**Question 1:**

**Keys:**

Banks: Primary - (BankName, City)

As the project states the banks are specified by the name of the bank and the city where it is located so together, they form the composite primary key. Added two check constraints to make sure the number of accounts was more than zero because having no accounts or negative accounts makes no sense. I also added a constraint to the Security making sure it was only values the database used. e.g "excellent","very good" etc.

Robberies: Primary - (BankName, City ,Date)

I have chosen BankName, City and Date as the composite primary key because while I maybe could have done BankName and Date alone, there is a possibility that a bank with the same name in a different city got robbed on the same day so its best to add city as well.

Plans: Primary - (BankName, City, PlannedDate)

The primary should be the composite BankName, City and PlannedDate because using only BankName and PlannedDate won't work since some banks have the same name but are in different cities so 2 banks with the same name could be hit on the same day so I need city in there as well.

Robbers: Primary - RobberId

An ID should be completely unique to each robber making it a perfect primary key. I will also need to check that the number of years in prison does not exceed the age.

Skills: Primary - SkillId

Just as before an ID should uniquely identify each skill, the perfect primary key. The Descirption has a unique constraint so no duplicate skills get inserted.

HasSkills: Primary and Foreign - (RobberId, SkillId)

This composite primary key was chosen because a robber could have multiple skills, or a skill could apply to multiple robbers but together they should be a unique identifier. Having the Skills table with the foreign key constraint for SkillId in this table means that if a SkillID is in this table it also must be in the Skills table. The two foreign keys have on update CASCADE and on delete RESTRICT constraints because both can be updated safely but deleting would cause problems. Preference is always greater than zero because you cannot have a preference at zero.

HasAccount: Primary - (RobberId, BankName, City)

All the attributes need to be part of the primary key because every other combination will almost definitely have duplicates. Each on their own will duplicate. RobberId and BankName come close to being good but some banks have the same name in different locations which would result in a duplicate so that will not work.

Accomplices: Primary - (RobberId, BankName, City ,Date)

I have decided that these attributes need to be part of the primary key because it is the only way to uniquely identify everything. From every combination I tried there was a possibility of something being repeated, while doing it this way, the only way it would be repeated is if a robber would go to the same bank on the same day and rob it again which is quite unlikely.

CREATE TABLE "Banks"

(

"BankName" char(50),

"City" char(50),

"NoAccounts" integer CHECK ("NoAccounts" > 0),

"Security" char(50) CHECK ("Security" IN ('excellent','very good','good','weak')),

PRIMARY KEY ("BankName", "City")

);

CREATE TABLE "Robberies"

(

"BankName" char(50),

"City" char(50),

"Date" date,

"Amount" decimal,

PRIMARY KEY ("BankName","City","Date")

);

CREATE TABLE "Plans"

(

"BankName" char(50),

"City" char(50),

"PlannedDate" date,

"NoRobbers" integer,

PRIMARY KEY ("BankName","City", "PlannedDate")

);

CREATE TABLE "Robbers"

(

"RobberId" serial,

"Nickname" char(50),

"Age" integer,

"NoYears" integer,

PRIMARY KEY ("RobberId"),

CHECK ("Age" > "NoYears")

);

CREATE TABLE "Skills"

(

"SkillId" serial,

"Description" char(50) UNIQUE,

PRIMARY KEY ("SkillId")

);

CREATE TABLE "HasSkills"

(

"RobberId" integer,

"SkillId" integer,

"Preference" integer CHECK ("Preference" > 0),

"Grade" char(5),

PRIMARY KEY ("RobberId", "SkillId"),

CONSTRAINT "FK\_RobbSkill" FOREIGN KEY ("RobberId")

REFERENCES public."Robbers" ("RobberId")

ON UPDATE CASCADE

ON DELETE RESTRICT,

CONSTRAINT "FK\_Skilled" FOREIGN KEY ("SkillId")

REFERENCES "Skills" ("SkillId")

ON UPDATE CASCADE

ON DELETE RESTRICT

);

CREATE TABLE "HasAccounts"

(

"RobberId" integer,

"BankName" char(50),

"City" char(50),

PRIMARY KEY ("RobberId","BankName","City")

);

CREATE TABLE "Accomplices"

(

"RobberId" integer,

"BankName" char(50),

"City" char(50),

"Date" date,

"Share" decimal,

PRIMARY KEY ("RobberId", "BankName", "City", "Date")

);

**Question 2:**

**Steps for filling database with data:**

These 3 tables can be filled first because it is quite simple to fill them with data being it only takes one line to achieve.

1. \copy "Banks" FROM ~/Pro1/banks\_20.data
2. \copy "Robberies" FROM ~/Pro1/robberies\_20.data
3. \copy "Plans" FROM ~/Pro1/plans\_20.data

The Robbers table should be done next because we will need the RobberId for other tables and it is not much more difficult than the last 3 tables.

1. \copy "Robbers"("Nickname","Age","NoYears") FROM ~/Pro1/robbers\_20.data

The next two tables HasAccounts and Accomplices will be done next as it the next easiest to achieve and requires the RobberId created in the 4th step.

5.

CREATE TABLE "AccTemp"

(

"NickName" char(50) NOT NULL,

"BankName" char(50),

"City" char(50)

);

\copy "AccTemp" FROM ~/Pro1/hasaccounts\_20.data

INSERT INTO "HasAccounts" ("RobberId", "BankName", "City")

SELECT "Robbers"."RobberId", "BankName", "City"

FROM "AccTemp"

INNER JOIN "Robbers" ON "Robbers"."Nickname" = "AccTemp"."NickName";

DROP TABLE "AccTemp";

6.

CREATE TABLE "AcoTemp"

(

"NickName" char(50) NOT NULL,

"BankName" char(50),

"City" char(50),

"Date" date NOT NULL,

"Share" decimal

);

\copy "AcoTemp" FROM ~/Pro1/accomplices\_20.data

INSERT INTO "Accomplices" ("RobberId", "BankName", "City","Date","Share")

SELECT "Robbers"."RobberId", "BankName", "City", "Date", "Share"

FROM "AcoTemp"

INNER JOIN "Robbers" ON "Robbers"."Nickname" = "AcoTemp"."NickName";

DROP TABLE "AcoTemp";

The last tables to be created are the two skill tables. The Skills table should be made before hasSkills because of the need to have a SkillId set up for the hasSkill table.

7.

CREATE TABLE "SkillTemp"

(

"NickName" char(50) NOT NULL,

"Describe" char(50),

"Prefer" integer,

"Grade" char(50)

);

\copy "SkillTemp" FROM ~/Pro1/hasskills\_20.data

INSERT INTO "Skills" ("Description")

SELECT DISTINCT "Describe"

FROM "SkillTemp";

8.

INSERT INTO "HasSkills" ("RobberId", "SkillId", "Preference", "Grade")

SELECT "Robbers"."RobberId", "Skills"."SkillId", "Prefer", "Grade"

FROM "SkillTemp"

INNER JOIN "Robbers" ON "Robbers"."Nickname" = "SkillTemp"."NickName"

INNER JOIN "Skills" ON "Skills"."Description" = "SkillTemp"."Describe";

DROP TABLE "SkillTemp";

**Question 3:**

**1. a.**

INSERT INTO "Banks" ("BankName","City","NoAccounts","Security")

VALUES ('Loanshark Bank', 'Evanston', 100, 'very good');

ERROR: duplicate key value violates unique constraint "Banks\_pkey"

DETAIL: Key ("BankName", "City")=(Loanshark Bank, Evanston) already exists.

**b.**

INSERT INTO "Banks" ("BankName","City","NoAccounts","Security")

VALUES ('EasyLoan Bank', 'Evanston', -5, 'excellent');

ERROR: new row for relation "Banks" violates check constraint "Banks\_NoAccounts\_check"

DETAIL: Failing row contains (EasyLoan Bank, Evanston, -5, excellent).

**c.**

INSERT INTO "Banks" ("BankName","City","NoAccounts","Security")

VALUES ('EasyLoan Bank', 'Evanston', 100, 'poor');

ERROR: new row for relation "Banks" violates check constraint "Banks\_Security\_check"

DETAIL: Failing row contains (EasyLoan Bank, Evanston, 100, poor).

**2. a.**

INSERT INTO "Skills" ("SkillId","Description")

VALUES (20, 'Guarding');

ERROR: duplicate key value violates unique constraint "Skills\_Description\_key"

DETAIL: Key ("Description") = (Guarding) already exists.

**3. a.**

INSERT INTO "Robberies" ("BankName", "City", "Date", "Amount")

VALUES ('NXP Bank', 'Chicago', '2019-01-08', 1000);

ERROR: duplicate key value violates unique constraint "Robberies\_pkey"

DETAIL: Key ("BankName", "City", "Date")=(NXP Bank, Chicago, 2019-01-08) already exists.

**4. a.**

DELETE FROM "Banks"

WHERE "BankName" = 'PickPocket Bank'

AND "City" = 'Evanston'

AND "NoAccounts" = 2000

AND "Security" = 'very good';

I am not sure why this cannot be deleted, maybe if a foreign key depended on it but I'm not sure how to achieve that in another table since the other tables require them to be apart of the primary key.

**5. a.**

DELETE FROM "Robberies"

WHERE "BankName" = 'Loanshark Bank'

AND "City" = 'Chicago'

AND "Date" = ''

AND "Amount" = '';

ERROR: invalid input syntax for type date: ""

LINE 4: AND "Date" = ''

**6. a.**

INSERT INTO "Robbers" ("RobberId", "Nickname", "Age", "NoYears")

VALUES (1, 'Shotgun', 70, 0);

ERROR: duplicate key value violates unique constraint "Robbers\_pkey"

DETAIL: Key ("RobberId")=(1) already exists.

**b.**

INSERT INTO "Robbers" ("RobberId", "Nickname", "Age", "NoYears")

VALUES (333, 'Jail Mouse', 25, 35);

ERROR: new row for relation "Robbers" violates check constraint "Robbers\_check"

DETAIL: Failing row contains (333, Jail Mouse, 25, 35).

**7. a.**

INSERT INTO "HasSkills" ("RobberId", "SkillId", "Preference", "Grade")

VALUES (333, 1, 1, 'B-');

ERROR: insert or update on table "HasSkills" violates foreign key constraint "FK\_RobbSkill"

DETAIL: Key (RobberId)=(333) is not present in table "Robbers".

**b.**

INSERT INTO "HasSkills" ("RobberId", "SkillId", "Preference", "Grade")

VALUES (3, 20, 3, 'B+');

ERROR: insert or update on table "HasSkills" violates foreign key constraint "FK\_Skilled"

DETAIL: Key (SkillId)=(20) is not present in table "Skills".

**c.**

INSERT INTO "HasSkills" ("RobberId", "SkillId", "Preference", "Grade")

VALUES (1, 7, 1, 'A+');

ERROR: duplicate key value violates unique constraint "HasSkills\_pkey"

DETAIL: Key ("RobberId", "SkillId")=(1, 7) already exists.

**d.**

INSERT INTO "HasSkills" ("RobberId", "SkillId", "Preference", "Grade")

VALUES (1, 2, 0, 'A');

ERROR: new row for relation "HasSkills" violates check constraint "HasSkills\_Preference\_check"

DETAIL: Failing row contains (1, 2, 0, A).

**8. a.**

DELETE FROM "Skills"

WHERE "SkillId" = 1

AND "Description" = 'Driving';

DELETE 0

**9. a.**

DELETE FROM "Robbers"

WHERE "RobberId" = 1

AND "Nickname" = 'Al Capone'

AND "Age" = 31

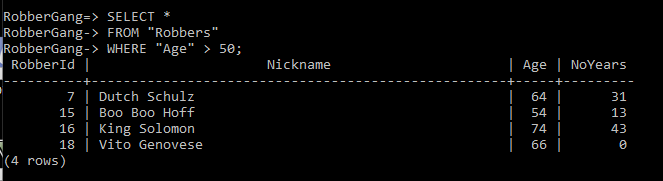
AND "NoYears" = 2;

ERROR: update or delete on table "Robbers" violates foreign key constraint "FK\_RobbSkill" on table "HasSkills"

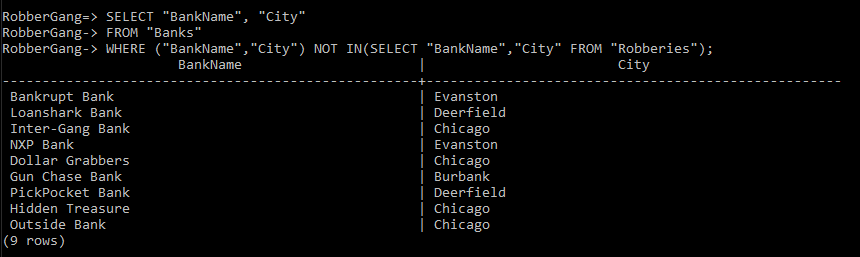
DETAIL: Key (RobberId)=(1) is still referenced from table "HasSkills".

**Question 4:**

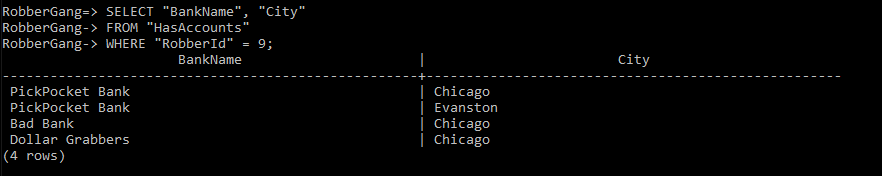
**1.**



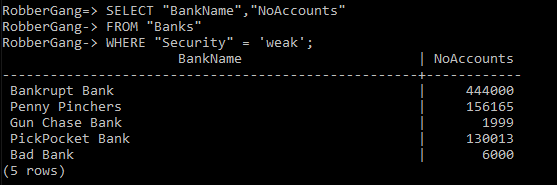
**2.**

****

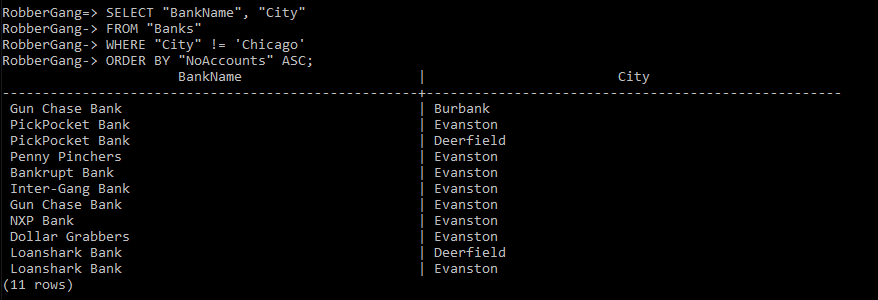
**3.**

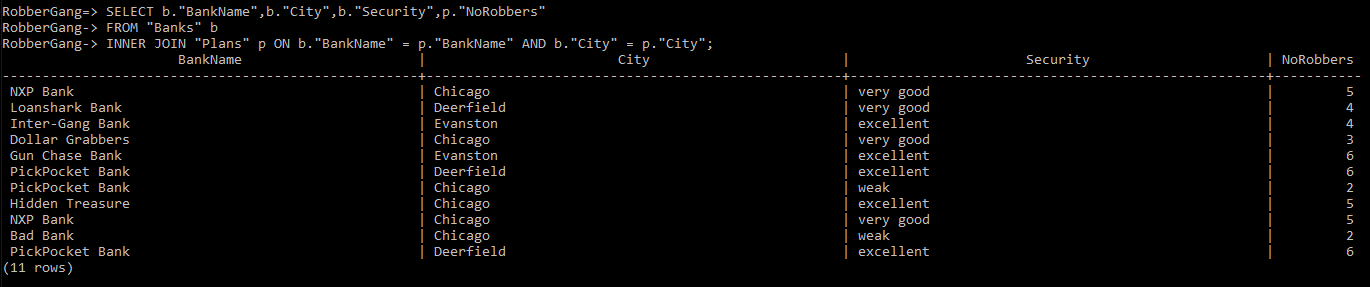
****

**4.**

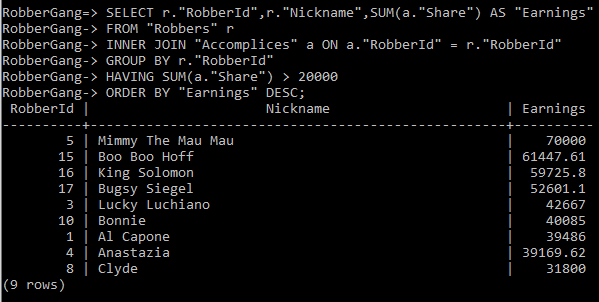
****

**5.**

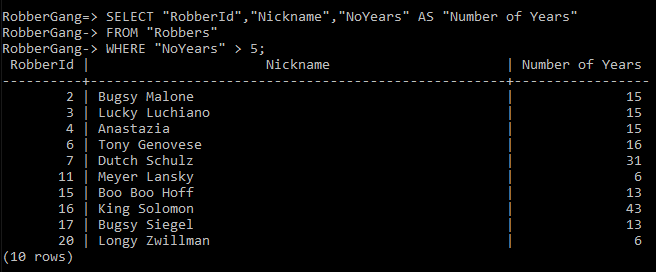
****

**6.**

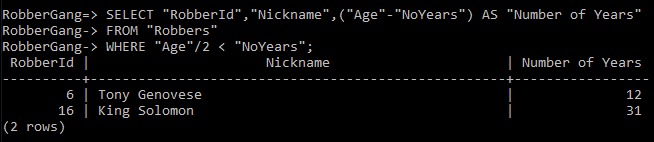
**7.**

****

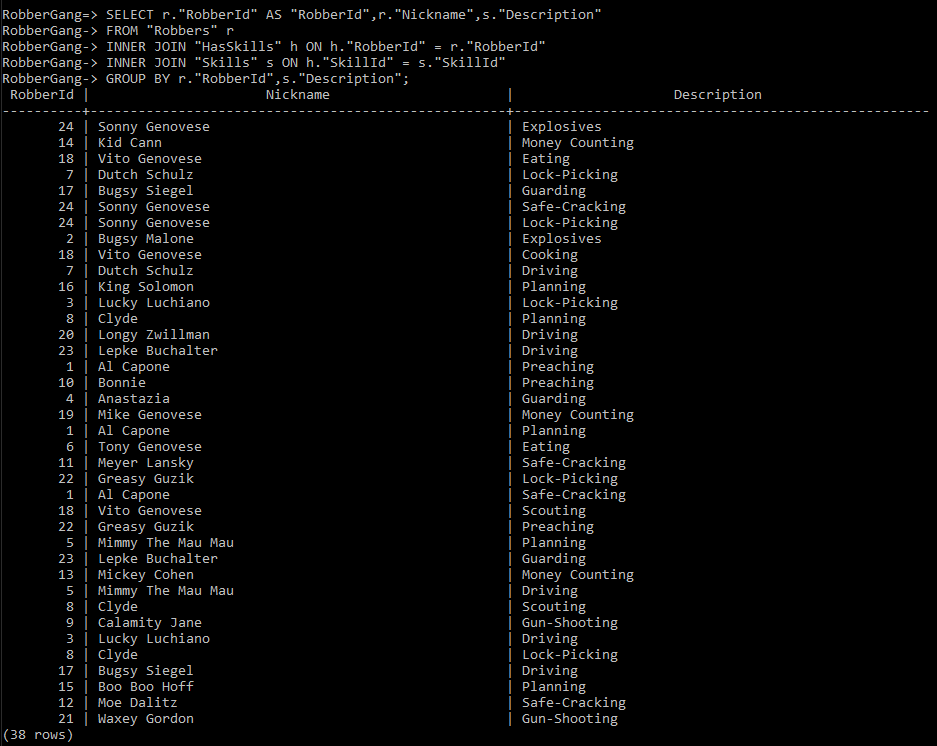
**8.**

****

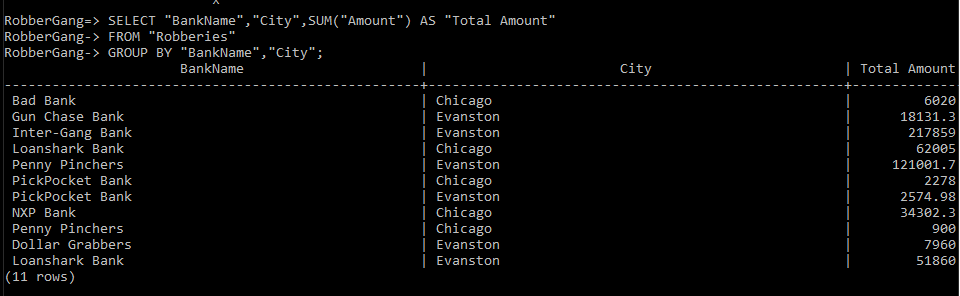
**9.**

****

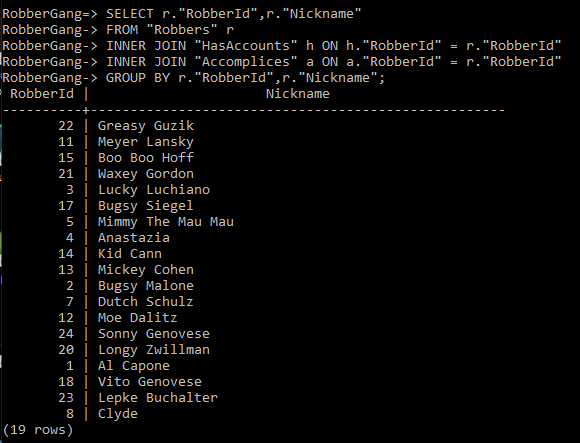
**10.**

****

**Question 5: 1.**

****

**2.**

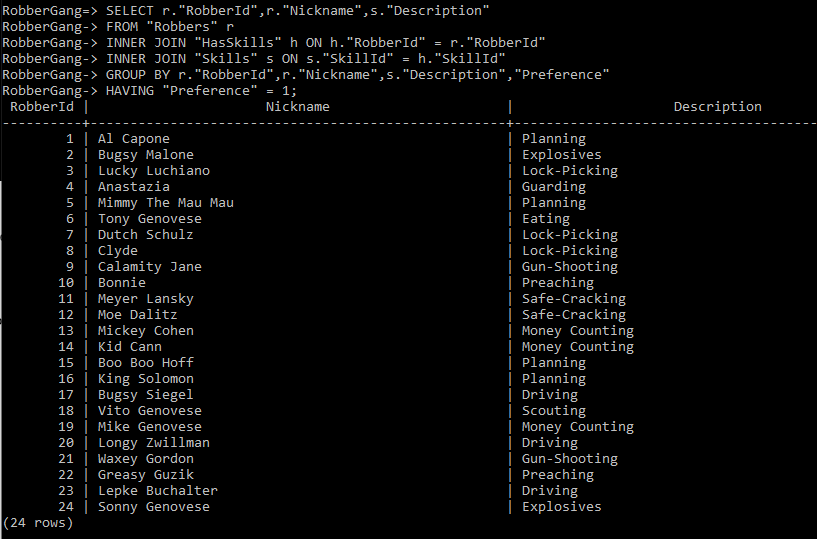
****

**3.**

I know that what I have is wrong, it shows the banks that were robbed in that same year there were plans to rob them.

**4.**

Tried to accomplish this one I only managed to get the robbers with there number one preference skill.



**5.**

I think I am supposed to do some sub querying here but I was not able to figure it out. The SQL file contains two queries that I tried to get it working. Most I managed was getting the two cities with the highest single share but without the right attributes.