-- Create the emp table

CREATE TABLE emp (

emp\_no NUMBER PRIMARY KEY,

salary NUMBER

);

-- Create the increment\_salary table

CREATE TABLE increment\_salary (

emp\_no NUMBER,

salary NUMBER,

increment\_date DATE DEFAULT SYSDATE

);

-- Insert sample data into the emp table

INSERT INTO emp (emp\_no, salary) VALUES (1, 5000);

INSERT INTO emp (emp\_no, salary) VALUES (2, 6000);

INSERT INTO emp (emp\_no, salary) VALUES (3, 4500);

INSERT INTO emp (emp\_no, salary) VALUES (4, 7000);

INSERT INTO emp (emp\_no, salary) VALUES (5, 4000);

-- PL/SQL Block to update salary and insert into increment\_salary table

DECLARE

avg\_salary NUMBER;

BEGIN

-- Calculate the average salary of the organization

SELECT AVG(salary) INTO avg\_salary FROM emp;

-- Loop through all employees who have a salary less than the average salary

FOR emp\_record IN (SELECT emp\_no, salary FROM emp WHERE salary < avg\_salary) LOOP

-- Update the salary by 10%

UPDATE emp

SET salary = salary \* 1.10

WHERE emp\_no = emp\_record.emp\_no;

-- Insert record into increment\_salary table to track the increment

INSERT INTO increment\_salary (emp\_no, salary)

VALUES (emp\_record.emp\_no, emp\_record.salary \* 1.10);

END LOOP;

-- Commit the transaction

COMMIT;

END;

-- Query to check the updated salaries and the increment records

SELECT \* FROM emp;

SELECT \* FROM increment\_salary;