Name: _____ USC ID: _____

Quiz 8: SQL, Constraints, Views (10 points. 15 minutes)

Consider again the following tables that you have seen in the last quiz. Note that attributes in the primary key of each table are underlined.

Laptop(SerialNo, OperatingSystem, HardDrive)
Product(SerialNo, Brand, Model, Price)
Store(ID, Name, City)
Purchase(BuyerName, SellerName, StoreID, SerialNo, Date)
Person(Name, PhoneNumber, City)

1. [2 points] Write a SQL query to "find brands such that at least five laptops for each brand have been sold." 没必要join仨表啊

```
SELECT Brand FROM Product P, Laptop L, Purchase R
WHERE P.SerialNo = L.SerialNo AND P.SerialNo = R.SerialNo
GROUP BY Brand HAVING COUNT(*) >= 5;
```

2. [2 points] List all foreign keys in the above tables. For each foreign key, indicates which attribute of which table it refers to.

Foreign Keys	References
Purchase(BuyerName, SellerName)	Person(Name)
Purchase(StoreID)	Store(ID)
Purchase(SerialNo)	Product(SerialNo)
Laptop(SerialNo)	Product(SerialNo)

3. [2 points] For which type(s) (insert, delete, and update) of modifications to the **Store** table, should database check possible violation of foreign key constraints? If yes, on which table? Is "set null" a good choice for possible action on the affected table when the violation occurs? Explain your answer.

No need to check when inserting while database should check possible violation of foreign key constraints on ***Purchase*** when deleting and updating ***Store***.
Set NULL is not a good idea because ***StoreID*** is part of the primary key in ***Purchase***.

4. [2 points] Write a view BrandStoreView to compute the total number of purchases per brand and store city.

```
CREATE VIEW BrandStoreView AS
      SELECT Brand, City, COUNT(*) NOP
      FROM Purchase P JOIN Store S ON P.StoreID = S.ID
           NATURAL JOIN Product
      GROUP BY Brand, City;
```

5. [2 points] Use the view obtained in the above question to "find the total number of purchased per brand".

```
SELECT Brand, SUM(NOP) FROM BrandStoreView GROUP BY Brand;
```