Name: Vinayak Kumar Singh **Subject:** DBMS Lab
Register No: 23MCA1030 **Date**:21-09-2023

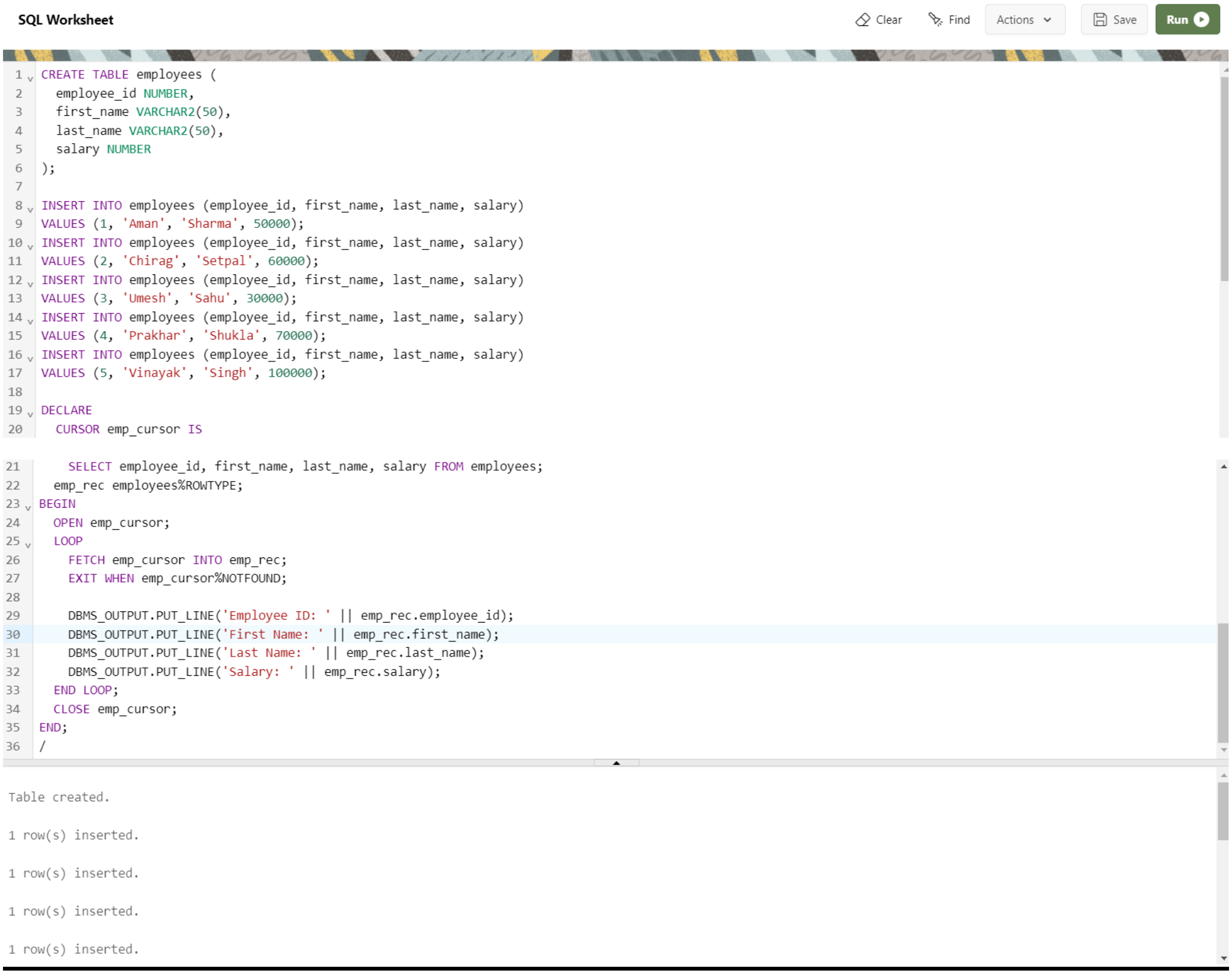
Exercise 6: PL SQL PROGRAMS

Cursors

```
CREATE TABLE employees (
  employee_id NUMBER,
  first_name VARCHAR2(50),
  last_name VARCHAR2(50),
  salary NUMBER
);


INSERT INTO employees (employee_id, first_name, last_name,
salary)
VALUES (1, 'Aman', 'Sharma', 50000);
INSERT INTO employees (employee_id, first_name, last_name,
salary)
VALUES (2, 'Chirag', 'Setpal', 60000);
INSERT INTO employees (employee_id, first_name, last_name,
salary)
VALUES (3, 'Umesh', 'Sahu', 30000);
INSERT INTO employees (employee_id, first_name, last_name,
salary)
VALUES (4, 'Prakhar', 'Shukla', 70000);
INSERT INTO employees (employee_id, first_name, last_name,
salary)
VALUES (5, 'Vinayak', 'Singh', 100000);


DECLARE
```

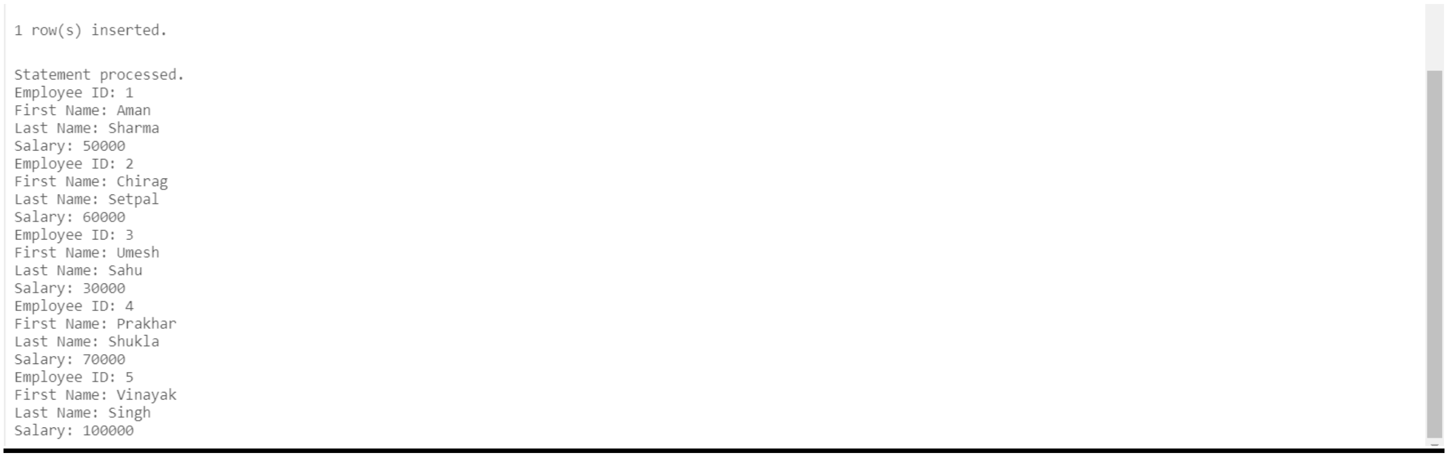
```
  CURSOR emp_cursor IS
    SELECT employee_id, first_name, last_name, salary FROM employees;
  emp_rec employees%ROWTYPE;
BEGIN
  OPEN emp_cursor;
  LOOP
    FETCH emp_cursor INTO emp_rec;
    EXIT WHEN emp_cursor%NOTFOUND;

    DBMS_OUTPUT.PUT_LINE('Employee ID: ' || emp_rec.employee_id);
    DBMS_OUTPUT.PUT_LINE('First Name: ' || emp_rec.first_name);
    DBMS_OUTPUT.PUT_LINE('Last Name: ' || emp_rec.last_name);
    DBMS_OUTPUT.PUT_LINE('Salary: ' || emp_rec.salary);
  END LOOP;
  CLOSE emp_cursor;
END;
/
```

Output:



```sql
CREATE TABLE employees (
  employee_id NUMBER,
  first_name VARCHAR2(50),
  last_name VARCHAR2(50),
  salary NUMBER
);

INSERT INTO employees (employee_id, first_name, last_name, salary)
VALUES (1, 'Aman', 'Sharma', 50000);
INSERT INTO employees (employee_id, first_name, last_name, salary)
VALUES (2, 'Chirag', 'Setpal', 60000);
INSERT INTO employees (employee_id, first_name, last_name, salary)
VALUES (3, 'Umesh', 'Sahu', 30000);
INSERT INTO employees (employee_id, first_name, last_name, salary)
VALUES (4, 'Prakhar', 'Shukla', 70000);
INSERT INTO employees (employee_id, first_name, last_name, salary)
VALUES (5, 'Vinayak', 'Singh', 100000);

DECLARE
  CURSOR emp_cursor IS
    SELECT employee_id, first_name, last_name, salary FROM employees;
  emp_rec employees%ROWTYPE;
BEGIN
  OPEN emp_cursor;
  LOOP
    FETCH emp_cursor INTO emp_rec;
    EXIT WHEN emp_cursor%NOTFOUND;

    DBMS_OUTPUT.PUT_LINE('Employee ID: ' || emp_rec.employee_id);
    DBMS_OUTPUT.PUT_LINE('First Name: ' || emp_rec.first_name);
    DBMS_OUTPUT.PUT_LINE('Last Name: ' || emp_rec.last_name);
    DBMS_OUTPUT.PUT_LINE('Salary: ' || emp_rec.salary);
  END LOOP;
  CLOSE emp_cursor;
END;
/
```

```
Table created.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.
```



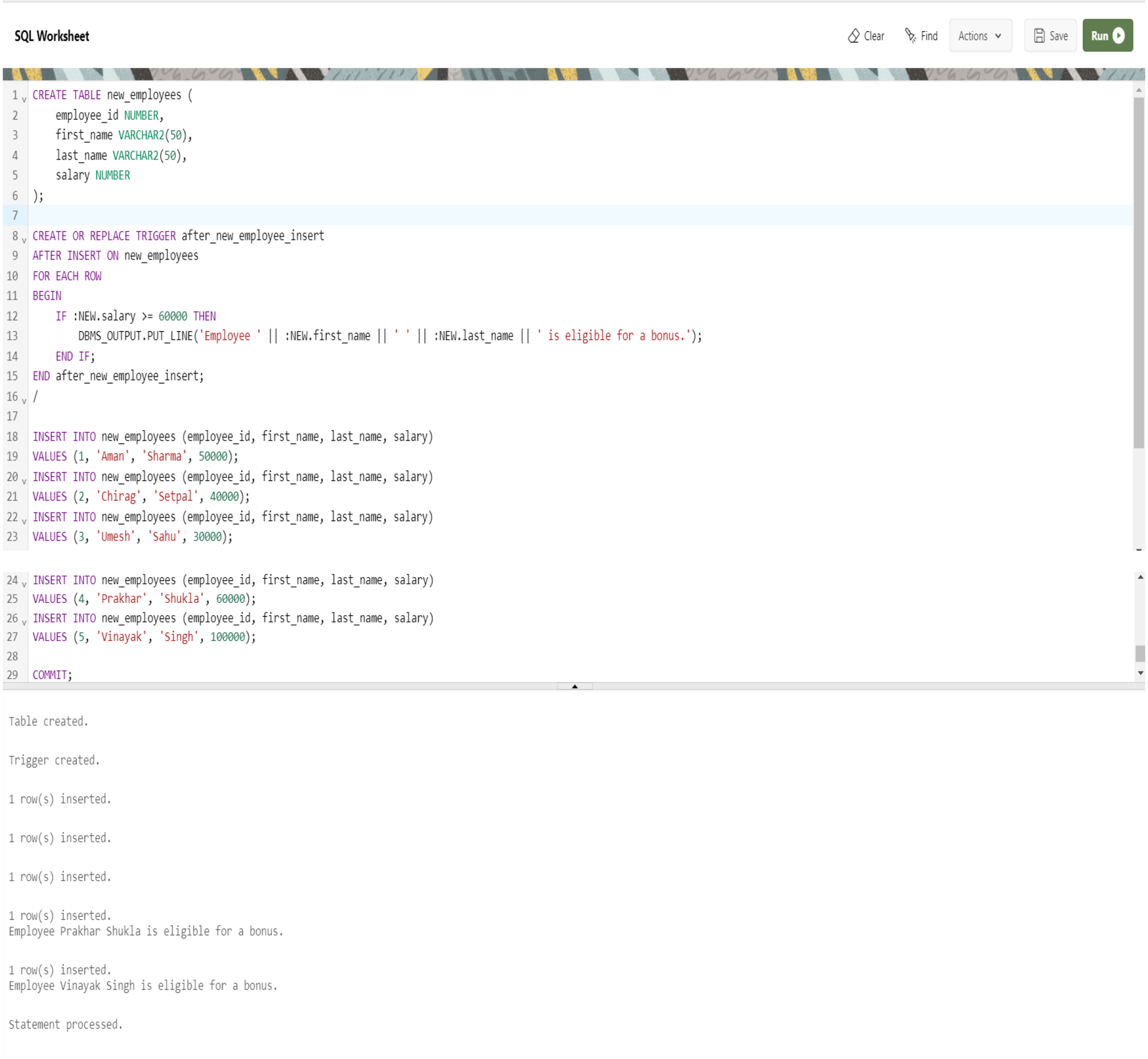
```
1 row(s) inserted.

Statement processed.
Employee ID: 1
First Name: Aman
Last Name: Sharma
Salary: 50000
Employee ID: 2
First Name: Chirag
Last Name: Setpal
Salary: 60000
Employee ID: 3
First Name: Umesh
Last Name: Sahu
Salary: 30000
Employee ID: 4
First Name: Prakhar
Last Name: Shukla
Salary: 70000
Employee ID: 5
First Name: Vinayak
Last Name: Singh
Salary: 100000
```

Triggers

```
CREATE TABLE new_employees (
    employee_id NUMBER,
    first_name VARCHAR2(50),
    last_name VARCHAR2(50),
    salary NUMBER
);

CREATE OR REPLACE TRIGGER after_new_employee_insert
AFTER INSERT ON new_employees
FOR EACH ROW
BEGIN
    IF :NEW.salary >= 60000 THEN
        DBMS_OUTPUT.PUT_LINE('Employee ' || :NEW.first_name ||
' ' || :NEW.last_name || ' is eligible for a bonus.');
    END IF;
END after_new_employee_insert;
/

INSERT INTO new_employees (employee_id, first_name,
last_name, salary)
VALUES (1, 'Aman', 'Sharma', 50000);
INSERT INTO new_employees (employee_id, first_name,
last_name, salary)
VALUES (2, 'Chirag', 'Setpal', 40000);
```

```
INSERT INTO new_employees (employee_id, first_name,
last_name, salary)

VALUES (3, 'Umesh', 'Sahu', 30000);

INSERT INTO new_employees (employee_id, first_name,
last_name, salary)

VALUES (4, 'Prakhar', 'Shukla', 60000);

INSERT INTO new_employees (employee_id, first_name,
last_name, salary)

VALUES (5, 'Vinayak', 'Singh', 100000);


COMMIT;
```

Output:



```sql
CREATE TABLE new_employees (
    employee_id NUMBER,
    first_name VARCHAR2(50),
    last_name VARCHAR2(50),
    salary NUMBER
);

CREATE OR REPLACE TRIGGER after_new_employee_insert
AFTER INSERT ON new_employees
FOR EACH ROW
BEGIN
    IF :NEW.salary >= 60000 THEN
        DBMS_OUTPUT.PUT_LINE('Employee ' || :NEW.first_name || ' ' || :NEW.last_name || ' is eligible for a bonus.');
    END IF;
END after_new_employee_insert;
/

INSERT INTO new_employees (employee_id, first_name, last_name, salary)
VALUES (1, 'Aman', 'Sharma', 50000);
INSERT INTO new_employees (employee_id, first_name, last_name, salary)
VALUES (2, 'Chirag', 'Setpal', 40000);
INSERT INTO new_employees (employee_id, first_name, last_name, salary)
VALUES (3, 'Umesh', 'Sahu', 30000);

INSERT INTO new_employees (employee_id, first_name, last_name, salary)
VALUES (4, 'Prakhar', 'Shukla', 60000);
INSERT INTO new_employees (employee_id, first_name, last_name, salary)
VALUES (5, 'Vinayak', 'Singh', 100000);

COMMIT;
```

```
Table created.

Trigger created.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.
Employee Prakhar Shukla is eligible for a bonus.

1 row(s) inserted.
Employee Vinayak Singh is eligible for a bonus.

Statement processed.
```

END