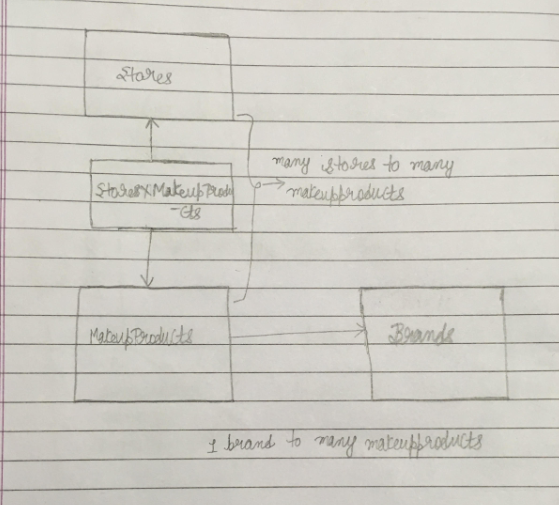
**ERD**

This system is about the makeup products.In this system, we have makeup-products, stores and the brands**.**The facts about this system are written below.

****

**fig.(a)-ERD diagram**

**1.One brand to many makeupproducts:-** This is because one brand can have many makeupproducts associated with it.Like ‘Loreal’ one brand can have many makeup products like foundations, hair-colours and moisturizing creams.So, one brand can have many makeup-products associated with it.But, on the other hand one product can’t be of many brands like one eyeliner can’t be of combined two or more brands.It could only belong to one brand.So, reverse does not exist.That is one product can have one brand associated with it.That is why this is one brand to many makeup-products relationship.So, the makeupproducts table will have a foreign key brandid that will point to the primary key brandid of the brands table .

**2.Many stores to many makeupproducts:-** This is because one store can have many makeupproducts.Like one store can have eyeliners, facecreams and other products.Similarly, one product can be present on many stores like one same brand loreal foundation can be found on different stores.So, this relationship is true from both sides.In other words, we have many stores to many makeupproducts relationship.

Since, it is a many to many relationship.That is why we also have bridging table that will connect this two tables let us say “StoresxMakeupProducts” as shown in the fig.(a).This table will have foreign keys storeid and makeproductid that will point to the primary keys storeid and the makeupproductid of both of the tables stores and makeupproducts.

**The data that each table will contain is represented through following fig.(b)Where primary keys and foreign keys are shown.But, the other data is just written as dot as shown below.**

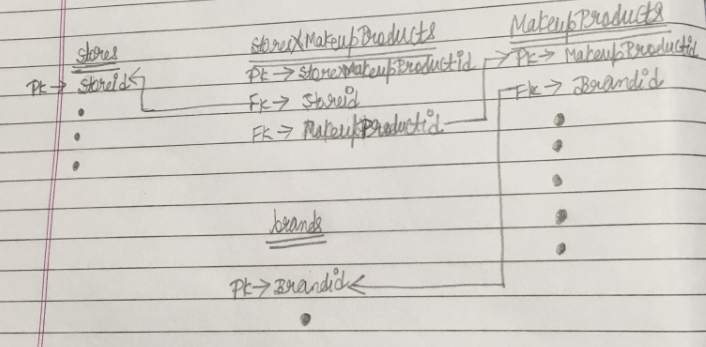


fig.(b)