Database Systems Laboratory 2

SQL Fundamentals

Chittaranjan Pradhan
School of Computer Engineering,
KIIT University

SQL Fundamentals

Student table

Roll	Name	City	Age	CGPA
101	Ram	Bhubaneswar	19	9.0
102	Hari	Bhubaneswar		6.7
103	Uday	Jharkhand	20	8.97
104	Vikas	Uttar Pradesh	19	8.5
105	Sweta	Ranchi	19	9.2
106	Yogesh	Rajastan	18	7.9
210	Smriti	Delhi	20	8.99
211	Sudam	Cuttack	21	8.6
212	Vikas	Kolkota	23	5.98
165	Manish		19	9.15

- The character data is displayed with left justification, while numeric data with right justification

SELECT Statement

SELECT statement

SELECT statement is used to retrieve data from the underlying table. The syntax is:
SELECT columns FROM tablename;

If the user wants to see all the columns in a table, * can be used in place of columns

SELECT Roll, CGPA FROM Student;

Roll	CGPA
101	9.0
102	6.7
103	8.97
104	8.5
105	9.2
106	7.9
210	8.99
211	8.6
212	5.98
165	9.15

SELECT Statement...

Displaying Distinct Rows

The DISTINCT keyword is used to suppress duplicate values.
The syntax is:
SELECT DISTINCT column FROM tablename;

SELECT DISTINCT City FROM Student;

City
Bhubaneswar
Jharkhand
Uttar Pradesh
Ranchi
Rajastan
Delhi
Cuttack
Kolkota

SELECT Statement...

Use of Arithmetic Expressions

The arithmetic expressions are used to display mathematically calculated data. The syntax is:
SELECT column, expression FROM tablename;

SELECT Name, Age, Age+3 FROM Student;

Name	Age	Age+3
Ram	19	22
Uday	20	23
Vikas	19	22
Sweta	19	22
Yogesh	18	21
Smriti	20	23
Sudam	21	24
Vikas	23	26
Manish	19	22

SELECT Statement...

Use of Alias

The column aliases are used to rename a table's columns for the purpose of a particular SQL query. The syntax is:
SELECT column1, column2 [AS] Alias FROM tablename;

SELECT Name, Age, Age+3 "Passing Age"FROM Student;

Name	Age	Passing Age
Ram	19	22
Uday	20	23
Vikas	19	22
Sweta	19	22
Yogesh	18	21
Smriti	20	23
Sudam	21	24
Vikas	23	26
Manish	19	22

SELECT Statement...

Concatenation

Concatenation joins a column or a character string to another column. The syntax is:
SELECT column1||' '||column2 [AS] ALIAS FROM tablename;

SELECT Name||' '||City FROM Student;
SELECT Name||' '||City AS "Address"FROM Student;

Name||' '||City	Address
Ram Bhubaneswar	Ram Bhubaneswar
Hari Bhubaneswar	Hari Bhubaneswar
Uday Jharkhand	Uday Jharkhand
Vikas Uttar Pradesh	Vikas Uttar Pradesh
Sweta Ranchi	Sweta Ranchi
Yogesh Rajastan	Yogesh Rajastan
Smriti Delhi	Smriti Delhi
Sudam Cuttack	Sudam Cuttack
Vikas Kolkota	Vikas Kolkota
Manish	Manish

SELECT Statement...

Displaying Time

Time of a date-type column can be displayed by using TO_CHAR(). The syntax is:

SELECT Roll, DOB FROM Student;
SELECT Roll, TO_CHAR(DOB, 'DD-MON-YYYY HH:MI:SS A.M.') FROM Student;

Selecting Specific Records

Specific records can be selected by using a WHERE clause with the SELECT statement. The syntax is:
SELECT columns FROM tablename WHERE *cond*^{n}**;**

*SELECT * FROM Student WHERE city= 'Bhubaneswar';*

Roll	Name	City	Age	CGPA
101	Ram	Bhubaneswar	19	9.0
102	Hari	Bhubaneswar		6.7

Operators used in WHERE condition

Relational Operators

=	ex: CGPA=9.0
>	ex: Age>20
<	ex: Age<20
>=	ex: Age>=20
<=	ex: Age<=20
<> or !=	ex: Name !='Hari'
ANY	ex: Age > ANY(20,23,19)
ALL	ex: Age > ALL(20,18)

Logical Operators

AND	ex: City='Bhubaneswar' AND Age=20
OR	ex: City ='Bhubaneswar' OR Age=20
NOT	ex: NOT(Age=20 OR Age=21)

AND has more precedence than OR

Operators used in WHERE condition...

LIKE Operator

LIKE operator uses wild cards for matching as:
%: represents zero or more characters
_: represents any one character

ex: Name LIKE 'S%'
ex: Name LIKE 'S__'
ex: Name LIKE '%i%'
ex: Name LIKE '_i%'

Special Operators

IN ex: City IN('Delhi','Cuttack','Ranchi')
BETWEEN ex: Age BETWEEN 20 AND 22
IS NULL ex: SELECT Name FROM Student WHERE Age is NULL;

Name	Age
Hari	

Sorting

ORDER BY clause using column name

ORDER BY clause is used to sort records in a table
SELECT columns FROM tablename [WHERE *cond*^{n}]
ORDER BY column [ASC/DESC];

*SELECT * FROM Student ORDER BY Age;*
*SELECT * FROM Student ORDER By CGPA, Age DESC;*
NULL values come at the end of the table in case of ORDER BY clause

ORDER BY clause using column number

Records can be sorted by using the column number
SELECT columns FROM tablename [WHERE *cond*^{n}]
ORDER BY columnno [ASC/DESC];
*SELECT * FROM Student ORDER BY 3;*

ALTER Statement

Stud (roll, name, age)

Column	NULL?	Datatype
ROLL		NUMBER(6)
NAME		VARCHAR2(20)
AGE		NUMBER(2)

Adding a New Column

ALTER TABLE tablename ADD(column definition);

ALTER TABLE Stud ADD (address number(20));

Column	NULL?	Datatype
ROLL		NUMBER(6)
NAME		VARCHAR2(20)
AGE		NUMBER(2)
ADDRESS		NUMBER(20)

ALTER Statement...

Modifying an Existing Column

ALTER TABLE tablename MODIFY(column definition);

ALTER TABLE Stud MODIFY(address varchar2(20));

Column	NULL?	Datatype
ROLL		NUMBER(6)
NAME		VARCHAR2(20)
AGE		NUMBER(2)
ADDRESS		VARCHAR2(20)

Dropping a Column

ALTER TABLE tablename DROP COLUMN columnname;

ALTER TABLE Stud DROP COLUMN address;

Column	NULL?	Datatype
ROLL		NUMBER(6)
NAME		VARCHAR2(20)
AGE		NUMBER(2)

ALTER Statement...

Renaming a Column

ALTER TABLE tablename RENAME COLUMN oldname to newname;

ALTER TABLE Stud RENAME COLUMN roll to id;

Column	NULL?	Datatype
ID		NUMBER(6)
NAME		VARCHAR2(20)
AGE		NUMBER(2)

UPDATE Statement

Roll	Name	Age	Branch
101	Vikas	19	
102	Soheb	20	
103	Gita	18	
104	Monalisa	19	
105	Ganesh	20	

UPDATE Statement

UPDATE tablename SET columnname=value [WHERE *cond*^{n}];

UPDATE Stud SET Branch='CSE' WHERE Roll=101;

Roll	Name	Age	Branch
101	Vikas	19	CSE
102	Soheb	20	
103	Gita	18	
104	Monalisa	19	
105	Ganesh	20	

UPDATE Statement...

UPDATE Stud SET Branch='CSE';

Roll	Name	Age	Branch
101	Vikas	19	CSE
102	Soheb	20	CSE
103	Gita	18	CSE
104	Monalisa	19	CSE
105	Ganesh	20	CSE

DROP Statement

DROP command

DROP TABLE tablename;

DROP TABLE Stud;

TRUNCATE Statement

TRUNCATE command

TRUNCATE TABLE tablename;

TRUNCATE TABLE Stud;

DELETE Statement

DELETE command

DELETE FROM tablename [WHERE *cond*^{n}];

DELETE FROM Stud WHERE Roll=101;

DELETE FROM Stud;

RENAME Statement

RENAME command

RENAME oldname TO newname;

RENAME Stud TO Student;

Viewing User Tables

Viewing all user Objects

```
SELECT * FROM TAB;
```

Viewing all user Tables

```
SELECT table_name FROM user_tables;
```

Creating Table from another Table

Creating Table from another Table

CREATE TABLE tablename(column1,column2) AS SELECT column1,column2 FROM tablename;

CREATE TABLE Person(Roll, Name, Age) AS SELECT Roll, Name, Age FROM Stud;

The SQL statement populates the target table with data from the source table

Inserting data into a Table from another Table

Inserting data into a Table from another Table

INSERT INTO tablename SELECT column1, column2 FROM tablename[WHERE *cond*^{n}];

INSERT INTO Person SELECT Roll, Name, Age FROM Stud WHERE Roll=101;

INSERT INTO Person SELECT Roll, Name, Age FROM Stud;

ROWID

- Each row has a unique ROWID
- It is an 18-bit number and represented as a base-64 number
- It contains the physical address of a row in a database

In case user has inputted same records more than one time, ROWID is used to distinguish each record

For example, consider Customer table

Cid	CName	Address
1	Akash	BBS
2	Amir	BBS
2	Amir	BBS
3	Ashok	CTC

ROWID...

If the user wants to delete the duplicate copies of the same record, then ROWID is used

SELECT ROWID, Cid FROM Customer;

ROWID	Cid
AAAF4YAABAAAHCKAAA	1
AAAF4YAABAAAHCKAAB	2
AAAF4YAABAAAHCKAAC	2
AAAF4YAABAAAHCKAAD	3

DELETE FROM Customer WHERE ROWID= 'AAAF4YAABAAAHCKAAC';