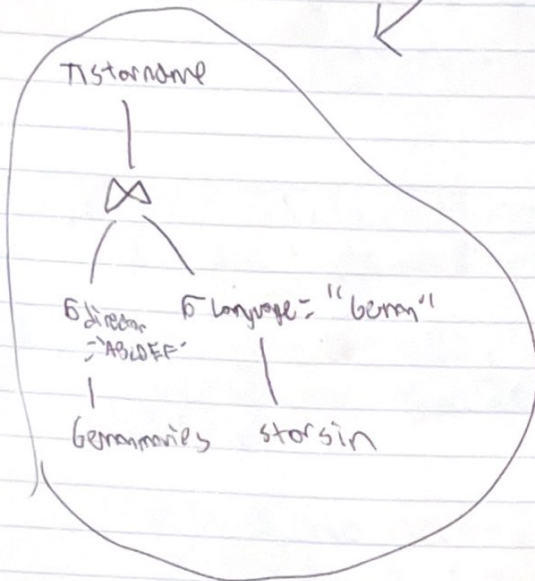
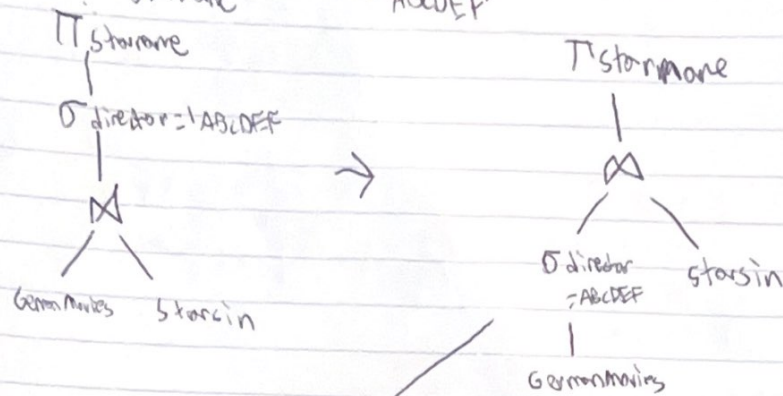


6.1 select R1.sid, R2.sid
FROM CATALOG R1, CATALOG R2
where R1.PID = R2.PID AND R1.SID <> R2.SID AND
R1.price > R2.price;

6.2 SELECT ~~SID~~ sid
FROM Supplier S
where sid IN (SELECT Catalog.sid FROM Parts, Catalog
WHERE S.sid = Catalog.sid AND Parts.pid =
Catalog.pid AND Parts.color = "red")
OR S.address = "Ruston City";

9/10

10.1 $\Pi_{\text{starmore}} (\sigma_{\text{director} = \text{'ABCDE'}} (\text{GermanMovies} \bowtie \text{starsin}))$



10.2



$A \rightarrow B$ p_d
 $A \rightarrow C$ p_d
 $AE \rightarrow D$ p_d
 $AEF \rightarrow G$

$AEF^+ : ABCDEFG$

$R(ABCEFG)$
 $\hookrightarrow R_1(AB)$
 $R_2(ACDEFG)$
 $\hookrightarrow R_{21}(AC)$
 $R_{22}(ADEFG)$
 $\hookrightarrow R_{221}(AED)$
 $(R_{222}(AEFG))$

$R_1(AB)$
 $R_2(AC)$
 $R_3(AED)$
 $R_4(AEFG)$

10.3. ~~Read S into Memory.~~
~~for each tuple in R check if it exists in S.~~

~~Read R into Memory~~
~~load a tuple at a time of S into memory.~~
~~check~~

Read S into memory, ~~check~~ check each tuple of R
 if it is in S, if it is, denoting, else output tuple.

8

2.1

Student (SSN, SName, SAddress, HSCode, HSName, HSCity, GPA, Priority)

HSCode \rightarrow HSName, HSCity

GPA \rightarrow Priority

SSN \rightarrow SName, SAddress, GPA

1NF ✓
2NF ✓

Student (SSN, SName, SAddress, GPA, Priority)

Attend (HSCode, SSN)

HS (HSCode, HSName, HSCity)

Student (SSN, SName, SAddress)

Glasses (SSN, GPA)

Grade (GPA, Priority)

Attend (HSCode, SSN)

HS (HSCode, HSName, HSCity)

3NF ✓

BCNF ✓

2.2

1. push selects as far down as possible to reduce size of the relations.
2. Cascade selects to reduce relation size faster
3. Insert projects near leaves to reduce redundant info to make tuples smaller.