**EXPERIMENT 5-6 Date: 17th February 2023**

**TITLE:** **Use of Inbuilt functions and relational algebra operation**

**OBJECTIVE:** To understand the use of inbuilt function and relational algebra with SQL query.

1. **Consider the given Table Structures and**
2. **Create Table**

**SUPPLIER - (SCODE, SNAME, SCITY, TURNOVER)**

CREATE TABLE SUPPLIER (

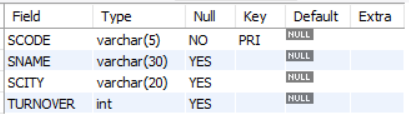
SCODE VARCHAR(5) PRIMARY KEY,

SNAME VARCHAR(30),

SCITY VARCHAR(20),

TURNOVER INTEGER

);



**PART - (PCODE, WEIGH, COLOR, COST, SELLINGPRICE)**

CREATE TABLE PART (

PCODE VARCHAR(5) PRIMARY KEY,

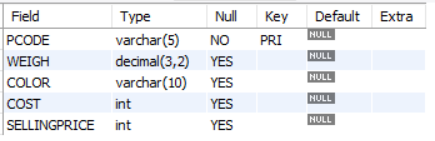
WEIGH DECIMAL(3,2),

COLOR VARCHAR(10),

COST INTEGER,

SELLINGPRICE INTEGER

);



**SUPPLIER\_PART - (SCODE, PCODE, QTY)**

CREATE TABLE SUPPLIER\_PART(

SCODE VARCHAR(5),

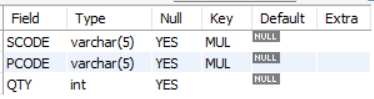
PCODE VARCHAR(5),

QTY INTEGER,

FOREIGN KEY (SCODE) REFERENCES SUPPLIER(SCODE),

FOREIGN KEY (PCODE) REFERENCES PART(PCODE)

);



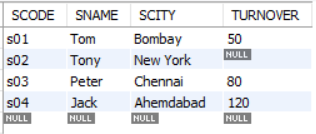
1. **Populate the tables.**

INSERT INTO SUPPLIER VALUES('S01','TOM','BOMBAY',50);

INSERT INTO SUPPLIER VALUES('S02','TONY','NEW YORK',NULL);

INSERT INTO SUPPLIER VALUES('S03','PETER','CHENNAI',80);

INSERT INTO SUPPLIER VALUES('S04','JACK','AHEMDABAD',120);

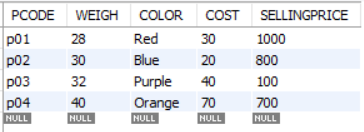
****

INSERT INTO PART VALUES("P01", 28, "RED", 30, 1000);

INSERT INTO PART VALUES("P02", 30, "BLUE",20, 800);

INSERT INTO PART VALUES("P03", 32, "PURPLE", 40, 100);

INSERT INTO PART VALUES("P04", 40, "ORANGE", 70, 700);

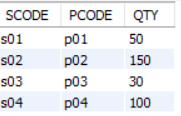
****

INSERT INTO SUPPLIER\_PART VALUES('S01',"P01",50);

INSERT INTO SUPPLIER\_PART VALUES('S02','P02',150);

INSERT INTO SUPPLIER\_PART VALUES('S03','P03',30);

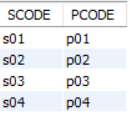
INSERT INTO SUPPLIER\_PART VALUES('S04','P04',100);

****

1. **Write appropriate SQL Statement for the following:**
2. **Get the supplier number and part number in ascending order of supplier number.**

∏SCODE,PCODE(σSUPPLIER.SCODE=PART.PCODE((SUPPLIER) (PART)))

SELECT SCODE, PCODE FROM SUPPLIER, PART ORDER BY SUPPLIER.SCODE;



1. **Get the details of supplier who operate from Bombay with turnover 50.**

∏SNAME (σ SCITY = "BOMBAY" *Λ* TURNOVER = 50 (SUPPLIER))

SELECT SNAME FROM SUPPLIER WHERE (SCITY = "BOMBAY" AND TURNOVER = 50);



1. **Get the total number of suppliers.**

∏COUNT(SCODE) (σ (SUPPLIER))

SELECT COUNT(SCODE) AS TOTAL\_NO\_OF\_SUPPLIER FROM SUPPLIER;



1. **Get the part number weighing between 25 and 35.**

∏PCODE (σ WEIGH>25 *Λ WEIGH<35*(PART))

SELECT PCODE AS PART FROM PART WHERE (WEIGH BETWEEN 25 AND 35);



1. **Get the supplier number whose turnover is null.**

∏SCODE (σ TURNOVER IS NULL (SUPPLIER))

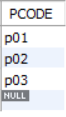
SELECT SCODE AS SUPPLIER\_NUMBER FROM SUPPLIER WHERE TURNOVER IS NULL;



1. **Get the part number that cost 20, 30 or 40 rupees.**

∏PCODE (σ COST IN (20, 30, 40) (PART))

SELECT PCODE FROM PART WHERE COST IN (20, 30, 40);



1. **Get the total quantity of part 2 that is supplied.**

∏SUM(QTY) (σ PCODE=’2’ (SUPPLIER\_PART))

SELECT SUM(QTY) AS TOTAL\_QUANTITY FROM SUPPLIER\_PART WHERE PCODE = "2";



1. **Get the name of supplier who supply part 2.**

∏SNAME (σ PCODE=’2’ (SUPPLIER SUPPLIER\_PART))

SELECT SNAME FROM SUPPLIER WHERE SCODE IN (SELECT SCODE FROM SUPPLIER\_PART WHERE PCODE = '2');



1. **Get the part number whose cost is greater than the average cost.**

∏PCODE (σ COST > (∏ AVG(COST) (PART)))

SELECT PCODE FROM PART WHERE COST > (SELECT AVG(COST) FROM PART);



1. **Get the supplier number and turnover in descending order of turnover.**

∏SNAME, TURNOVER (SUPPLIER)

SELECT SNAME, TURNOVER FROM SUPPLIER ORDER BY TURNOVER DESC;

