**Name:    Yosief Hailemariam Abraham**

```
CREATE DATABASE ass3;
USE ass3;
```

Q1.

```
   CREATE TABLE Dept (DId INT NOT NULL AUTO_INCREMENT,
                 Floor_Number INT NOT NULL,Budget REAL NOT NULL,
                 CONSTRAINT PK_Dept_DId PRIMARY KEY (DId));

   CREATE TABLE Emp ( EId INT NOT NULL AUTO_INCREMENT, Salary INT NOT NULL,
                      AGE INT NOT NULL, DId INT NOT NULL, CONSTRAINT Pk_Emp_EId
PRIMARY KEY (EId), CONSTRAINT Fk_Dept_DId FOREIGN KEY (DId) REFERENCES Dept
(DId)
);
```

Q2.

```
#2
-----Dept Records
DELIMITER $$
CREATE PROCEDURE InsertDept()
        BEGIN
                DECLARE i INT;
                DECLARE NumRows INT;
                DECLARE Floor_no INT;
                SET i = 1;
                SET NumRows=5;
                START TRANSACTION;
                WHILE i <= NumRows DO
                     SET Floor_no=( FLOOR(RAND()*(5-1+1)+1));
                        if not exists (select * from Dept where Floor_number = Floor_no) then
                             INSERT INTO Dept (Floor_number, Budget) values
(Floor_no,(CEIL(RAND() * (100 - 20+1) + 20)));
                            SET i = i + 1;
                     end if;
                END WHILE;
                COMMIT;
        END$$
DELIMITER ;
CALL InsertDept();
select * from Dept;
```

...Emp Records

```
DELIMITER $$
CREATE PROCEDURE InsertEmp()
	BEGIN
		DECLARE i INT;
		DECLARE NumRows INT;
		SET i = 1;
		SET NumRows=300000;
		START TRANSACTION;
		WHILE i <= NumRows DO
			INSERT INTO emp (Salary, AGE,Did) values
(( FLOOR(RAND()*(30-15+1)+15)),( FLOOR(RAND()*(60-18+1)+18)),( FLOOR(RAND()*(
5-1+1)+1)));
			SET i = i + 1;
		END WHILE;
		COMMIT;
	END$$
DELIMITER ;
CALL InsertEmp();
select * from emp;
```

Q3.

3.1

```
	SELECT COUNT(EId) AS Emp45	FROM Emp
	WHERE Age = 45;----7028
```

3.2

```
	SELECT COUNT(EId) AS Salary40 FROM Emp
	WHERE Salary > 40; ----- 0
```

3.3

```
	      SELECT count(EId) FROM Emp e
	      INNER JOIN Dept d ON
	      e.Did = d.DId
	      where d.Did = 1;----60248

	SELECT count(EId) FROM Emp e
	INNER JOIN Dept d ON
	e.Did = d.DId
	where d.Did = 2;----59825

	SELECT count(EId) FROM Emp e
	INNER JOIN Dept d ON
	e.Did = d.DId
	where d.Did = 3;----59703

	SELECT count(EId) FROM Emp e
```

```
INNER JOIN Dept d ON
e.Did = d.DId
where d.Did = 4;----59742

SELECT count(EId) FROM Emp e
INNER JOIN Dept d ON
e.Did = d.DId
where d.Did = 5;----60482
```

3.4 SELECT (SUM(Budget)/5) FROM Dept;---- 64.2

3.5

```
SELECT COUNT(EId) AS Emp33B45 FROM Emp
WHERE AGE BETWEEN 33 AND 45 AND Salary > 20;---- 56832
```

3.6

```
SELECT MAX(Salary) FROM    Emp AS e
INNER JOIN Dept AS d
ON e.DId = d.DId
WHERE d.Floor_Number = 5;----30
```

Q4.

Before Indexing

---query Q3.1
Execution time 0.25577325

---query Q3.2
Execution time 0.27405725

---query Q3.3
a. Execution time 0.16916000
b. Execution time 0.16863725
c. Execution time 0.11925875
d. Execution time 0.16606750
e. Execution time 0.11284450

---query Q3.4
Execution time 0.00157025

---query Q3.5
Execution time 0.32089900

---query Q3.6
Execution time 4.71282200

After Indexing
Adding and showing indexing
--------------------------------------------------

```
     ALTER TABLE Emp ADD INDEX index_age (Age);
     ALTER TABLE Emp ADD INDEX index_salary (Salary);
     ALTER TABLE Dept ADD INDEX index_budget (Budget);
     ALTER TABLE Dept ADD INDEX index_floor (Floor_Number);

     SHOW INDEX FROM Emp;
     SHOW INDEX FROM Dept;
----------------------------------------------------
---query Q3.1
    Execution time 0.01628625

---query Q3.2
    Execution time 0.00856325

---query Q3.3
   a. Execution time 0.05211850
   b. Execution time 0.06815950
   c. Execution time 0.06150575
   d. Execution time 0.04748900
   e. Execution time 0.04302500

---query Q3.4
    Execution time 0.00047075

---query Q3.5
   Execution time 0.22185425

---query Q3.6
   Execution time 4.81701325
```

NB. There seems to be improvement in execution time in all the queries except for query in Q3.6

Q5.

Q5.1

```
 DELIMITER $$
   CREATE procedure RecruitEmployee(IN floor_no int,IN salary int,IN age int )
    BEGIN
        DECLARE DeptId INT;
        START TRANSACTION;
        SET DeptId    = (Select DId from Dept where floor_number = floor_no);
            INSERT INTO emp (Salary, AGE, DId) values (salary,age, DeptId);
       COMMIT;
    END$$
   DELIMITER ;

   CALL RecruitEmployee(2, 20, 40);
```

Q5.2

```
DELIMITER $$
CREATE procedure ReduceBudget(IN floor_no int)
    BEGIN
        declare new_Budget real;
        START TRANSACTION;
        Set new_Budget = (SELECT budget from Dept where floor_number =
        floor_no)-(0.2*(SELECT                  budget from Dept where floor_number =
        floor_no));
        UPDATE Dept
        SET Budget = new_Budget
        WHERE Floor_Number = floor_no;
        COMMIT;
  END$$
DELIMITER ;
```

Q5.3

```
DELIMITER $$
CREATE procedure IncreaseSalary(IN Budget Real)
        BEGIN
                DECLARE empID INT;
                DECLARE new_Salary INT;
                DECLARE finished INTEGER DEFAULT 0;
                DECLARE curEmpID CURSOR FOR SELECT     e.Eid FROM Emp e INNER JOIN
                Dept d ON d.DId = e.DId WHERE d.BUdget > Budget;


              DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished = 1;

              START TRANSACTION;
                OPEN curEmpID;

                  read_loop: LOOP
                          FETCH curEmpID INTO empID;
                          IF finished THEN
                                    LEAVE read_loop;
                       END IF;
                          set new_Salary= (SELECT Salary from Emp where Eid = 2)+
                          ((0.1)*(SELECT Salary from Emp where Eid = 2));
                          UPDATE emp
                          SET salary = new_salary
                          WHERE Eid= empID;



                  END LOOP;
                  commit;
                CLOSE curEmpID;
        END$$
DELIMITER ;
call IncreaseSalary(80);
```