE datatype, except it also stores fractional seconds.
    - The default is 6 decimal places for fractional seconds precision

# Large Object (LOB) Datatypes

* columnName LOB\_data\_type
* Used to store binary data, such as digitized sounds or images or references to binary files from a word processor or spreadsheet.
* There are 4 LOB datatypes:

1. BLOB – Binary LOB stores up to 4GB of data in the database
2. BFILE – Binary files stores a reference to a binary file located outside the database in a file maintained by the OS
3. CLOB – Character LOB stores up to 4GB of character data in the database
4. NCLOB – Character LOB that supports 2-byte character codes, stores in the database – up to 4GB

# Constraints

* Constraints are rules that restrict the data values that you can enter into a column in a database table.
* There are two types of constraints:
  + Integrity constraints:
  + Value constraints:
* There are two levels of constraints
  + Table constraints:
    - Ex: Primary key constraint – specifies that a column value must be unique and cannot appear in more than one record.
  + Column constraints:
* You can place constraint definitions at the end of the CREATE TABLE command, after you declare all of the columns.
* OR you can place each constraint definition within the column definition.
* Each constraint must have a unique constraint name. You can define your own or omit them and Oracle will name your constraint. Max of 30 characters.

|  |  |
| --- | --- |
| **Constraint Type** | **Constraint ID Abbreviation** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Use TableName\_columnName\_constraintIDAbbreviation

Ex: Project\_projectID\_pk

* Types of constraints
  + Integrity constraints
    - Defines primary key columns and specifies foreign keys corresponding table and column references.
    - Primary Keys
      * Syntax:
        + Within a column
        + At the end of the CREATE TABLE
    - Foreign Keys
      * Syntax:
        + Within a column
        + At the end of the CREATE TABLE
        + Note: Before you can create a Foreign Key constraint the parent table must already exist.
    - Composite Key
      * You will need to list all of the columns involved.
      * Syntax (only creatable at the end of the CREATE TABLE statement):
  + Value Constraints
    - Column-level constraints that restrict the data values that users can enter into a given column
    - Check constraints
      * Specify that a column value must be a specific value or fall within a range of values.
      * Make sure you specify all conditions because once the table is created and populated it is difficult to modify the constraint.
      * DBMS must be able to evaluate each check condition to either true or false. You can combine expressions using logical operators AND and OR.
      * Syntax:
      * Examples:
    - NOT NULL Constraints
      * Specify whether the user MUST enter a value for a specific field or whether the value can be NULL.
      * Syntax:
    - Default Constraints
      * Specifies that a particular column has a default value that the DBMS automatically inserts for every record, unless the user specifies an alternate value.
      * Syntax:
    - Unique Constraints
      * Specifies that a column must have a unique value for every record
      * All primary keys are automatically assigned a unique constraint
      * Syntax:

# System Tables

We can retrieve information about a variety of database objects using different data dictionary views. These views simply contain data (structure and contents) about our data; this Metadata can be queried, and the result is a “view”.

|  |  |
| --- | --- |
| Object Name | Object Type |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## Unrestricted Actions and Restricted Actions

Various actions involving the database schema are classified as restricted or unrestricted.

* Modifying and Deleting Database Tables

1. Unrestricted Actions
2. Restricted Action

|  |  |
| --- | --- |
| **Restricted Action** | **Restriction** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## Renaming Existing Tables

Syntax:



## Alter Table - Making Changes to Existing Tables

* After a table has been created, its structure, properties, or constraints can be changed using the ALTER statement.
* Altering an empty table usually poses no difficulties. Altering a populated table may require ninja DBA skills to fix the data before the alteration can occur.

### Adding Fields to an Existing Table

Syntax:

### Deleting Fields from Existing Table

Syntax:

### Adding and Deleting Constraints Using Alter



### Modifying Existing Fields Using Alter Table