

ORACLE | DAY 4

SQL*Plus: Release 10.2.0.1.0 - Production on Wed Apr 30 16:13:38 2025

Copyright (c) 1982, 2005, Oracle. All rights reserved.

Connecting to oracle db /user user_H1 using connect command

Syntax:

Connect username/password;

Example:

```
SQL> connect user_H1/uh1;
```

```
Connected.
```

Creating table student in user_H1 but got **error** as same name table exists already

```
SQL> create table student(id int,name varchar(10),course varchar(10));
```

```
create table student(id int,name varchar(10),course varchar(10))
```

```
*
```

```
ERROR at line 1:
```

```
ORA-00955: name is already used by an existing object
```

Removing / dropping table student from database user_H1 to create same name table with different columns

Syntax:

Drop table tablename;

Example:

```
SQL> drop table student;
```

```
Table dropped.
```

Creating table student in user_H1

Syntax:

Create table table_name(column_name1 datatype, column_name2 datatype,..
column_nameN datatype);

Example:

```
SQL> create table student (id int, name varchar(10),course varchar(10));
```

Table created.

Inserted /added rows using complete insert i.e. inserting into all columns

Syntax:

Insert into table_name values (values to be inserted separated with comma);
or

Insert into table_name(column_names in table) values (values to be inserted
separated with comma);

Note: columnname beside tablename is optional in complete insert

Example:

```
SQL> insert into student values (1,'ram','C');
```

1 row created.

```
SQL> insert into student(id,name,course) values (2,'Sam','C++');
```

1 row created.

ORACLE | DAY 4

Inserted /added rows using partial insert i.e. inserting into specified/mentioned columns only but not all columns

Syntax:

Insert into table_name (**column_names in table in which values to be inserted**) values (values to be inserted separated with comma);

Note: column_name beside table_name is compulsory as it is partial insert

Example:

```
SQL> insert into student(id) values (3);
```

1 row created.

```
SQL> insert into student(name) values ('Deepa');
```

1 row created.

```
SQL> insert into student(course) values ('Sql');
```

1 row created.

ORACLE | DAY 4

Retrieving tables data/columns using select command.

Syntax:

Select columnnames from tablename;

Note: Asterisk * symbol can be used for all columns

SQL> select * from student;

ID	NAME	COURSE
1	ram	C
2	Sam	C++
3	Deepa	Sql

Inserted /added rows using partial insert into more than one column

SQL> insert into student(id,name) values (6,'Deepika');

1 row created.

SQL> insert into student(name,course) values ('Deepti','Java');

1 row created.

Retrieving tables data/columns using select command

SQL> select * from student;

ID	NAME	COURSE
1	ram	C
2	Sam	C++
3	Deepta	Sql
6	Deepika	
	Deepti	Java

7 rows selected.

Updating tables existing data using update command

Syntax:

Update table_name set column_name=new_value where condition;

Note: where condition optional depending on requirement

Example:

name ,course - 2 columns data updated

SQL> update student set name='Rita',course='Sql' where id=3;

1 row updated.

Retrieving tables data/columns using select command

SQL> select * from student;

ID	NAME	COURSE
1	ram	C
2	Sam	C++
3	Rita	Sql
	Deepa	
		Sql
6	Deepika	
	Deepti	Java

7 rows selected.

Updating tables existing data using update command

Id ,course - 2 columns data updated

SQL> update student set id=4,course='Sql' where name='Deepa';

1 row updated.

SQL> select * from student;

ID	NAME	COURSE
1	ram	C
2	Sam	C++
3	Rita	Sql
4	Deepa	Sql
		Sql
6	Deepika	

ORACLE | DAY 4

Deepti Java

7 rows selected.

Id ,name - 2 columns data updated

SQL> update student set id=5,name='Sarita' where course='Sql';

3 rows updated.

SQL> select * from student where rownum<6;

ID	NAME	COURSE
1	ram	C
2	Sam	C++
5	Sarita	Sql
5	Sarita	Sql
5	Sarita	Sql

Deleted rows where id is 5 using delete command

Syntax:

Delete from table_name where condition;

Note: where condition is optional.

Example:

SQL> delete from student where id=5;

3 rows deleted.

ORACLE | DAY 4

```
SQL> select * from student;
```

ID	NAME	COURSE
1	ram	C
2	Sam	C++
6	Deepika	
	Deepti	Java

course – single/1 column data updated

```
SQL> update student set course='Sql' where id=6;
```

```
1 row updated.
```

Id - 2 columns data updated

```
SQL> update student set id=7 where name='Deepti';
```

```
1 row updated.
```