# **Implementing Row-Level Security (RLS) in Oracle**

This document outlines the steps to implement Row-Level Security (RLS) in Oracle for the user PRACTICE1. It includes creating a new user, setting up an EMP table with sample data, defining the necessary context, procedure, and function, and applying the RLS policy. Testing steps are also included to validate the implementation.

**Step 1: Create the User** 

Create the user PRACTICE1 and grant the required privileges.

**CREATE USER PRACTICE1 IDENTIFIED BY practice1;** 

GRANT CONNECT, RESOURCE TO PRACTICE1;

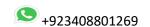
ALTER USER PRACTICE1 QUOTA UNLIMITED ON USERS;

GRANT CREATE SESSION, CREATE TABLE, CREATE PROCEDURE,

CREATE VIEW, CREATE TRIGGER TO PRACTICE1;

GRANT EXECUTE ON DBMS\_RLS TO PRACTICE1;







## **Step 2: Create the EMP Table**

Create the EMP table under the PRACTICE1 schema.

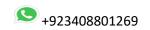
```
CREATE TABLE EMP (
          NUMBER(4) PRIMARY KEY,
 EMPNO
 ENAME
          VARCHAR2(20),
 JOB
        VARCHAR2(20),
 MGR
         NUMBER(4),
 HIREDATE DATE,
        NUMBER(10, 2),
 SAL
 COMM
          NUMBER(10, 2),
 DEPTNO
          NUMBER(4),
 USERNAME VARCHAR2(20)
);
```

# **Step 3: Insert Sample Data**

Populate the EMP table with sample records.

INSERT INTO EMP VALUES (222, 'ANNAR', 'MANAGER', NULL, SYSDATE, 3000, NULL, 10, 'ALI');

INSERT INTO EMP VALUES (7788, 'SCOTT', 'ANALYST', 7566, SYSDATE, 3000, NULL, 20, 'DANISH');



INSERT INTO EMP VALUES (7698, 'BLAKE', 'MANAGER', 7839, SYSDATE, 2850, NULL, 30, 'SYSTEM');

INSERT INTO EMP VALUES (7566, 'JONES', 'MANAGER', 7839, SYSDATE, 2975, NULL, 20, 'DANISH');

INSERT INTO EMP VALUES (7839, 'KING', 'PRESIDENT', NULL, SYSDATE, 5000, NULL, 10, 'ALI');

INSERT INTO EMP VALUES (7499, 'ALLEN', 'SALESMAN', 7698, SYSDATE, 1900, 300, 30, 'SYSTEM');

INSERT INTO EMP VALUES (7782, 'CLARK', 'MANAGER', 7839, SYSDATE, 2450, NULL, 10, 'ALI');

INSERT INTO EMP VALUES (7521, 'WARD', 'SALESMAN', 7698, SYSDATE, 1250, 500, 30, 'SYSTEM');

INSERT INTO EMP VALUES (7934, 'MILLER', 'CLERK', 7782, SYSDATE, 1300, NULL, 10, 'ALI');

INSERT INTO EMP VALUES (7902, 'FORD', 'ANALYST', 7566, SYSDATE, 3000, NULL, 20, 'DANISH');

INSERT INTO EMP VALUES (7654, 'MARTIN', 'SALESMAN', 7698, SYSDATE, 1250, 1400, 30, 'SYSTEM');

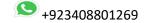
## **Step 4: Create a Context**

Define a context named TEST that will store user-specific attributes.

### create context test using set\_deptno;

SQL>
SQL> create context test using set\_deptno;
Context created.







#### **Step 5: Define a Procedure to Set Context**

Create a procedure to set the context for the TEST namespace.

```
create or replace procedure set_deptno(p1 varchar2) as
begin
```

```
dbms_session.set_context(
    namespace => 'test',
    attribute => 'username',
    value => p1
  );
end;
SQL> create or replace procedure set deptno(p1 varchar2) as
  2 begin
  3 dbms session.set context(
  4 namespace => 'test',
  5 attribute => 'username',
  6 value => p1
  7);
  8 end;
Procedure created.
```



#### **Step 6: Define a Function for RLS Predicate**

Create a function that defines the RLS predicate.

```
create or replace function dept_predicate(schema_name in varchar2,
object_name in varchar2)
return varchar2
is
  lv_predicate varchar2(1000);
begin
  lv_predicate := 'username = sys_context(''test'', ''username'')';
  return lv_predicate;
exception
  when others then
    raise_application_error(-20001, 'Error in dept_predicate: ' || sqlerrm);
end;
Step 7: Apply the RLS Policy
Attach the RLS policy to the EMP table.
BEGIN
  DBMS_RLS.ADD_POLICY(
    object_schema => 'PRACTICE1',
```

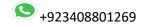






```
object_name => 'EMP',
    policy_name => 'EMP_CL_POLICY',
    function_schema => 'PRACTICE1',
    policy_function => 'dept_predicate'
 );
END;
Step 8: Test the Implementation
Set the Context for ALI:
begin
  set_deptno('ALI');
end;
/
SQL> begin
       set deptno('ALI');
  3 end;
PL/SQL procedure successfully completed.
cnı ֊
Verify the Context Setting:
```

## SELECT SYS\_CONTEXT('TEST', 'USERNAME') FROM DUAL;



#### **OUTPUT:**

# SYS\_CONTEXT('TEST', 'USERNAME')

-----

#### ALI

```
SQL> SELECT SYS_CONTEXT('TEST', 'USERNAME') FROM DUAL;

SYS_CONTEXT('TEST','USERNAME')

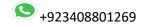
ALI

SOL>
```

# Query the EMP Table:

# **SELECT \* FROM EMP;**

SQL> select * from emp;						
	EMPNO	ENAME		10B	MGR	HIREDATE
	SAL	CO	MM DEPTNO			
	222 3000	ANNAR		1ANAGER ALI		10-JAN-25
	7839 5000	KING	_	PRESIDENT ALI		10-JAN-25
	7782 2450	CLARK		1ANAGER ALI	7839	10-JAN-25
	EMPNO	ENAME	:	JOB	MGR	HIREDATE
	SAL	CO	MM DEPTNO	USERNAME		
	7934 1300	MILLER		CLERK	7782	10-JAN-25





## Change the Context to DANISH:

```
begin
  set_deptno('DANISH');
end;
/
SQL> begin
      set_deptno('DANISH');
 3 end;
PL/SQL procedure successfully completed.
Verify the Context Setting:
SELECT SYS_CONTEXT('TEST', 'USERNAME') FROM DUAL;
OUTPUT:
SYS_CONTEXT('TEST', 'USERNAME')
DANISH
SELECT * FROM EMP;
OUTPUT:
```





#### SQL> select \* from emp;

EMPNO	ENAME	JOB	MGR HIREDATE
SAL	СОММ	DEPTNO USERNAME	_
7788 3000	SCOTT	ANALYST 20 DANISH	7566 10-JAN-25
7566 2975	JONES	MANAGER 20 DANISH	7839 10-JAN-25
7902 3000	FORD	ANALYST 20 DANISH	7566 10-JAN-25

