1. SQL Statements - 1

(a) Writing basic SQL SELECT Statement.

SELECT Query

SELECT query is used to retrieve the data from database. SELECT query never make any

change in the database.

Syntax

[SELECT column\_name1, column\_namel... from table\_name;

If we want to retrieve data from all the columns of a table then instead of writing all the

column name just use ‘\*’. The ‘\* ‘symbol represent all the columns.

For Example

With SELECT statement different clauses can be used to display the data as per our

requirements.

WHERE clause

WHERE clause is used to specify condition in SELECT statement while fetching records

from the database. The records satisfying the condition given by where clause are retrieved.

Syntax

SELECT column\_1,columns\_2... from table\_name where condition;

For Example

select \* from Employee where Salary > 15000;

Select \* from Employee where Name='Smar ;

b) Restricting and Sorting Data.

DISTINCT clause

This clause is used to avoid selection of duplicate rows.

Consider a situation where duplicate values for Salary column are present in Employee

table, and the manager wants to know the salary amount given to the employees. To avoid

duplication of salary amount we use distinct keyword as follows:

select distinct(salary) from Employee;

ORDER BY clause

To arrange the displayed rows in ascending or descending order of given column, Order

By Clause is used.

For Example

We need to display the employee information as per their joining dates in ascending order,

the query will be

Select \* from Employee order by Name;

Now to display same information in descending order on job the query will be

Select \* from Employee order by Name desc;

(c) Single - Row Functions

Single row functions can be

o Case Conversion functions : Accepts character input and returns a character value.

Functions under the category are UPPER, LOWER and INITCAP.

o UPPER: UPPER function converts a string to upper case.

o LOWER : LOWER function convert a string to lower case.

o INITCAP function converts only the initial alphabets of a string to upper

For example

Select Emp\_no, UPPER(Name), LOWER(city) from employees;

o Character functions : Accepts character input and returns number or character

value. Character functions are as follows:

o CONCAT : CONCAT function concatenates two string values.

For example

Select CONCAT(First\_name, Last\_name) from Employee2;

o LENGTH: LENGTH function returns the length of the input string

For example

Select First\_name, cesT CHP srame)f from Employee2;

o SUBSTR : SUBSTR function returns a portion of a string from a given start point to

an end point.

o INSTR : INSTR function returns numeric position of a character or a string in a

given string.

For example

select First\_name, INSTER(First. name, ‘s

o LPAD and RPAD functions pad the given string upto a specific length with a given

character. |

o TRIM: TRIM function trims the string input from the start or end,

For example

o REPLACE : REPLACE function os Selnes characters from the input string with a

given character.

o Number functions : Accepts numeric input and returns numeric values. Functions

under the category are ROUND, TRUNC, and MOD.

o ROUND and TRUNC functions are used to round and truncate the number value.

For example

SELECT ROUND(84.555,1) from dual;

© MOD is used to return the remainder of the division operation between two numbers.

For example

SELECT MOD(10,2) from dual;

2. SQL Statements - 2

(a) Displaying Data from Multiple Tables.

To understand joins consider following two tables.

For example

inner Join (Equi Join)

The INNER JOIN is used to display records that have matching values in both tables.

Syntax

Select column\_name\_list from table\_1

INNER JOIN table\_2 |

where table\_1.column\_name = table\_2.column\_name

For Example

select product\_details.product\_id, product\_name, customer\_name , sale\_details.quantity,

product\_details.price from product\_details INNER JOIN sale\_details on —

product\_details.product\_id=sale\_details. product\_id ;

Bsa select product\_details.product\_id, product\_nane, custoner\_nane , sale\_de

ails.quantity, product\_details.price fron product\_details INNER JOIN sale\_detai

Is on product\_details.product\_id=sale\_details.product\_id ;

Outer Join -

Outer Join is based on both matched and unmatched data. Outer Joins subdivide further

into,

(i) Left Outer Join (ii) Right Outer Join

The SQL LEFT JOIN returns all rows from the left table, even if there are no matches in

the right table, Null values are shown at the place of right table values.

Syntax

SELECT column-name-list

from table-name1

LEFT OUTER JOIN

table-name2

on table- -namel. column-name = ‘table- name2. column- -name;

For Example

SELECT product\_details.product\_id, product’. details. product\_name,

sale\_details.customer\_name FROM product\_details LEFT OUTER JOIN sale. details ON.

sale\_details. product\_id =product\_detdils.product\_id:

Jails. customer\_name FROM product\_details LEFT OUTER JOIN sale\_details ON saledet

jails. profuctid oye ees. product\_id;

(ii) Right Outer Join

Returns all rows from the right table even if there are no matches in the left table. Null

values are shown at the place of left table values.

Syntax

select column-name-list from table-name1

RIGHT OUTER JOIN

table-name2

pon table-name1.column-name = table-name2.column-name;

For Example

SELECT product\_details.product\_id, product\_details.product\_name,

sale\_details.customer\_name FROM product\_details RIGHT OUTER JOIN sale\_details ON

sale\_details.product\_id =product\_details. product\_id; a

(b) Aggregating Data Using Group Functions.

Count()

This function returns total number of values of specified column of the table.

For Example

Write a query to retrieve count of employees who join in 2015 year.

‘Select ‘count(Emp.no):f from. Employee2 where First\_name like 'swa%';

Sum(Q)

This function returns sum of all the values of specified column of the table.

For Example

Write a query to to find total salary amount paid to all the employees.

select ct Sum Salary) from ployed

Min()

This function returns smallest value from specified column of the table.

For Example

Write a query to retrieve record of employee who gets least salary.

[select \* from Employee2 where Salary =(Select min(Salary) from Employee2); |

Max()

This function returns greatest value from specified column of the table.

For Example

Write a query to retrieve record of employee who gets maximum salary.

[Select \* from Employee2 where Salary=(Select max(Salary) from Employee2);

Avg()

The following SQL statement finds the average salary of all employee:

For example

select Avg(salary) from Employee2;

(c) Sub queries

Sub-Queries

Writing a query inside another query is known as nested query or subquery. The inner

query get executed first, then the output on inner query is given as input to outer query.

Consider the previous emp table

Example

To display records of employees working in SMITH’s department

Select \* from emp where deptno =

(select deptno from emp where ename = ‘SMITH’);

Output

EMPNO | ENAME | JOB MGR | HIREDATE | SAL | COMM | DEPTNO

7566 JONES \_| MANAGER | 7839 | 04/02/1981 | 2975 20

[7788 ~~ | scott | ANALYsT | 7566 | 12/09/1982 | 3000 20

7902 FORD | ANALYST | 7566 | 12/03/1981 | 3000 20

7369 SMITH | CLERK 7902 | 12/17/1980 | 800 20

7876 ADAMS | CLERK 7788 | 01/12/1983 | 1100 20

3. Manipulating Data -

Data Manipulation Language (DML)

- The Data Manipulation Language (DML) is used for accessing and manipulating data in a

\_ database. It allows users to access, insert, update, and delete data from the database.

© To insert record into the table - INSERT

o Toaccess or read records from table -SELECT

© Update the records in table - UPDATE

© Delete the records from the table- DELETE

(a) Using INSERT STATEMENT

1 Inserting record into tables

After creation of table the insert command is used to insert one or more records in the

lable,

Insert query has different forms.

https://E-next.in Scanned by CamScanner

Lab Manual

[YF vams imu-asc 17) rn

Format 1: Inserting a single row of data into a table

Syntax

Insert into table\_name[(column\_namet, column \_name2,--)]

values (value1, value2....);

For example

|. Insert 5 records in Employee table (102 Swati’, 9015-01-

Insert into Employee values(101,'Rajesh’, 1995-110 05-02-

07°.20000), (103,"Vedant’’1999-05-03',30000), (10%

10',52000),(105,’Ankita’, '201603-01',8000):

After inserting 5 records in table the table will look like as follow\*:

-11-02! ,12000);

uj Chaamp bi sqhmpaLatbin mya.ne :

ysql> Insert into Employee values(101, Rajesh’ s "1995

uery OK, 1 row affected (0.02 sec)

12915-01-07' ,20000);

ysq|> Insert into Employee values (102, Swati’,

uery OK, 1 row affected (0.00 sec)

ysq|> Insert into Employee values(103,'vedant’,

uery OK, 1 row affected (0.00 sec)

ysql> Insert into Employee values(104, 'vedika’,

very OK, 1 row affected (0.00 sec)

values(105, Ankita’,

ysql> Insert into Employee 2

uery OK, 1 row affected, 1 warning (0.00 sec)

1999-05-03’, 30000);

2005-02-10’, 52000);

1901603-01’ ,8000) ;

ysqi> select \* from Employee;

paneeen----- wpenn nner wonnn nnn trent

| Employee\_no | Employee\_name | joining\_date | Salary |

ae pocec cen nemo nee npr mmr en mere rs eo

101 {| Rajesh | 1995-11-02 12000 |

102 | Swati | 2015-01-07 20000

103 | Vedant 1999-05-03 | 30000

104 | Vedika 2005-02-10 | 52000

105 | Ankita 0000-00-00 [| 8000

wneeenee nnn fence ep rn pn 9:

il

os

5 rows in set (0.00 sec)

verte

2. Consider we do not have value for salary while inserting a new record in Employee

table. Then the query will be

Insert into Employee(Employee\_no, Employee\_name, Joini |

— ' ‘ae , n n

values(106,’Ankur,'2017-03-15'); Ingedate)

It will set salary value for the employee as NULL.

https://E-next.in

Scanned by CamScanner

a

3} chwamplbennysqhi sg 5.1.) bininysal.ee

pe i

‘

i

t. a a

nys I> Ins

F aku’ 2017 Ose agey vee Cemmoyee no, Employee\_nane, Joining\_date) values (106

query OK, 1 row affected (0.02 sec)

rysal> select \* from Employee;

ett apr nr een e nen ey

mpeneenene nnn en- foenenn=n+

| Exployee.no | Employee. name joining\_date | Salary |

wee een een), naam ee eases memes oponnnn---Ft

| 101 | Rajesh 1 i95- 11-02 | 12000

| 102 | Swati 2015-01-07 20000

| 103 | Vedant 1999-05-03 30000

| 104 Vedika 2005-02-10 52000

| 105 | Ankita 0000-00-00 8000

106 | Ankur 2017-03-15 | NULL |

pomnn naan anna onn nnn nennn en penennnenenene mponennn= “+

6 rows in Set (0.00 sec)

mysql>

Format 2 : Inserting data into a table from another table

Syntax

«GR

Insert into table\_name select column\_name1, columne\_name2 from table\_name;

Insert records in newEmployeel table same as in Employee table.

For Example

Insert into newEmployee1 select \* from Employee;

i

Fett 0a jer

ysql> select \* from wenkng Toye;

moty set (0.00 sec)

ysql> Insert into newmployeel select \* from Employee;

puery OK, 6 rows affected (0.01 sec)

ecords: 6 Duplicates: 0 Warnings: 0

pysal> select \* from newemployeel;

f Emp loyeeno | | Enployeeriane | | joining date Salary |

nnn mewn nn nn eng mm enn en nnn nnn brn e ++

101 7 Rajesh 7 1995-11-02 | 12000 |

| 102 | Swati 2015-01-07 | 20000 |

103 | Vedant 1999-05-03 30000

104 | Vedika 2005-02-10 | 52000

105 | Ankita 0000-00-00 | 8000

| 106 | Ankur 2017-03-15 NULL |

lp enw nme nn mam mpa mere nn nnn” opr mer nn eenmnnn mpaceeenn= +

6 rows in set “0. 00 sec) :

https://E-next.in

Scanned by CamScanner

\_

)} Dems (Mu-Bsc iT) L-18 Lab Manual

SS

(b) Using DELETE STATEMENT

Delete

As per requirement, the records from existing table can be removed using delete

command. Delete command can have ‘WHERE’ clause optionally.

Syntax

Delete from table\_name;

For Example

Write a query to remove record of Employee\_no 106 from Employee table,

Delete from Employee where Employee\_no = 106;

EI chuamplboimpahmyqSa mbites 9 PE he SE

mysql> select \* from Employee; ;

f—-—-------- + +------------- + t----- eo errcon tora arn + 4

| Employee\_no | Employee\_name | joining\_date | Salary | e

anole t---------------4------------ --4-- - = t mT

| 101 | Rajesh | 1995-11-02 | 12000 |

| 102 | Swati | 2015-01-07 | 20000 |

| 103 | Vedant | 1999-05-03 | 30000 |

| 104 | vedika | 2005-02-10 | 52000 |

| 105 | Ankita | 0000-00-00 | 8000 |

| 106 ! Ankur | 2017-03-15 | NULL |

Hh -- - === ---=- -4------- -- = + +-------------- 4+-------- +

6 rows in set Co. 0S sec)

mysql> Delete from Employee where Employee\_no = 106;

Query OK, 1 row affected (0.00 sec)

mysqi> select \* from Employee;

ot \_—\_—\_——o—o eo ee a a a a a a ae ae ee ee

+-------- +

| Employee\_no 1 Employee\_name 1 joining\_date | Salary |

------------- torn tren nn nto -- =

| 101 | Rajesh | 1995-11-02 | 12000 |

| 102 | Swati | 2015-01-07 | 20000 |

| 103 | vedant | 1999-05-03 | 30000 |

| 104 | vedika | 2005-02-10 | 52000 |

| 105 | Ankita | 0000-00-00 | sooo |

t+t------------- teccccsccrctc tenn noon +-------- + -

i

fi

e

eR

(c) Using UPDATE STATEMENT

Update

To make changes in the database ‘update’ command is used. The update command

consists of ‘set’ clause and an optional ‘where’ clause’. ‘WHERE? clause is used to make

changes in specific records.

Syntax

Update table\_name set column\_name = new\_value [where condition];

https://E-next.in Scanned by CamScanner

wa

Ee L-19 \_\_\_ bab Manual

Hor Example

[ (ipdate Employee set =eianye50000 where Employee. no=105; |

§ J ciaanwihamnealinn act LANbir onsale ee Td

ysql> select \* From. Employee:

-o----------- to------ ney == --4

| Employee\_no | Employee\_name | joining\_date | salary |

eer rt eee 4--------------4--------+

| 101 | Rajesh | 1995-11-02 12000 |

| 102 | swati | 2015-01-07 20000 |

| 103 | vedant | 1999-05-03 | 30000 |

| 104 | vedika 2005-02-10 | 52000 |

| 105 | ankita 0000-00-00 | 8000 |

we nr tn nr a ee ee $------- - - - - - - - 4 -- ot

5 rows in set Co. 00 sec)

ysql> Update Employee set Salary=50000 where Employee\_no=105;

uery OK, 1 row affected (0.00 sec)

Rows matched: 1 Changed: 1 warnings: 0

ysqi> select \* from Employee;

errr rr ro t------ - -- - -- - - - 4 - - - - - nt

| Employee\_no I Employee\_name | joining\_date | Salary |

eth er ne - ha rt rr rat

| 101 Rajesh | 1995-11-02 {| 12000 |

| 102 | swati |. 2015-01-07 | 20000 |

| 103 | Vedant 1999-05-03 | 30000 |

| 104 | Vedika. 2005-02-10 {| 52000 |

| 105 | Ankita | 0000-00-00 | ©50000 |

----+--=----- 4+-----------~----+--------------4--------+

Picco. i oe ETE, a

ySql> Update Employee set coining Hate 2015 03- iv where Employee pei

uery OK, 1 row affected (0.01 sec)

ows matched: 1 Changed: 1 warnings: 0

nysql> select \* from Employee;

~ aaa nanan pom n nnn repose nse rrr rsa pane ecnnd :

H Emp loyee\_no ! Emp loyee\_name | joining\_date | Salary |

Hanan nn nana nnn pon nnn nnn nn nnn npeennn cn neenen= feoen-n=- +

| 101 | Rajesh | 1995-11-02 | 12000 |

| 102 | Swati | 2015-01-07 | 20000 |

| 103 | vedant | 1999-05-03 | 30000 |

| 104 | Vedika | 2005-02-10 | 52000 |

| 105 | Ankita | 2015-03-15 | 50000 |

iM wenn nnn nnn perenne nan enne torent rrr rnc foenannna +

5 rows in set (0. 00 sec)

Pisa

par tS,

https://B-next.in Scanned by CamScanner

Serene we eee

apm

Ee g eos ch IR

MP Domne

heen

4. Creating and Managing Tables

(a) Creating and Managing Tables

Creating Table

The CREATE TABLE statement is used to create table in database.

Syntax .

| DBMS (MU-BSC IT) L-20 Lab Manuay

ES SSS.SsSsSsSsSMSSSS08$0Moe&SSS

CREATE TABLE table\_name (

column] datatype[size],

column2 datatype [size],

column3 datatype[size],

wees

3

Example

Following command creates a table Employee having four columns

: Employee\_no

>

Employee\_name, Joining \_date, and Salary.

CREATE Table Employee ( —

Employee\_no integer(3), °

Employee\_name varchar(20),.

joining\_date date ,

Salary integer(6));

1. Creating New Table from Existing Table

Consider the existing table Employee

Creating new table same as of existing table.

Syntax

Create table-table\_ name as select \* from existing table name;

For Example

Create table nEwEmpleyess ; RES

PAS ee

select \* -from Erployessi:

Employee table.

The newly created table newEmployee! will include all the fields and records as in

https://E-next.in Scanned by CamScanner

at DBMS (MU-BSC IT) 24 Lab Manual

—\_—

Creating new table having specific fields but all the records from existing table.

Syntax

[ Create table table\_name as select field\_\,field\_2... from existing\_table\_name, a

For Example

Create table newEmployee2

As

select Employee\_no, Employee\_name from Employee;

After executing this command the newly created table will contain two fields such as

Employee\_no and Employee\_name , and also contain all the corresponding records as in

Employee table.

Creating new table having specific records but all the fields from existing table.

Syntax

[ Create table table\_name as select \* from existing \_table\_name where condition

For Example

Create table newEmployee3

As

select \* from Employee

where Salary > 80000;

The newly created table will have same structure as of employee table, but it will contain

records of only those Employees who got Salary above 80000 Rs.

Creating new table having no records but all the fields from existing table.

That means copying only structure of existing table

Syntax

Create table table\_name as select \* from existing table\_name where false condition

For Example

Create table newEmployee4

As

select \* from Employee

where 1=2;

htips://E-next.in Scanned by CamScanner

[ase DBMS (MU-BS

i x U-BS >

Se NU-BSC IT) L-22 Lab Manual

Se —

a

“re | =9 ic \ }

Here 1=2 is the false condition

The newly create “exist iw

1ewly created table will have same structure as of existing table, but it will not copy

ANY records from,

4 Bhan

2. Modifying Table

ALTER TABLE query is used to modify structure of a table which is already exists in the

database. We can add, delete or modify column.

Adding New Column in a Table

Syntax

ALTER TABLE table\_name ADD column\_name datatype;

For Example

ALTER TABLE Emp\_dept ADD column city varchar(20);

1 clvang Balms bye | lo) 8 heal

sql> alter table Enp\_dept add coluan city varchar(20); '

uery OK, } rows affected (0.11 sec)

ecords: } Duplicates: 0 Warnings: 0

gl> alter table Emp\_dept add colunn ph\_no integer(20);

uery OK, 5 rows affected (0.09 sec)

ecords: 5 Duplicates: 0 Warnings: 0

fees]

Lp I POPP PUES A SF rr rrr

Dropping Column from Table

Syntax

ALTER TABLE table\_name DROP COLUMN column\_name;

For Example:

ALTER TABLE Emp\_dept DROP COLUMN dept;

https://E-next.in Scanned by CamScanner

>

(et DBMS (MU-BSC IT) L023 Lab Manual

—\_—\_

on St Bg aa = ae

Tchnang be nisg msqilSbalmsal ae oS ee

|

ysql> alter table Enp\_dept Drop colunn dept’

very OK, 4 rows affected (0.10 sec)

ecords: 4 Duplicates: 0 Warnings: 0

——

=—

nysql>

i

Modifying Column of a Table

Here we are changing the data type and size of column roll\_number.

Syntax

ALTER TABLE table\_name MODIFY COLUMN column\_name data\_type;

For Example

ALTER TABLE Employee modify column Employee\_no varchar(4);

=a

T evang bi mysghmsqhl Minimal

ysql> ALTER TABLE Employee MODIFY COLUMN name varchar (50);

uery OK, 3 rows affected (0.11 sec)

ecords: 3 Duplicates: 0 Warnings: 0

psa?

3. Renaming Table

Syntax

rename table current\_table\_name to new\_table\_name;

For Example

[rename table Employee to Emptable;

https://E-next.in

Scanned by CamScanner

Lab Manual

ie DBMS (MU-BSC IT) L-24

14 chnamelbin gah WSqNLS bavmysal ee | fo) 6 fe

m rename table Enployee to Enptable; U

\_ OK, 0 rows affected (0.00 sec)

A

yap ,

| " -~ Le Is

Nees eesn a a et Ac a Aan anne ennsonan enaieniaie

4. Deleting Table

DROP TABLE query is used to delete table.

Syntax

drop table table\_name;

For Example

Hop table Emptable:

7 Ucw ane nyahyo63 >in mee :

mysql> Drop table Emptable; a

uery OK, 0 rows affected (0.00 sec)

(b) Including Constraints

NOT NULL constraints

The NOT NULL constraint enforces a column to NOT accept NULL values

For example

Create table Persons(ID integer NOT NULL, LastName varchar(50) NOT NULL, FirstName

varchar(50) NOT NULL, Age integer(10));

https://E-next.in Scanned by CamScanner

>

° ams (MU-BSC IT) 26 Lab Manual

ij cjwangitin mysqf.mysq5.d Mibin' mysqlee [of 5 be

ysq|> CREATE TABLE PERSONS (

-> ID int NOT NULL, 7

-> LastName varchar(200) NOT NULL,

-> FirstName varchar (200) NOT NULL,

-> Age int);

query OK, 0 rows affected (0.08 sec)

nysq|> . .

1 si A ket ag acai eae z

pst insert into PERSONS values(1, ‘sawant',‘sandhy" ,nul1); iL

uery OK, 1 row affected (0.00 sec) ni

ysql> insert into PERSONS values(2,sawant’,'nul1' null);

uery OK, 1 row affected (0.00 sec)

ysql> insert into PERSONS values(2, ‘sawant" ,nu11,null);

ERROR 1048 (23000): Column ‘FirstName’ cannot be null

rsa :

Unique constraints

The UNIQUE constraint ensures that all values in a column are different.

For example

Create table Persons(ID integer NOT NULL UNIQUE, LastName varchar(50) NOT NULL,

FirstName varchar(50) NOT NULL, Age integer(10));

https://E-next.in Scanned by CamScanner

[FP dems mmu-asc IT) L-26 \_\_bab Manual

4) clwampt bin tal mysql Ri bin\ryrgloe i= ea

mysql> CREATE TABLE PERSONS2(

-> IO int NOT NULL UNIQUE,

~> LastName varchar(200) NOT NULL,

~> FirstName varchar(200) ,

-> Age int);

Query OK, 0 rows affected (0.05 sec)

Gs}

mysql> insert into PERSONS2 values(1,'sawant', samarth' null);

Query OK, 1 row affected (0.02 sec)

mysq|> insert into PERSONS2 values(1, ‘sane’, 'sanika’ ,nu11);

ERROR 1062 (23000): Duplicate entry '1' for key ‘ID

hysal>

Primary key constraints: Primary keys must contain UNIQUE values, and cannot

contain NULL values

Primary key on create table

For example

Create table Persons(ID integer NOT NULL UNIQUE, LastName varchar(50) NOT NULL,

FirstName varchar(50) NOT NULL, Age integer(10) PRIMARY KEY(ID));

tans

Tekeybaauinepitiindce eee

mysq]> CREATE TABLE PERSONS2( i

-> ID int NOT NULL, :

-> LastName varchar(200) NOT NULL,

-> FirstName varchar(200) ,

-> Age int,

-> PRIMARY KEY(ID)

-> );

Query 0K, 0 rows affected (0.03 sec)

|

htps://E-next.in Scanned by CamScanner

r-

MU-

(onus | BSC IT)

Manual

L-27 a

Primary key on Alter table

For example

[Alter table Person2 ADD PRIMARY KEY(ID); sa

I chang bry ry. Mig | 3 | ol 6

ysql> ALTER TABLE PERSONS

~> ADD PRIMARY KEY(ID);

Uery OK, 2 rows affected (0.05 sec) u

ecords: 2 Duplicates: 0 Warnings: 0

ysq]> 7

"testa eer a L eames J

Drop primary key constraints:

For example

Alter table Person2 DROP PRIMARY KEY: -

mysql> ALTER TABLE PERSONS2

| => DROP PRIMARY KEY; Qo

uery OK, 0 rows affected (0.06 sec) ?

ecords: 0 Duplicates: 0 warnings: 0

hysol> A

CHECK Constraint

The CHECK constraint is used to limit the value range that can be placed in a column

CHECK on create table

For example

create table student(ID integer NOT NULL, LastName varchar(50) NOT NULL, FirstName

| varchar(50) NOT NULL, Age integer(10) CHECK(Age>=18));

https://B-next.in Scanned by CamScanner

(ae? pps (mu-ssc m L-28 eat

Ss ——— . O\_o SS

EIS eee a lel

Lea create table student(

-> ID integer NOT NULL,

-> LastName varchar (50) NOT NULL, 5

\_ > FirstName varchar(50) NOT NULL, :

-> Age integer(10), CHECK(Age>=18));

Query OK, 0 rows affected (0.03 sec)

ysol> .

CHECK on Alter Table

For example

Alter table student ADD CHECK (Age>=18);

nysql> Alter table student ADD CHECK (Age>=18);

uery OK, 0 rows affected (0.08 sec) 0

ecords: 0 Duplicates: 0 warnings: 0 ;

ysq|> ., ;

5. Creating and Managing Other Database Objects

(a) Creating Views

Creating View

Consider we have existing table as Employee (Employee\_no, Employee\_name,

joining\_date, Salary).

1. Creating view having all records and fields from existing table

Syntax

CREATE or replace VIEW view\_name

AS

SELECT column], column2, ...

FROM table\_name :

—\_

https://E-next.in Scanned by CamScanner

fos IT) \_Le

WHERE condition; Tape or

For Example

Lab Manual

Create or replace view Emp\_view1

AS

select \* from Employee;

This statement will create view having all the fields as in Employee table.

2, Creating view having specific fields but all the records from existing table

Syntax

Create or replace view view\_name

AS

select field\_1, field\_2... from existing\_table\_name;

For Example

Create or replace view Emp\_view2

As

| select Employee\_no, Employee\_name from Employee;

This statement will create view having specific fields from Employee table.

3. Creating new view having specific records but all the fields from existing table

Syntax

Create or replace view view\_name

As

select \* from existing\_table\_name

where condition;

For Example

Create or replace view Emp\_view3

As

select \* from Employee

where Salary > 80000;

(b) Other Database Objects

An index is a pointer to data in a table. An index in a database is similar to the

alphabetical index of a book present at the end of book.

Indexes can be created or dropped with no effect on the data.

Creating Index

CREATE INDEX statement is used to create an index. In this statement we have to

mention name of the index, the table and column, and whether the index is in ascending or

descending order.

https://B-next.in Scanned by CamScanner

OP ams (Mu-BSc IT)

There are different types of indexes.

Syntax

For Example

CREATE [UNIQUE] INDEX index\_name

ON table\_name (column\_name1,[column\_name?, column\_name3,...]);

Lab Manual

res

ss

1. CREATE INDEX emp\_ind! on Employee(Employee\_name);

nm

CREATE INDEX emp\_ind2 ON Employee(Employee\_no, Employee\_name),

3. CREATE UNIQUE INDEX emp\_ind3 on Employee(Employee\_name);

Displaying Index : To display index information regarding table following query is used.

Syntax

Show index from table\_name;

For Example

Show index from Employee;

Ti Chwamp bn rengh mmr gB Ibi myagbene

+

| Employee” | 1]

NULL | NULL | NULL

| Employee | 1le

NULL | NULL | NULL

NULL | NULL

| Employee | 1le

NULL | NULL | NULL

4 rows ao ‘set (0. 03 sec)

Dropping Index

To drop index of a table following query is used.

ysql> create index emp\_ind2 on Emp

OR 1072 (42000): Key column ‘no

sql> create index emp\_ind2 on Employee(id,name) ;

uery OK, O rows affected (0.09 sec)

ecords: 0 Duplicates: 0 Warnings: 0

sql> create index emp\_ind3 on Employee (name);

ery OK, O rows affected (0.05 sec)

ecords: 0 Duplicates: 0 Warnings: 0

Sql> Show index from Eup loyee;

emp\_ indi |

| YES | BTREE

emp\_ind2 |

| Employee | 1 | emp\_ind2 |

NULL |

| YES | BTREE

\_ind3 |

| YES | BTREE

https://E-next.in

Toyee(no, name);

doesn't exist in table

sql> create Index emp\_ indi on Emp Toyee (name);

uery OK, O rows affected (0.09 sec)

ecords: 0 Duplicates: 0 Warnings: 0

i

}

;

| Table 7 Non. canique | Key\_name | Seq\_in- ‘index ] Colin pase | co '

pdinality | Sbaart | Packed | nut | Index\_type | Comment | ‘

Scanned by CamScanner

(Foams (Mu-Bsc 7)

Syntax

Drop index index\_name on table name;

SMC

For Example

L-31

———— — —

Lab Manual

————

SS

Drop index emp\_ind2 on Employee;

Sequence

Sequence :

requirement.

Creating sequence

For example

A se i i i

5 quence is a set of integers. Sequences are generated in order as per

e an .

quences are used to create unique values for the rows.

create table emp2( eno integer(3) auto\_increment, primary key(eno), ename

| varchar(20),sal integer(6));

"Hetearpbiim

Sai

ysqi> create table emp2(— “a

-> eno integer (3) auto\_increnent,

-> primary

ey(eno), a

~> ename varchar (20),

-> val integer (6)

=>):

Query OK, 0 rows affected (0.05 sec)

‘ i

mysql>

Now insert records in the table

For example

er

5

sql>

insert into emp2 values(null,’sam’,9000);

sqi> insert into emp2 values(null, sam’ ,9000);

uery OK, 1 row affected (0.00 sec)

| eno | ename | sal |

tetas poncenn pennant

1 m |

2 | teju |

— ae $¢------

rows in set (0.00 sec)

ee +

ites aaah ace ass

sqi> insert into emp2 values(null, ’teju’,10000); 0

ery OK, 1 row affected (0.00 sec)

9000 |

10000 |

eM OR NSA AIRY LLAL UO KARO AMMA HMA cht

iat

https://E-next.in

Scanned by CamScanner

ey DBMS (MU-BSC IT) L-32 Lab Manual

(b)

Here the column eno has auto increment values.

Synonym

Synonyms are basically alternative names for a table, view, sequence, procedure, stored

function, package etc. Synonyms provide both data independence and \_ location

transparency.

Creating Synonym in Oracle

Create synonym e for emp;

Controlling User Access

© Grant : to allow specified users to perform specified tasks.

o Revoke : to cancel previously granted or denied permissions.

Grant command

Syntax

GRANT <object privileges>

ON <object\_name>

TO <User\_Name>

[WITH GRANT OPTION]

Example

GRANT ALL

ON cust TO anand

WITH GRANT OPTION;

FRO IIOE LTT

ROEM CO hy yah we LUG itt

1 Cae ay feed

‘ Caan be mspemphS M bitensd ne

—\_

At

sql> grant select update ON cust TO anand; |

ery OK, 0 rows affected (0.03 sec) G

ysql> |

ee LB ONIDH LVM LEELA BEEBE

https://E-next.in Scanned by CamScanner

a

Granting one privilege to a user is as follows:

Hi ehaanptbenyngtvrysqS 1 abninongien soso: enees ssidictsetitie ia seeeecressnmassmsscnscttisiil ea

vor]

fen f @

ysq!> select \* from emp2;

----- tenn naan p anne nent

| eno | ename | sal |

----- terenn npn n-ne a4

} 1 sam | 9000 |

| 2 | teju | 10000 |

----- teannnnnt---n- = nt

2 rows in set (0.00 sec)

ysql> GRANT select ON emp2 TO sam;

uery OK, 0 rows affected (0.00 sec)

i eno i ename i sal |

pemenapann nnn apannnn ant

B

| 1 | sam | 9000 |

2 | tej | s0000 |

+ ana ede eer n ea pes ese

2 roval in set, (0. 00 sec)

ysql> GRANT select ON emp2 TO sam;

buery OK, 0 rows affected (0.00 sec)

bap GRANT select, insert,delete ON emp2 TO sam;

ery OK, 0 rows affected (0.00 sec)

———

Revoke command

Syntax

REVOKE <Object\_Privileges>

ON <Object\_Name> ae

FROM <User\_Name>.

Example

REVOKE select inser del

https://E-next.in Scanned by CamScanner

\& DBMS (MU-BSC IT) L-34 Lab Manual

‘etry ined nL ee ea ee 1a g

ysql> REVOKE select, insert, delete ON enp2 FROM Saf

uery OK, 0 rows affected (0.00 sec) U

6. Using SET operators, Date/Time Functlons, GROUP BY Seer ane

features) and advanced subqueries

(a) Using SET Operators

— The different Set Operators are as follows

o Union o Union All o Intersect o Minus

Consider following two tables product\_details and Sale\_details :

i 1001 1 endrive | 100 1 $00 i ee

Al 1002 | harddisk | 200 | 4000 | :

Al 1003 | headphone att | 15000 | é

| 1004 | DvD 1000 | a

1005 | speaker 600 2400 | Dh \

fbenwneee -+ qeonreae + 3

rows in set “0. 03 sec) “

“mysql> select \* from Sale setallss :

ppa enema pen enna nnn n nn penne nnn a tonnnene pracnnec nen anenn + at

| sale\_no i product\_id i quantity i price | customer\_name | :

+ oe ae + | einlembesrioe mm inetetadmanleleds oro sadm ia mc ed rere ner ii + ;

HI 2001 | 1001 | 50 | 900 | savni | }

| 2002 | 1004 | 10 | 1000 | savni | a3

| 2003 | 1003 | 120 | 15000 | savni | :

| 2004 | 1005 420 | 2400 | harsh | :

| 2005 | nz. 40 | 4000 | Akash | §

-+ -+ 4

5 :

Union

The union operator returns all distinct rows selected by either query

Syntax

Select column\_name from table\_1

Union

Select column\_name from table\_2 -

htips://E-next.in Scanned by CamScanner

For Example

write a query to retrieve ids of all the products even if they were sale or present in the

storage room.

Product\_id from product\_details

select

ynion

select Product\_id from sale\_details;

Teampthinatey gS atieinapee ERNIE TERETE TTT

mysql> Select Product\_id from product\_detai1s

-> union

-> select Product\_id from sale\_details;

is rows in set (0.00 sec) a

rvs ie

te 3

Union All

The Union All operator returns all rows selected by either query including duplicates.

Syntax

| Select column\_ \_name from: tables 1

Union all

Select column. : ren able. 2

For Example

- Write a query to retrieve ids of all the products even if they were sold or present in the

Storage room.

Select Product, \_id from product details A ee

Union ; all - Saas Ges

Select pidace Na from sale. details; ae

https://E-next.in Scanned by CamScanner

e

DBMS (MU-BSC IT) L-36

= eee

oaceinaansetncmemno

inept ceil

ites on tgp nah Pe me

ance sina sinensis darmcato

im Select Product\_id from product. details

=> Union all }

-> select feet from sale details;

ne ne

Intersect

The intersect operator returns only those rows which are common to both the queries

Syntax

Select column\_name from table\_1

intersect

Select column\_name from table\_2

For Example

Write a query to retrieve ids of all the sold products.

Select Product\_id from proniccees

intersect

select Product\_id from sale\_ details;

Minus

Minus operator displays the rows which are present in the first query but absent in the

second query, with no duplicates and data is arranged in ascending order by default.

Syntax

Select column\_name from yes 1

minus

Select column\_name from aca 2%

https://E-next.in Scanned by CamScanner

|

DBMS (MU-BSC |

gw us L-37 Lab Manual

For Example

Select Product\_id from product. details

minus

select Product\_id from sale\_details;

(b) Datetime Functions

Now0Q:

[ Now() returns the current date and time.

Example

The following SELECT statement:

SELECT NOW();

DateO

Extracts the date part of a date or date/time expression

Example

Select date('2014-5-1));

Select date(‘2014-5-1 11:00:00. oy

https://E-next.in Scanned by CamScanner

(FF pas (MU-BSC IT)

L-38

Lab Manual

Day()

In SQL Server (Transact-SQL), the DAY function returns the day of the month (a number

from I to 31) given a date value.

Example

Select day('2014-4-28 11:00:00.0’);

TimeQ

pe SELECT DAY(" 2013-4-28);

we ee ww ne eee wwe.

+

1 row in set (0.01 sec)

psa SELECT DAY('2015-4-28 ui: 00:00.0');

1 row in set (0.00 sec)

psa

Ata het tonne A WEEE ed Dire Il

The time function returns the time.

Example

uy Camytefemtd mag herein 1 [ols kM

Select time(’11:00:00.0’);

Or Select time(’2017-3-1 11:00:00.0);

f row in set (0.00 sec)

ars Select cing 13:00:00’);

https://E-next.in

Scanned by CamScanner

, Be

pams (MU-BSC IT Lab Manual

«e) Enhancement to the GROUP RY Caines

Group by clause

tROUP BY ¢

The GROUP BY clause is used in collaboration with the SELBCT etatement. It helps to

ange simil ‘ . i:

arrange similar data into groups. It i¢ alsa used with SQL functions to group the femur

from one oF more tables,

Syntax

LECT columnt, column2

FROM table\_name

WHERE [ conditions ]

GROUP BY column1, column2

For example

Display sum of salaries department wise.

[Select deptno,sum(sal) from emp group by deptno;

Output

DEPTNO | SUM(SAL)

30 9400

20 10875

10 8750

(d) Advanced Subqueries

In

select sub query:

syntax

SELECT column\_name [, column\_name }

FROM table1 [, table2 }

WHERE column\_name OPERATOR

IN (SELECT column\_name [, column\_name )

\_ FROM table1 [, table2 ]

https://E-next.in Scanned by CamScanner

app DBMS (MU-BSC IT) L-40 Lab Manual

SS

WHERE Employee\_no IN (SELECT Employee\_no

FROM Emp\_dept

WHERE Employee\_dept=’computer);

1) chovamng bntryef niysgGh 1 Mibin\enyrgl ene 2) 3 a

Iysqi> select ? from Employee; f

panne n npn tee een.

|! Employee\_no i Employee\_pame | | joining\_date i Salary 1

wt re ep wn ee eg ee eee eee -+

101 | Rajesh 1995-11-02 12000

102 | Swati 2015-01-07 20000

103 | Vedant 1999-05-03 30000

104 | Vedika 2005-02-10 52000

105 | Ankita 2015-03-15 50000

Power e nnn n ne npann----n-n---- ocr ree s cen nna- fama nnn +

5 rows in set 0. 00 sec)

Lvsar> select \* \* fron Emp\_dept;

101 1 computer |

102 | computer

103 | civil

104 | ETC Aa

105 | Electrical ;

5 rows in set . 00 sec)

= ;

bysql> select \* from Employee where Employee\_no IN(select Employee\_no from Emp,

prt where Emp loyee\_ dept=' computer yi

101 | Rajesh “Y 1995-11-02 | 12000 |

102 | Swati 2015-01-07 20000 :

Prromenn cnn cc nap acen anna enn nee Spo mon rn nnn nn nnpen nanan + 3

42 Tows in set 0. 03 sec) 4

Update sub query

Syntax

UPDATE table

SET column\_name = new\_value \_

[ WHERE OPERATOR [ VALUE ] ©

(SELECT COLUMN\_NAME

FROM TABLE\_NAME) |

[WHERE)] =

https://E-next.in Scanned by CamScanner

woes weet L441 Lab Manual

For example:

pPDATE Employee

SET SALARY = SALARY \* 0.50

WHERE Employee\_no IN (SELECT Em

WHERE Employee\_dept=’ ETC’);

ployee\_no FROM Emp\_dept

Tedeanplbn msehnyxgS1 Dbrimaine ne Ne aa ' soomee “ om ef MER ENeeLeenate tna tsesneceni-cerennerssspeteat roan econ VOT OMe

ysq|> UPDATE Employee [oh 9) :

> lf SALARY \* 0.50

> Employee\_no IN (SELECT Employee\_no FROM Emp\_dept

> WHERE Employee\_dept='ETC’): eee \_

uery OX, 1 row affected (0.02 sec)

Rows matched: 1 Changed: 1 Warnings: 0

‘mysqi> select \* from Employee;

poe wnn mene nafonen ~~.

ep wet ween eeeenenlLeneeaea=,

{ Emp loyee\_no a Employee \_nane | | joining date 7 salary | i

Pr erent serena pwn anew wen ne nL eee nen enn

+ ewe wee:

| 101 7 Rajesh 1 1995-11-02 | 12000 "|

102 | Swati 2015-01-07 | 20000

103 | Vedant 1999-05-03 30000 ig

104 | Vedika 2005-02-10 | 26000 ag

4| 105 | Ankita 2015-03-15 50000

tt-- nto--- a-poo-- no-t- +

45 rows in set (0.00 sec)

fhysql>

Delete sub query

DELETE FROM TABLE\_NAME |

[WHERE OPERATOR [VALUE]

IN (SELECT COLUMN\_NAME

FROM TABLE\_NAME)

| \_[WHERE)]}-

For example

| DELETE FROM Employee \_ a8 A Pe a

WHERE Employee\_no IN ster Empaye. no FROM M Emp\_ sept be

| WHERE Employee\_dept="civil’); \_ ot =

htps://E-next.in Scanned by CamScanner

ke DBMS (MU-BSC IT) L-42

ysql> DELETE FROM Employee

> WHERE Employee \_dept=" ao ye

Uery OK, 1 row affected (0.00 sec

ysql> elt \* fron Employee;

| Employee\_no I Employee\_name joining\_date | Salary |

Insert sub query

For example

~> WHERE Employee\_no IN (SELECT Employee\_no FROM Emp\_dept

101] Rajesh 1 1995- -11-02 12000

102 | Swati 2015-01-07 20000

104 | Vedika 2005-02-10 26000

~ Ankita 2015-03-15 50000 |

aw ennennennnpanennen-nnn--- mp enennnnnnnnnnnpenenne nny

rows in set “0. 00 sec)

i ql> es spesssnnsnecsseTilitl alls ittiiilil aA

SQL> Insert into EplOyee

Select \* from Emp\_dept

~ WHERE Employee\_ no IN (SELECT Employee ‘no , FROM Emp. dei

ql> Insert into Employee

-> Select \* from Emp\_dept

-> WHERE Employee\_no N (SELECT Employee\_no FROM Emp\_dept);

uery OK, 5 rows affected (0.00 sec)

Records: 5 Duplicates: 0 Warnings: 0

ql> select \* from Pay lovers

ih $------ $-Hn- nn -H4

{| Employee\_no ! Emp loyee\_name “joining\_date | Salary |

fi------=------ 4 --------- =p --- $-------= +

| 105 | Electrical | NULL | NULL |

| 104 | ETC | NULL | NULL |

il 103 | civil | NULL } NULL |

{I 102 | computer | NULL | NULL |

| 101 | computer ] NULL | NULL |

ip-------------4---~ <p + -4

https://E-next.in

Scanned by CamScanner

ESS ee” ee hhh

to DBMS (MU-BSC IT)

L-43 Lab Manual

Update sub query

PDATE table

U

seT column\_name = new value

[ WHERE OPERATOR [ VALUE ]

NOT IN (SELECT COLUMN\_NAME

FROM TABLE\_NAME)

[ WHERE) ]

fa el

For example:

UPDATE Employee

SET SALARY = SALARY \* 0.50

WHERE Employee\_no not IN (SELECT Employee\_no FROM Emp\_dept

La

WHERE employes. cepts ey:

101 | “Rajesh Y 1995- 11-02 72000 |

102 | swati 2015-01-07 20000

104 | Vedika 2005-02-10 26000

us Ankita 2015-03-15 50000

Pome n enna enn p wena anew nn aoe —}-- +

4 rows in set 0. 00 sec)

mysql> UPDATE Employee

; => SET SALARY = SALARY \* 0.50

~> WHERE Employee\_no not IN oa Emp loyee\_no FROM Emp\_dept

-> WHERE Employee\_i ept="ETC’);

Query OK, 3 rows affected ; 00 sec)

‘Rows matched: 3 Changed: 3 warnings: 0

Lysql> select \* from Employee;

| Emp loyee\_no i Emp loyee\_name joining\_date 1 “Salary i

101 | Rajesh | 1935- 11-02 6000 |

in Swati 2015-01-07 10000

Vedika 2005-02-10 26000

105 Ankita ~ | 2015-03-15 25000 |

+ >

Delete sub query

Syntax

A rons in set 0.0 86)

DELETE FROM TABLE\_NAME

[ WHERE OPERATOR [ VALUE ]

IN (SELECT. COLUMN NAME

FROM TABLE\_NAME)

[ WHERE) ]

https://E-next.in

Scanned by CamScanner

DBMS (MU-BSC IT) L-44 Lab Manual

== SSS

For example

DELETE FROM Employee

WHERE Employee\_no not IN (SELECT Employee\_no FROM Emp\_dept WHERE

Employee\_ depte'civil’);

“Tearing mf rtnretn Ey . [ole as

ys select fron Eaployee

wee ee wen] mew enw ewe ewwe) cemneenncescan)] ssaneeas +

Employee.no | i Emp oyee\_nane | joining\_date Salary |

ponnennnnnnnnnpanncnnnennannne fonnnnncnannnnpennnnn= “+

| ial | Rajesh 1995-11-02 | 6000

102 | Swati 2015-01-07 | 10000

Vedika | 2005-02-10 | 26000

105 | Ankita | 2015-03-15 | 25000 |

pom nnnn nnn nnanponnnnnnnnncnnn fromeceneeanne= pronnenes +

4 rows in set 0 00 sec)

hysql> DELETE FROM Employee C

-> WHERE Enployeeno not IN (SELECT Employee\_no FROM Enmp\_dept WHERE

Employee\_dept='civil’);

Query OK, 4 rows affected (0.00 sec)

os

ysql> select \* from Employee;

mpty set (0.00 sec)

Some

SOME compare a value to each value in a list or results from a query and evaluate to true

if the result of an inner query contains at least one row. SOME must match at least one row in

the subquery and must be preceded by comparison operators. Suppose using greater than (>)

with SOME means greater than at least one value same as it if we use less than (<) with

SOME means less than at least one value.

Syntax

SELECT {column\_name... | expression1 ]

FROM [table\_name]

WHERE expression2 comparison\_ operator {ALL 1 ANY | even ( slbqueiy )

For example

SELECT \* from Employee

WHERE no=SOME( SELECT no FROM Emp\_dept

WHERE depte'computer);,

htips://E-next.in Scanned by CamScanner

tae DBMS (MU-BSC IT) L-45 Lab Mans

erat et rota ,

{me rane Tei |

+

| 1 | sanjay | pune |

1 2] ram il

| 3 | samarth | | saxgli i|

3 rows in set (0. 00 sec) ,

jaysql> select \* fron Emp\_dept;

He---- ~4------------

{l no | dept |

‘\y----- ~t----------- -+

al 1 | Computer =|

q| 2 | civil |

4] 3 | Etc |

{| 4 | Electrical |

4 +

+

‘A rows in set (0.00 sec)

jaysql> SELECT \* from Employee

-> WHERE no=SOME( SELECT no FROM Emp\_dept

-> WHERE dept=" computer De

4] no 1 name city 1

Horse and n ance nto +

4 1 | sanjay pune |

----— 4$--------4}------ c

row in set (0. 00 va). iy:

hrysal a

For example

SELECT. \* from peinhe

- WHERE city= pune ye

SELECT \* ‘from emp?

7 WHERE city= pune)

SELECT A from ‘emp2

\_ WHERE c ‘city=' pune’ )

” WHERE Galan = SoHE SELECT salary FROM emp Vee

\_ WHERE Slay) < =SOME c SELECT Talay F FROM Menpa :

fe "WHERE E salary >=SOME ( SELECT salary FROM e emp1\_

https://E-next.in

Scanned by CamScanner

Lg

41 ehapapi been aya mek wee

-> empl a city='pune’);

Hm a mn penn ee een penn eee fooccenn +

| id | name city | salary |

---=-- tone nn enn g-------4--------

| 1 | Ramesh i DeThi i 2000 {

[tonne nt n------ $onnn-n- tan-nnnH- +

1 row in set (0.00 sec)

-> empl where city='pune');

totam eaten anna t------- teonncnon +

| id | name | city | salary |

t------ t-------- $o------ pocnann-- +

| 3 | komal pune 10000 |

| 2 | kajal pune 10000 |

| 1 | Ramesh ! Delhi 2000 |

Ht------ tenn nna -t-------$-------- +

3 rows in set (0. 00 sec)

-> enpt where a pune ai,

pune |. 10000 |

pune 10000 |

2 rows in set (0. 00 sec)

PYSG|> mm

+ —— + — +

rx

°

3,

All

a

Lab Manual

[ole tay

rn ary from +

mysql> Select \* from enp2 mere aon — Salary rom

mysql> Select \* from emp2 where salary<=some (select salary from

ysqi> Select \* from emp2 where salary>=some(select salary from

———

ALL is used to select all records of aSELECT STATEMENT. It compares a value to

every value in a list or results from a query. The ALL must be preceded by the comparison

operators and evaluates to TRUE if the query returns no rows. For example, ALL means

greater than every value, means greater than the maximum value. Suppose ALL (1, 2, 3)

means greater than 3.

Syntax

FROM [table\_name]

SELECT [column\_name...

| expression1 1 Sone

WHERE expression2 einpaneone operator {ALL ra ANY. | SOME} ( ie —

https://E-next.in Scanned by CamScanner

—————

DBMS (MU-BSC IT)

Lab Manual

Fo ———

rea : aes

Pee eal wee Sea salaryaliGelect salary from

[dre “Thy Faery |

[4 | names Delhi | 2000 1

1 row in set (0.01 sec) \*

pysql> Select \* from emp2 where salary<=all (select salary from

> cap where city="pune');

| id H name H city | salary i

Ht-----= $------- -t------- $onn----- +

| 3 | komal | pune | 10000 |

| 2 | kajal\_ | pune | 10000 |

| 1 | Ramesh | delhi | 2000 |

Ht-----— +-------- toncnen- $-------- +

3 rows in set (0.00 sec)

nysql> Select \* from emp2 where salary>=all(select salary from

-> empl where citys pune’);

it------ fononne- teccensten-o-- +

| id | name | city 1 salary |

konmal | pune | 10000

| kajal i pune | 10000

}------ fee -~-t------ $------- +

2 rows in set 0. 00 sec)

¢

wk el

7. PL/SQL Basics -

(a) Declaring Variables

Question : Displaying sum ot two numbers

Set serveroutput One

declare it,

v\_x number(5); -

-v\_y number(5);\_

-V\_sum pumber(9)

BEGIN fe

e Moe ae :

te XEV\_Y x

ine(‘Sum ills v. sum)

htips://E-next.in Scanned by CamScanner

—— SS

OUTPUT

Sum is 30

P/L SQL procedure successfully completed.

(b) Writing Executable Statements

Question : Displaying salary of a specific employee

Set serveroutput on;

declare

var\_sal number(6);

var\_emp\_no number(4);

BEGIN

var\_emp\_no:='7369';

SELECT sal

INTO var\_sal

FROM emp

WHERE empno = var\_emp\_no;

dbms\_output.put\_line(var\_sal);

end;

Se eee meee en ee nee oe anen aaa : - ee ewan n nee wee eee

at a a a

eww eee eee eee ee eeene

800 GP ees ge 7a

PL/SQL procedure successfully completed.

(c) Interacting with the Oracle Server

set serveroutput on;

declare

var\_x number(5);

var\_ y number(5);

var\_sum number(10);

var\_avg number(10);

begin

var\_x:=8&var\_x;

vary:=8var\_y; aa

dbms\_output.put\_ line(‘Sum =|] var\_x+var ys?

dbms\_output.put\_ line(‘average “ll var. aver yi

end; my

a

CET pBMs (Mu-BSC IT) L-48 Lab Manual

|

|

4

htips://E-next.in Scanned by CamScanner

SEE a Pals, SPE NE BE. DOO

(7 veMs (Mu-Bsc 17)

SSS

OUTPUT

Lab Manual

ee

Enter value for var\_x: 2

old 7: var\_x:=&var\_x;

new 7: Var\_Xi=2;

Enter value for var\_y: 2

old 8: var\_y:=8&var\_y;

Sum=4

average=2

| PL/SQL procedure successfully completed.

(d) Writing Control Structures

(i) Using if Statement

Set serveroutput on;

DECLARE

str1 VARCHAR(12);

srt2 VARCHAR(12);

BEGIN

str1 = ‘TECHMAX';

str2 := 'TECHMAX';

IF stri LIKE str2 THEN ,

DBMS\_OUTPUT.PUT\_LINE(str1 II" is same. Une Il st)

END IF;

END;

OUTPUT:

TECHMAX is same like TECHMAX

PL/SQL procedure successfully completed.

ee eee een eee

(ii) Using if -then-else statements

Set serveroutput Ong eee

declare Hey

x number(10);

y number(10);

| begin ©

https://E-next.in

Scanned by CamScanner

DBMS (MU-BSC IT) L-50

Lab Manual

X:=7782;

select sal into y from emp where empno=x;

if y>4500

then update emp set sal=(sal+2500) where empno=x;

elsif y>3500

then

update emp set sal=(sal+1500) where empno=x;

else

update emp set sal=(sal+500) where empno=x;

end if;

end;

/

OUTPUT

SQL> select \* from emp where empno=7782;

EMPNO ENAME JOB MGR HIREDATE SAL. #2 COMM

SQL> select \* from emp where empno=7782; \_

EMPNO ENAME JOB MGR HIREDATE SAL COMM

PL/SQL procedure successfully completed.

7782 CLARK MANAGER —7839.09-JUN-81 5348 s100

eam een ene ene een ne en ne eee sees eee cece meee seen es eee Stews meen een em maer ens enme

7782 CLARK MANAGER 7839 09-JUN-81 5159. 10.

(iii) Nested if -then-else statements

Set serveroutput on;

DECLARE

percent NUMERIC;

BEGIN

percent := '83';

IF percent >= 75 THEN

DBMS\_OUTPUT.PUT\_LINE('DISINCTION’);

ELSIF percent >= 60 AND percent <75 THEN

DBMS\_OUTPUT.PUT\_LINE('FIRST CLASS’);

ELSIF percent >= 50 AND.percent<60 THEN

https://E-next.in Scanned by CamScanner

—

a DBMS (MU-BSC IT) .

—=-=e-

L-51 Lab Manual

2

\_——

DBMS\_OUTPUT.PUT\_LINE('SECOND CLASS ');

ELSIF percent >= '40" AND percent <50 iB

DBMS\_OUTPUT.PUT\_LINE('PASS CLASS ‘)

ELSE

DBMS\_OUTPUT.PUT\_LINE('FAIL

END IF;

END;

.

,

bance ‘3

He:

OUTPUT

DISINCTION

| PL/SQL procedure successfully completed.

(iv) Using CASE - WHEN statement

SQL>set serveroutput on;

DECLARE

grade CHAR(1);

BEGIN |

grade :='A -

CASE grade - ; Se

WHEN 'O’ THEN DBMS\_OUTPUT.PUT\_LINE(‘Outstanding’);

WHEN 'A’ THEN DBMS\_OUTPUT.PUT\_LINE(‘Excellent’);

WHEN 'B' THEN DBMS\_ OUTPUT.PUT\_LINE('Good’);

WHEN 'C' THEN DBMS\_OUTPUT.PUT\_LINE('Satisfactory’);

WHEN 'F' THEN DBMS\_OUTPUT.PUT\_LINE(‘Fail);.

ELSE DBMS. OUTPUT.PUT\_LINE(‘Invalid grade’);

END CASE;

END;

/

OUTPUT

Excellent =

PL/SQL procedure successfully completed.

reel

https://E-next.in Scanned by CamScanner

[7 pams (mu-ssc 17) b62

grade CHAR(1);

BEGIN

grade :='A’;;

CASE grade

WHEN ‘0’ THEN DBMS\_OUTPUT.PUT\_LINE(‘Outstanding’);

WHEN ‘A’ THEN DBMS\_OUTPUT,PUT\_LINE('Excellent’);

WHEN 'B' THEN DBMS\_OUTPUT.PUT\_LINE('Good');

WHEN 'C' THEN DBMS\_OUTPUT,.PUT\_LINE(‘Satisfactory’);

WHEN 'F THEN DBMS\_OUTPUT.PUT\_LINE(‘Fail’);

ELSE DBMS\_OUTPUT.PUT\_LINE('Invalid grade’);

END CASE;

END;

/

OUTPUT

Excellent

PL/SQL procedure successfully completed.

(vi) Using For Loop

Set serveroutput on;

DECLARE

ans NUMBER(4);

BEGIN

ans := 5;

FOR num IN 1..10 LOOP os

DBMS\_OUTPUT.PUT\_LINE('S x’ || num ||'="|| ans);

ans := ans + 5; oe

END LOOP;

END;

/

OUTPUT

5 xi=5

5 x2=10

5 x3=15

5x4—20 °°

5 x6=3

\_

https://E-next.in Scanned by CamScanner

DBMS (MU-BSC IT) \_

Lab Manual

5 x7=35

5 x8=40

5 x9=45

5 x10=50

| PL/SQL procedure successfully completed.

(vii) Using Continue —When in Loop

SQL> BEGIN

FOR num IN 1.. 10 LOOP

CONTINUE WHEN MOD(num,2) != 0;

DBMS\_OUTPUT.PUT\_LINE(num);

END LOOP;

END;

he

OUTPUT

00 ans nm

10

PL/SQL procedure successfully completed.

8. Composite data types, cursors and exceptions.

(a) Working with Composite Data Types

‘SQL> DECLARE

TYPE t\_type IS TABLE OF NUMBER(10)

“INDEX BY BINARY\_ INTEGER;

var\_ \_tabt\_types.

var] indx NOMBER;

BEGIN:

<< twasing » >>

END LOOP loadin

Var tab: DELETE(!

https://E-next.in

Scanned by CamScanner

Qe? ppas (MU-BSC IT) LBA Lab Manual

ST See SS

—

Var\_indx := var\_tab.FIRST;

<< displaying >>

WHILE var\_indx IS NOT NULL LOOP

DBMS\_OUTPUT.PUT\_LINE(‘Displaying....' || var\_tab(var\_indx));

vat\_indx := var\_tab,NEXT(var\_indx);

END LOOP displaying;

END;

/

OUTPUT

Displaying...

Displaying...

Displaying...

Displaying...

Displaying...

Displaying....7

PL/SQL procedure successfully completed.

AR WNP

(b) Writing Explicit Cursors

(i) Using explicit cursors

SQL> DECLARE

var\_emp\_no char(4);

var\_emp\_namevarchar(10);

var\_emp\_sal number(8,2);

var\_emp\_addrvarchar(10);

cursor c\_emp is

select emp\_no, emp\_name, emp\_sal, emp\_addr from emp;

BEGIN

open c\_emp;

loop

fetch c\_emp into var\_emp\_no,var\_emp\_name,var\_emp\_sal,var\_emp\_. addr;

exit when c\_emp%notfound;

var\_new\_sal := var\_emp\_sal +5000;

insert into emp

values(var\_emp\_no,var\_emp\_name,var\_new\_sal,var\_emp\_addr);

end loop;

close c\_emp;

END;

/

https://B-next.in Scanned by CamScanner

-

° DBMS (MU-BSC IT)

oN ee

OUTPUT

SQL> select \* from emp;

EMPNO ENAME SAL ADDR

pesee nc ene Tam rme ners sence nnn AW.

4012 CLARK 9000 MUMBAI

PL/SQL procedure successfully completed.

(ii) Using loops in cursor

DECLARE Oe

cursor C1 is select \* frome emp where sal>2500 and job! president; a

xemp%orowtype;

BEGIN

open cl;

LOOP. .

fetch c1 into xX} pe

DBMS\_OUTPUT. PUT LINE(x. name);

exit when(cl%notfound); ca

if(x.job=" managers’ and x. sal>2500)

then update emp set sal=sal+50;

elsif(x.job='analyst' and x. sal>2500)

then update emp set sal=sal+70;\_\_

end if;

end loop; -

close ci;

END;

/

OUTPUT

JONES =.

BLAKE

SCOTT —

KING

FORD

FORD

https://E-next.in Scanned by CamScanner

Ty

ey, DBMS (MU-BSC IT) L-56 Lab Manual

———

(c) Handling Exceptions

SQL> CREATE OR REPLACE PROCEDURE new\_dept (d\_no IN

CHAR, d\_name IN VARCHAR2)

IS

BEGIN

INSERT INTO department (dept\_no, dept\_name) VALUES ( d\_no, d\_name );

EXCEPTION

WHEN DUP\_VAL\_ON\_INDEX THEN

raise\_application\_error (-25001,'Department No. already existing’);

WHEN OTHERS THEN

raise\_application\_error (-25002,'Error inserting new Department.’);

END;

/

OUTPUT

Case 1

exec new\_dept;

Department No. already existing.

Case 2

Exec new \_dept;

Error inserting new department.

9. Procedures and Functions

(a) Creating Procedures

Question 1 : Create a procedure that adds the details(empno,ename,job,sal) of newest

employee from table Emp into a table mewperson having structure as

(eno, firstname, lastname,design,salary).The last name of a person is to be passed to the

procedure as read only value.

Create or replace procedure newperson(Iname varchar2) — a

As :

Id emp.empno%type;

Fnameemp.ename%otype;

Des emp.job%type;

Salary emp. saltotypes

Begin

Select eno ename os sal into id

From emp. - ee .

frame, des salary

htips://E-next.in Scanned by CamScanner

A

wa DBMS (MU-BSC IT)

Go L-57 Lab Manual

ee

Where empno=(select max(empno) from emp);

Insert into person(eno,firstname,lastname design,salary)

values(id.fname,lname,des,satary) | aa

End newperson;

.

e

/

OUTPUT

Fxec newperson;

Select \* from person;

ENO FIRSTNAME LASTNAME -DESIGN. SALARY

4012 JHONNY DEP © HEAD 4023

Question 2 : Create a stored procedure namely raise that receives the employee number

and raise in salary as parameter. It then raises the salary of that employee in the table emp.

Create or replace procedure raise\_emp(emp1 integer,increase float)

Is :

Currentsalary float;

Salary exception;

Begin

Select sal into currentsalary from emp where empno=emp1;

If currentsalary is null then

Raise salary\_exception;

Else

Update emp

Set sal=sal+increase

Where empno=emp1;

End if;

Exception

When no\_data\_found then

Insert into audit Values(emp1,unknow employee’);

When salary\_exception then

Insert into audit values(emp1,‘nullsalary);

HENds oo oo ee

https://B-next.in Scanned by CamScanner

GT oes (MU-BSC IT) es)

OUTPUT

Exec raise;

Select \* from emp;

EMPNO = ENAME SAL ADDR

weno eee

Bee ncwe emee Mentind nnanncdua huuedianene auncnmane SEaeees Rem Ser

Ae RN Re

4012 JAY 13000 NEW YORK

4012 \_nullsalary

tails of customername and city

Question 3 : Create a stored procedure that provides the de

dit).To obtain these details

from table customers having structure (cid,name,city,status,cre

the customer number is passed to the procedure.

Create table cust01(cidinteger,name varchar2(20),status varchar2(1),credit integer); :

Insert into cust01 values(101,'aa','s1’,1000);,

Insert into cust01 values(102,'bb','s2,1001);

Insert into cust O1values(103,'cc','s3',1002);.

Create or replace procedure getcustomer ae

( custid in customer.cid%type,

Custname out cutomer.cname%otype,

Custcity out customer.city%type)

Is ; ODS SEES ete

Begin ; Pg OT OS

Select cname,city into custname,custcity.

From customers DOES SHE SR ig

Where cid=custid; ne

End getcustomer;

OUTPUT

exec getcustomer;,

oe

y

htips://E-next.in Scanned by CamScanner

SC CC se OO —————\_—\_

a DBMS (MU-BSC IT)

Lab Manual

a

b) Creating Functions

Question : Create a fur

iction that return doubled contributed amount (contamt) from table

donation if it is more than ]

\_———— 0 o.w. it returns tripled contributed amount for a given id no.

Create or replace function func\_cal1

(vidnodonation.idno%type) return number

Is

yamtdonation.contamt%type;

yreturndonation.contamt%type;

Begin

Select contamt into vamt

‘From donation

Where idno=vidno;

If vamt>10 then

return vreturn;

else Bathe

vreturn := vamt iy .

return vreturny mo

end if; :

end func Sala

OUTPUT

ed created

Case 1:

2000 ~

Case 2: ze as

3000.

(c) Managing Subprograms

Declare ee

X number:= =2;.

procedure abe j is

Begin

Insert into temp valus( in abe):

AENids. 0 oo :

Procedure par is

in a

https://E-next.in Scanned by CamScanner

Begin

Insert into temp values(x,’in main’);

abc;

pqr;

End;

/

(d) Creating Packages

Create or replace package emp\_salas

Procedure find\_sal(e\_no number);

CREATE OR REPLACE PACKAGE BODY emp\_sal AS

PROCEDURE find\_sal(e\_no number) IS

e\_sal number(10);

BEGIN

SELECT emp\_sal INTO e\_sal

FROM customers

WHERE emp\_no = e\_no;

dbms\_output.put\_line(‘Salary: ‘|| e\_sal);

END find\_sal;

END emp\_sal;

/

Lab Manual

ee DBMS (MU-BSC IT) 160

OUTPUT

Package body created.

40. Creating Database Triggers

(i) Trigger

create or replace trigger xyZ

before insert on fileA

for each row .

when(new.firl=10)

begin Se ae

insert into file\_B values(:new-fir, record’);

an

https://E-next.in

Scanned by CamScanner

\_

Y DBMS (MU-BSC IT) 64 Lab Manual

OO a a

QUTPUT

Insert into file\_ A values(12,’hello’);

\_

(ii) Trigger

create or replace trigger upd\_trig

before update of sal on emp

for each row

begin

if :new.sal<:old.sal then

raise\_application\_error(-20001,'salary can not be reduced’);

end if;

end;

/

OUTPUT

First of all create a copy of emp table and then use this query

Update emp set sal=sal-1000 where sal>4000;

(iii) Trigger

create or replace trigger psal ©

before delete or insert or update on emp

foreach row | a7 de

when(new.empno>0)

declare saldiff number(8,2);

begin ae

saldiff:= :new.sal-:old.sal;

dbms\_output.put\_line(:old.sal);

dbms\_ output.put\_line(:new.sal);

dbms\_output.put\_line(:new.sal-:old.sal);

/.

OUTPUT

Update emp set sal-sal-1000 where sal>4000;

PL/SQL Procedure Successfully executed

O00

htips://E-next.in Scanned by CamScanner