...../...../.....

Mehmet Alperen Sahin 110 510 242

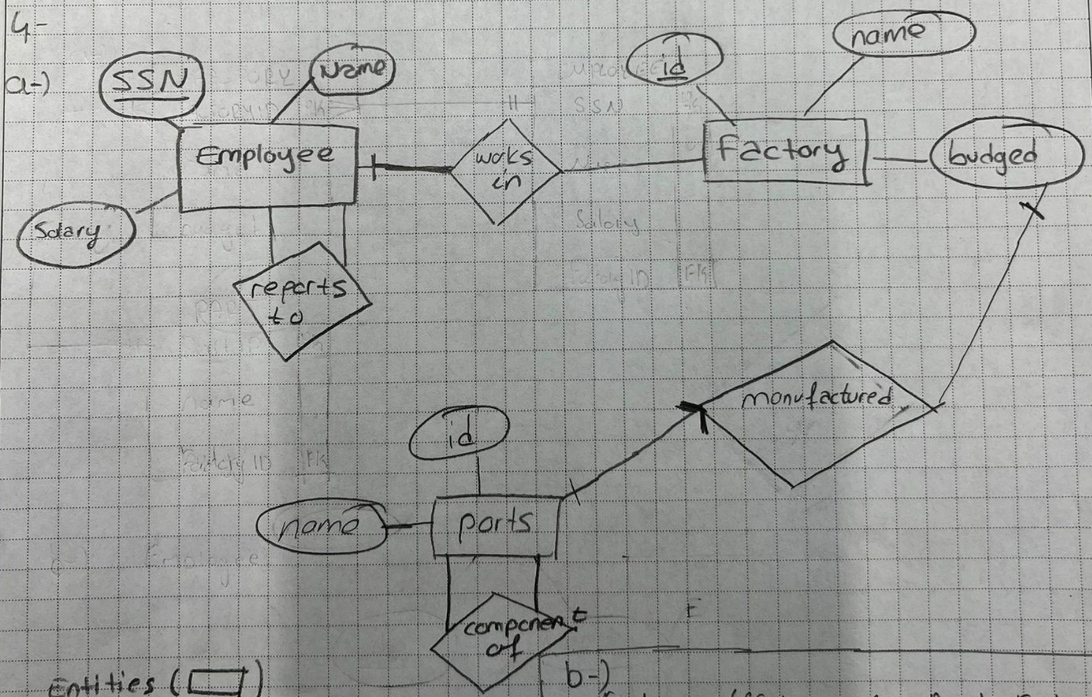
1- False

2- D

3-
```
SELECT User.name , OccupationName , City.cityName
FROM User
JOIN Occupation ON user.Occupationid = Occupation.Occupationid
JOIN City ON user.cityId = city.cityId
where User.age >= 18
```

4-

a-)



Entities (▭)

- Employee → SSN (primary key)
- factory – id (primary key)
- Port – id (primary key)

b-)

Employees (SSN, name, Salary, Reportsto, FactoryID)

→ FK referencing SSN in employee table

Factory (ID , Name, budget)
PK

Parts (ID, Nome, FactoryID)

...../...../.....

Mehmet Alperen Sahin 110510242

5-)

① CustomerID → City, PlateNo   productID → city

② Customer table

Customer ID	city	Palet No
u145	Denizli	20
u151	Tokat	60
u149	Samsun	55
u148	Tokat	60

ProductID

Product ID	City
U1	100
U2	150
U3	200

③ Sales

Customer ID	ProductID
u145	U1
u145	U2
u151	U2
u149	U3
u148	U1

in the normalized form we hahe three table;

1-) Customer table with customerID as the primary key

2-) Product Table with productID as primary key

3-) Sales table with customerID and productID as foreign keys the Customers and Products tables.

6-)

a-)
```
SELECT InvNo, city FROM INVOICE WHERE InvNo IN(
SELECT InvNo FROM INVITEM where itemNo = (
SELECT itemNo FROM ITEM where itemName = 'Washer') );
```

b-)
```
SELECT i.InvNo, ii.city FROM INVOICE i
JOIN INVITEM ii ON i.InvNO = ii.InvNo
JOIN ITEM it ON ii.itemNo = it.itemNo
where it.itemName = 'washer'
```

c-)
```
SELECT InvNo, COUNT(itemNo) as NumberOf Item
FROM INVITEM   GROUP BY InvNo
HAVING  COUNT(itemNo) >= 2
```

...../...../.....

Mehmet Alperen Şahin 110 510 242

7-)

```
  DELIMITER //
CREATE TRIGGER update_totalSales
AFTER INSERT ON Order_items
FOR EACH ROW
BEGIN
   UPDATE orders
   SET totalSales = totalSales + NEW.itemPrice
   WHERE orderId = NEW.orderId;
 END //
DELIMITER ;
```