**Day 5**

**Problem Statement:**

Write a query to output the names of those students whose friends got offered a higher salary than them.

Names must be ordered by the salary amount offered to the friends.

It is guaranteed that no two students got same salary offer.

**Solution Code:**

/\*Creating a student\_Salary database and using the database\*/

create database student\_Salary;

use student\_Salary;

/\*Creating tables Students, Packages, Friends and inseting values in them\*/

create table Students(ID int, Name varchar(100));

insert into Students(ID,Name)values(1,"Ashley"),(2,"Samantha"),(3,"Julia"),(4,"Scarlet");

create table Packages(ID int, Salary float);

insert into Packages(ID,Salary) values(1,15.20),(2,10.06),(3,11.55),(4,12.12);

Create table Friends(ID int, Friends\_ID int);

insert into Friends(ID,Friends\_ID) values(1,2),(2,3),(3,4),(4,1);

/\*Checking the tables\*/

select \* from students;

select \* from Packages;

select \* from Friends;

/\*Write a query to output the names of those students whose friends got offered a higher salary than them.

Names must be ordered by the salary amount offered to the friends.

It is guaranteed that no two students got same salary offer.\*/

select Name from

(select S1.ID, S1.Name as Name, P1.Salary, F1.Friends\_ID,S2.Name as Friends\_name, P2.Salary as Friends\_Salary

from Students as S1

inner join Packages as P1 on S1.ID = P1.ID

inner join Friends F1 on F1.ID = S1.ID

inner join Students as S2 on F1.Friends\_ID = S2.ID

inner join Packages as P2 on F1.Friends\_ID = P2.ID

) AS T1

where T1.Friends\_Salary > T1.Salary

order by T1.Friends\_Salary

**Screenshot of the Code:**

