



4.Write An SQL Query To Fetch “FIRST\_NAME” From Worker Table Using The Alias Name As <WORKER\_NAME>.

Ans. Select first\_name||last\_name as worker\_name

from worker;

1. Write An SQL Query To Fetch “FIRST\_NAME” From Worker Table In Upper Case.

Ans. Select upper(first\_name)

From worker;

1. Write An SQL Query To Fetch Unique Values Of DEPARTMENT From Worker Table.

Ans. select unique department

from worker;

1. Write An SQL Query To Find The Position Of The Alphabet (‘A’) In The First Name Column ‘Amitabh’ From Worker Table.

select instr(first\_name,'A')

from worker

where first\_name='Amitabh';

1. Write An SQL Query To Print The First Three Characters Of  FIRST\_NAME From Worker Table.

Ans. select substr(first\_name,0,3)

from worker;

1. Write An SQL Query To Print The FIRST\_NAME From Worker Table After Removing White Spaces From The Right Side.

Ans. select rtrim (first\_name)

from worker;

1. Write An SQL Query To Print The DEPARTMENT From Worker Table After Removing White Spaces From The Left Side.

Asn. select ltrim (department)

from worker;

1. Write An SQL Query That Fetches The Unique Values Of DEPARTMENT From Worker Table And Prints Its Length.

Ans. select length(unique(department) )

from worker;

12.Write An SQL Query To Print The FIRST\_NAME From Worker Table After Replacing ‘A’ With ‘a’.

Ans. select replace(first\_name,'A','a')

from worker;

1. Write An SQL Query To Print The FIRST\_NAME And LAST\_NAME From Worker Table Into A Single Column COMPLETE\_NAME. A Space Char Should Separate Them.

select first\_name||' '||last\_name as complete\_name

from worker;

1. Write An SQL Query To Print All Worker Details From The Worker Table Order By FIRST\_NAME Ascending.

Select \*

From worker

Order by first\_name asc;

1. Write An SQL Query To Print All Worker Details From The Worker Table Order By FIRST\_NAME Ascending And DEPARTMENT Descending.

Ans. Select \*

From worker

Order by first\_name asc,department dsc;

1. Write An SQL Query To Print Details For Workers With The First Name As “Vipul” And “Satish” From Worker Table.

Ans. select \*

from worker

where first\_name='Vipul'

or first\_name='Satish';

1. Write An SQL Query To Print Details Of Workers Excluding First Names, “Vipul” And “Satish” From Worker Table.

Ans. select \*

from worker

where first\_name not'Vipul'

or first\_name not 'Satish';

1. Write An SQL Query To Print Details Of Workers With DEPARTMENT Name As “Admin”.

Ans. select \*

from worker

where department= 'Admin';

1. Write An SQL Query To Print Details Of The Workers Whose FIRST\_NAME Contains ‘A’.

Ans. select \*

from worker

where First\_name like '%A%';

1. .Write An SQL Query To Print Details Of The Workers Whose FIRST\_NAME Ends With ‘A’.

Ans. select \*

from worker

where First\_name like '%A';

1. Write An SQL Query To Print Details Of The Workers Whose FIRST\_NAME Ends With ‘H’ And Contains Six Alphabets.

Ans. select first\_name

from worker

where first\_name like '%H' and

length(first\_name)=6;

1. Start the executable section with the BEGIN keyword and include a SELECT statement to retrieve the maximum department\_id from the departments table.

Ans. set serveroutput on

BEGIN

select max(department\_id) into v\_max\_deptno from departments;

1. Write a PL/SQL block to show a reserved word can be used as a user-define identifier.

Ans. DECLARE

"WORLD" varchar2(20) := 'world';

"DECLARE" varchar2(20) := 'declare';

BEGIN

DBMS\_Output.Put\_Line(World);

DBMS\_Output.Put\_Line(DECLARE);

end;

/

1. Write PL/SQL blocks to show the scope and visibility of local and global identifiers.

Ans. DECLARE

var\_a INTEGER;

var\_b REAL; -- Scope of var\_b is REAL

BEGIN

var\_a:=5;

var\_b:=10.25;

-- Visible: var\_a (INTEGER), var\_b (REAL)

DBMS\_OUTPUT.PUT\_LINE('In the Outer Block');

DBMS\_OUTPUT.PUT\_LINE('var\_a = ' || var\_a);

DBMS\_OUTPUT.PUT\_LINE('var\_b = ' || var\_b);

1. Write a PL/SQL block to adjust the salary of the employee whose ID 122.

Ans.

DECLARE

salary\_of\_emp NUMBER(8,2);

PROCEDURE approx\_salary (

emp NUMBER,

empsal IN OUT NUMBER,

addless NUMBER

) IS

BEGIN

empsal := empsal + addless;

END;

BEGIN

SELECT salary INTO salary\_of\_emp

FROM employees

WHERE employee\_id = 122;

DBMS\_OUTPUT.PUT\_LINE

('Before invoking procedure, salary\_of\_emp: ' || salary\_of\_emp);

approx\_salary (100, salary\_of\_emp, 1000);

DBMS\_OUTPUT.PUT\_LINE

('After invoking procedure, salary\_of\_emp: ' || salary\_of\_emp);

END;

/