CREATE TABLE Galleries(  
    ID INT NOT NULL PRIMARY KEY,  
      CITY VARCHAR(30) NOT NULL  
);  
CREATE TABLE Paintings(  
  ID INT not null primary key,  
  name VARCHAR(30) NOT NULL,  
  gallery\_id INT NOT NULL,  
  price INT NOT NULL,  
  FOREIGN KEY(gallery\_id) REFERENCES Galleries(ID)  
);  
insert into Galleries values (1,'Jaipur'),(2,'Kolkata'),(3,'Madhubani');

insert into Paintings values (1,'Patterns',3,5000),(2,'Ringer',1,4500),(3,'Gift',1,3200),(4,'Violin Lessons',2,6700),(5,'Curiosity',2,9800);

Create Table Sales\_agents(

id INT NOT NULL PRIMARY KEY,

last\_name VARCHAR(30) NOT NULL,

first\_name VARCHAR(30) NOT NULL,

gallery\_id INT,

agency\_fee INT,

FOREIGN KEY (gallery\_id) REFERENCES Galleries(ID)

);

CREATE TABLE Managers(

id INT NOT NULL PRIMARY KEY,

gallery\_id INT NOT NULL,

FOREIGN KEY (gallery\_id) REFERENCES Galleries(ID)

);

insert into Sales\_agents values (1,'Brown','Denis',2,2250),(2,'White','Kate',3,3120),(3,'Black','Sarah',2,1640),(4,'Smith','Helen',1,4500),(5,'Stewart','Tom',3,2130);

insert into Managers values (1,2),(2,3),(4,1);

**SINGLE ROW SUBQUERY**

select name,price,(select avg(price) from Paintings) AvgPrice from Paintings;

**MULTIPLE ROW SUBQUERY**

Subquery that returns multiple rows as an output to their parent query are called multiple row subquery.

select id,name,price from Paintings

where (name,price) IN (select name,min(price) from Paintings);

**CORRELATED SUBQUERY**

Subquery that returns multiple columns as output depending on the information obtained form parent query.

select city,(select count(\*) from Paintings p where g.id=p.gallery\_id) count\_painting from galleries g;

Question: Find the information of those sales agents whose agency fee is greater than or equal to the agency fee of their gallery.

select last\_name,first\_name,agency\_fee from sales\_agents sa1 where sa1.agency\_fee >= (select avg(agency\_fee) from sales\_agents sa2 where sa1.gallery\_id=sa2.gallery\_id);

**VIEWS**

create view vw\_PaintingPrice as

select name as painting,price,

(select avg(price) from paintings where price in (select price from paintings where price>=5000)) as avg\_price

from paintings;

select \* from vw\_PaintingPrice;

**INDEXES**

1. Clustered Index: create clustered index idx\_salesorderlinekey on sales\_order\_data (salesorderlinekey);
2. Non-Clustered Index: create nonclustered index idx\_