SWEN 303

Project 1 300376862 Campbell Longmire

Question 1:

BANK TABLE:

Because of the information stated in the brief that all names of the banks along with their respective cities would be unique, and because other tables had the same information that then could be used as a foreign key, I decided to use this as the primary kay for this table and make it a NOT NULL column as it was needed for database structure.

```
CREATE TABLE Bank(BankName varchar(255) NOT NULL, City varchar(255)
NOT NULL, Account int, Security varchar(255), PRIMARY KEY(BankName,
City));
```

ROBBERIES TABLE:

Because I was planning on using the Bank name and City as the foreign key to the Bank table, as more importantly because it was not unique, I decided the best primary key to use would be a unique integer that would be added to each instance in this table. I then used the City and Name as the foriegn key. Because of this the three columns Bankname, City, and RobberiesID were created with NOT NULL constraints.

```
CREATE TABLE Robberies(RobberiesID int NOT NULL, BankName
varchar(255) NOT NULL, City varchar(255) NOT NULL, Date date, Amount
int, PRIMARY KEY(RobberiesID),FOREIGN KEY (BankName,City) REFERENCES
Bank(BankName,City));
```

ROBBERS TABLE:

Within the Robber table data there were no columns that could have been the primary key or no combination of columns that could have been the primary key. Becasue of this I created a new column for RobberID’s that would consist of unique integers that could identify each instance. Because of this this value was made to be not null. I would then also use this column to link this table to multiple other tables.

```
CREATE TABLE Robbers(RobberID int NOT NULL, Nickname varchar(100),
Age int, Noyear int, PRIMARY KEY(RobberID));
```

PLANS TABLE:
Again with this table and many others, there was no obvious choice for a primary key, So one needed to be added, in this case it was PlansID which meant that column needed to be not null. The foreign keys used were Bankname and city that referenced the Bank table which were also not null values.

```
CREATE TABLE Plans(PlansID int NOT NULL, BankName varchar(255) NOT
NULL, City varchar(255) NOT NULL, NoRobbers int, PlannedDate date,
PRIMARY KEY(PlansID), FOREIGN KEY (BankName,City) REFERENCES
Bank(BankName,City));
```

HASACCOUNT TABLE;

A primary key had to be created for this table which was accountID, along with this and the foreign key RobberID were the columns made to be not null, I also made the columns Bank Name and city not null just in case they would later be used for a foreign key

```
CREATE TABLE HasAccount(AccountID int NOT NULL, RobberID int NOT
NULL, BankName varchar(255) NOT NULL, City varchar(255) NOT NULL,
PRIMARY KEY(AccountID), FOREIGN KEY (RobberID) REFERENCES
Robbers(RobberID));
```

ACCOMPLICES TABLE:

A primary key had to be created for this table which was AccompID, along with this and the foreign key RobberID were the columns made to be not null.

```
CREATE TABLE Accomplices(AccompID int NOT NULL, RobberID int NOT
NULL, BankName varchar(255) NOT NULL, City varchar(255)
, RobberyDate date, Share Decimal(10,2),PRIMARY KEY(AccompID),FOREIGN
KEY (RobberID) REFERENCES Robbers(RobberID));
```

HASSKILLS TABLE:

A primary key had to be created for this table which was HasShillID, along with this and the foreign key RobberID and SkillsID were the columns made to be not null.

```
CREATE TABLE HasSkills(HasSkillID int NOT NULL, RobberID int NOT
NULL, SkillsID int NOT NULL, Preference int, Grade varchar(3),PRIMARY
KEY(HasSkillID),FOREIGN KEY (RobberID) REFERENCES Robbers(RobberID));
```

SKILLS TABLE:

Because there were only 2 columns, one being the created primary integer key and one being required defining information, therefore both were made to be not null.

```
CREATE TABLE Skills(SkillsID int NOT NULL, Description varchar(50)
NOT NULL,PRIMARY KEY(SkillsID));
```

ALTERATIONS:

Because I made the skills table after the hasSkills table, I was required to alter the hasskills table to contain the foriegn key SkillsID.

```
ALTER TABLE HasSkills
ADD FOREIGN KEY(SkillsID) REFERENCES Skills(SkillsID);
```

Question 2:

Data Conversion:

In order for all of my datafiles to be in the correct format for the tables I created I used Excel to add in primary key column and populate them with unique values and shift around columns so that it would fit the table structure. I also used excel to create a Skills table and change the values within HasSkills to match. From there I would save them as csv files and upload them to the database using the commands below.

Order:

I chose this order to add the tables so that there was violation of the foreign key. Meaning that the table with the primary key that would link to another tables forign key would be added first so avoid errors in my insertion.

1. Banks `\COPY bank FROM '/am/courtenay/home1/longmicamp/SWEN304/FinalData/banks_19.csv' WITH (FORMAT csv);`

2. Robberies `\COPY robberies FROM '/am/courtenay/home1/longmicamp/SWEN304/FinalData/robberies_19.csv' WITH (FORMAT csv);`

3. Plans `\COPY plans FROM '/am/courtenay/home1/longmicamp/SWEN304/FinalData/plans_19.csv' WITH (FORMAT csv);`

4. Robbers `\COPY robbers FROM '/am/courtenay/home1/longmicamp/SWEN304/FinalData/robbers_19.csv' WITH (FORMAT csv);`

5. Hasaccount `\COPY hasaccount FROM '/am/courtenay/home1/longmicamp/SWEN304/FinalData/hasaccounts.csv' WITH (FORMAT csv);`

6. Accomplices `\COPY accomplices FROM '/am/courtenay/home1/longmicamp/SWEN304/FinalData/accomplices.csv' WITH (FORMAT csv);`

7. Skills `\COPY skills FROM '/am/courtenay/home1/longmicamp/SWEN304/FinalData/Skills,csv.csv' WITH (FORMAT csv);`

8. HasSkills `\COPY hasskills FROM '/am/courtenay/home1/longmicamp/SWEN304/FinalData/hasskills_19.csv' WITH (FORMAT csv);`

Question 3:

INSERT INTO bank VALUES('LoansharkBank','Evanston','100','verygood');

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

This should not function as the number of accounts would not be a negative number. A way that this could have been fixed would have been to either create the Account column with an Unsigned contraited, or alter the table later on to add the constraint. this would would not allow negative numbers.

INSERT INTO bank VALUES('EasyLoan Bank','Evanston','100','poor');
ERROR: duplicate key value violates unique constraint "bank_pkey"
DETAIL: Key (bankname, city)=(EasyLoan Bank, Evanston) already exists.

INSERT INTO skills VALUES('20','Guarding');

There doesn't seem anything inherently wrong with allowing duplicate descriptions into this skills database, however it would also seem to be a case of bad table design, a fix for this issue would be to add a UNIQUE constraint to this column.

INSERT INTO robberies(bankname,city,date,amount) VALUES ('NXP Bank','Chicago','2019-01-08','1000');

CLdatabase=> create sequence robberiesid_seq;
CREATE SEQUENCE
CLdatabase=> alter table robberies alter robberiesid set default nextval('robberies id_seq');
ALTER TABLE
CLdatabase=> Select setval('robberiesid_seq',21);

These lines above were used to alter the tables that used primary keys that I had added in myself to ensure that they incremented every time a new instance was added so that the primary key remained unique.

```
CLdatabase=> DELETE FROM bank WHERE bankname = 'PickPocket Bank' AND  city
= 'Evanston';
ERROR:  update or delete on table "bank" violates foreign key constraint
"robberies_bankname_fkey" on table "robberies"
DETAIL:  Key (bankname, city)=(PickPocket Bank, Evanston) is still referenced from
table "robberies".


CLdatabase=> DELETE FROM bank WHERE bankname = 'Outside Bank' AND  city =
'Chicago';
DELETE 1


CLdatabase=> create sequence robberid_seq;
CREATE SEQUENCE
CLdatabase=> alter table robbers alter robberid set default nextval('robberid_seq');
ALTER TABLE
CLdatabase=> Select setval('robberid_seq',24);


CLdatabase=> INSERT INTO robbers VALUES('1','Shotgun','70','0');
ERROR:  duplicate key value violates unique constraint "robbers_pkey"
DETAIL:  Key (robberid)=(1) already exists.

CLdatabase=> INSERT INTO robbers VALUES('333','Jail Mouse','25','35');
INSERT 0 1

CLdatabase=> create sequence hasskillid_seq;
CREATE SEQUENCE
CLdatabase=> alter table hasskill alter hasskillid set default nextval('hasskillid_seq');
ERROR:  relation "hasskill" does not exist
CLdatabase=> alter table hasskills alter hasskillid set default nextval('hasskillid_seq');
ALTER TABLE
CLdatabase=> Select setval('hasskillid_seq',38);
```

```
CLdatabase=> INSERT INTO hasskills(robberid,skillsid,preference,grade) VALUES
('1','2','0','A');
INSERT 0 1
```

```
CLdatabase=> INSERT INTO hasskills(robberid,skillsid,preference,grade) VALUES
('333','1','1','B-');
INSERT 0 1
```

```
CLdatabase=> INSERT INTO hasskills(robberid,skillsid,preference,grade) VALUES
('3','20','3','B+');
INSERT 0 1

CLdatabase=> INSERT INTO hasskills(robberid,skillsid,preference,grade) VALUES
('1','7','1','A+');
INSERT 0 1
```

Constraints violated would be preferences being the same for different tasks of the same robber. I.e bother explosives and driving could be preference ‘1’ for a specific robber which doesn’t make sense. A way around that I could find would be to use `CREATE UNIQUE INDEX [UNQ_SampleTable_Code] ON dbo.[SampleTable]([Code])` on the column that is meant to be unique.

```
CLdatabase=> DELETE FROM skills WHERE skillsid = '1' AND  description = 'Driving';
DELETE 0
```

This doesn’t function as the skillsid = ‘1’ does not contain the description = ‘Driving’.

Question 4:

I was unable to dump sql files for parts 4 and 5 as I was at home in Auckland using Putty and was not able to get WinSCP to work to retrieve the dumped files. I hope this is okay and can submit the files when I return to Wellington on tuesday if that helps.

Task 1:

```
CLdatabase=> SELECT robbers.nickname,hasaccount.bankname FROM robbers
INNER JOIN hasaccount ON robbers.robberID=hasaccount.robberID WHERE
robbers.nickname='Calamity Jane';
   nickname    |    bankname
---------------+-----------------
 Calamity Jane | Dollar Grabbers
 Calamity Jane | Bad Bank
 Calamity Jane | PickPocket Bank
 Calamity Jane | PickPocket Bank
(4 rows)
```

Task 2:

```
CLdatabase=> SELECT bankname,security,city FROM bank WHERE account>9000
AND city = 'Chicago';
    bankname     | security  |  city
-----------------+-----------+---------
 NXP Bank        | very good | Chicago
 Loanshark Bank  | excellent | Chicago
 Inter-Gang Bank | excellent | Chicago
 Penny Pinchers  | weak      | Chicago
 Dollar Grabbers | very good | Chicago
 PickPocket Bank | weak      | Chicago
 Hidden Treasure | excellent | Chicago
```

Task 3:

```
CLdatabase=> SELECT bankname,city,account FROM bank WHERE city != 'Chicago'
ORDER BY account;
    bankname     |   city    | account
-----------------+-----------+---------
 EasyLoan Bank   | Evanston  |      -5
 LoansharkBank   | Evanston  |     100
 Gun Chase Bank  | Burbank   |    1999
 PickPocket Bank | Evanston  |    2000
 PickPocket Bank | Deerfield |    6565
 Penny Pinchers  | Evanston  |  130013
 Bankrupt Bank   | Evanston  |  444000
 Inter-Gang Bank | Evanston  |  555555
 Gun Chase Bank  | Evanston  |  656565
 NXP Bank        | Evanston  |  656565
 Dollar Grabbers | Evanston  |  909090
 Loanshark Bank  | Deerfield | 3456789
 Loanshark Bank  | Evanston  | 7654321
(13 rows)
```

Task 4:

```
CLdatabase=> SELECT bankname,MIN(date) AS mindate FROM robberies GROUP BY
bankname;
    bankname     |  mindate
-----------------+------------
 PickPocket Bank | 2015-09-21
 NXP Bank        | 2019-01-08
 Inter-Gang Bank | 2016-02-16
 Loanshark Bank  | 2016-04-20
 Gun Chase Bank  | 2016-04-30
 Penny Pinchers  | 2016-08-30
 Bad Bank        | 2017-02-02
 Dollar Grabbers | 2017-06-28
```

Task 6:

```
CLdatabase=> SELECT robberid,nickname,noyear FROM robbers WHERE noyear>3;
 robberid |    nickname    | noyear
----------+----------------+--------
        2 | Bugsy Malone   |     15
        3 | Lucky Luchiano |     15
        4 | Anastazia      |     15
        6 | Tony Genovese  |     16
        7 | Dutch Schulz   |     31
       11 | Meyer Lansky   |      6
```

```
      15 | Boo Boo Hoff   |     13
      16 | King Solomon   |     43
      17 | Bugsy Siegel   |     13
      20 | Longy Zwillman |      6
     333 | Jail Mouse     |     35
```

Task 7:

```
CLdatabase=> SELECT skills.Description, hasskills.robberid,robbers.nickname FROM
hasskills, skills, robbers WHERE Skills.skillsID=hasskills.skillsID AND
hasskills.robberid=robbers.robberid ORDER BY skills.description;
  description   | robberid |     nickname
----------------+----------+-------------------
 Cooking        |       18 | Vito Genovese
 Driving        |        7 | Dutch Schulz
 Driving        |       23 | Lepke Buchalter
 Driving        |        5 | Mimmy The Mau Mau
 Driving        |        3 | Lucky Luchiano
 Driving        |       17 | Bugsy Siegel
 Eating         |        6 | Tony Genovese
 Eating         |       18 | Vito Genovese
 Explosives     |       24 | Sonny Genovese
 Explosives     |       17 | Bugsy Siegel
 Guarding       |       17 | Bugsy Siegel
 Guarding       |        4 | Anastazia
 Guarding       |       23 | Lepke Buchalter
 Guarding       |        3 | Lucky Luchiano
 Gun-Shooting   |        9 | Calamity Jane
 Gun-Shooting   |       21 | Waxey Gordon
 Gun-Shooting   |        1 | Al Capone
 Lock-Picking   |       22 | Greasy Guzik
 Lock-Picking   |       24 | Sonny Genovese
 Lock-Picking   |        7 | Dutch Schulz
 Lock-Picking   |        8 | Clyde
 Lock-Picking   |        3 | Lucky Luchiano
 Money Counting |       14 | Kid Cann
 Money Counting |       13 | Mickey Cohen
 Money Counting |       19 | Mike Genovese
 Planning       |        8 | Clyde
 Planning       |       16 | King Solomon
 Planning       |        1 | Al Capone
 Planning       |       20 | Longy Zwillman
 Planning       |       15 | Boo Boo Hoff
 Planning       |        5 | Mimmy The Mau Mau
 Planning       |      333 | Jail Mouse
 Preaching      |        1 | Al Capone
 Preaching      |       22 | Greasy Guzik
 Preaching      |       10 | Bonnie
 Safe-Cracking  |       24 | Sonny Genovese
 Safe-Cracking  |        1 | Al Capone
 Safe-Cracking  |       11 | Meyer Lansky
 Safe-Cracking  |        1 | Al Capone
 Safe-Cracking  |       12 | Moe Dalitz
 Scouting       |        8 | Clyde
 Scouting       |       18 | Vito Genovese
```

Task 8:

```
CLdatabase=> SELECT RobberId, Nickname, (Age - NoYear) AS Yearsout FROM
ROBBERS WHERE NoYear > Age/2;
 robberid |    nickname    | yearsout
----------+---------------+----------
        6 | Tony Genovese |       12
       16 | King Solomon  |       31
      333 | Jail Mouse    |      -10
(3 rows)
```

Question 5

Task 1:

```
CLdatabase=> SELECT Security, AVG(Amount) AS AVAmount, COUNT(Security)
RobberiesNum
FROM(SELECT BankName, City, Amount, Security FROM Robberies NATURAL JOIN Bank)
AS Robszsecutiyamount
GROUP BY Security;

 security  |       avamount        | robberiesnum
-----------+-----------------------+--------------
 weak      | 2299.5000000000000000 |            4
 good      | 3980.0000000000000000 |            2
 very good | 9469.3200000000000000 |            4
 excellent |    39238.083333333333 |           12
```

Task 4:

```
CLdatabase=> SELECT DISTINCT Nickname, Security, Description FROM ACCOMPLICES
NATURAL JOIN robbers NATURAL JOIN bank NATURAL JOIN hasskills NATURAL JOIN skills
ORDER BY
 Security;
     nickname      | security  |  description
-------------------+-----------+----------------
 Al Capone         | excellent | Gun-Shooting
 Al Capone         | excellent | Planning
 Al Capone         | excellent | Preaching
 Al Capone         | excellent | Safe-Cracking
 Anastazia         | excellent | Guarding
 Bonnie            | excellent | Preaching
 Boo Boo Hoff      | excellent | Planning
 Bugsy Siegel      | excellent | Driving
 Bugsy Siegel      | excellent | Explosives
 Bugsy Siegel      | excellent | Guarding
 Clyde             | excellent | Lock-Picking
 Clyde             | excellent | Planning
 Clyde             | excellent | Scouting
 Dutch Schulz      | excellent | Driving
 Dutch Schulz      | excellent | Lock-Picking
 Greasy Guzik      | excellent | Lock-Picking
 Greasy Guzik      | excellent | Preaching
 King Solomon      | excellent | Planning
 Longy Zwillman    | excellent | Planning
```

```
Lucky Luchiano    | excellent | Driving
Lucky Luchiano    | excellent | Guarding
Lucky Luchiano    | excellent | Lock-Picking
Meyer Lansky      | excellent | Safe-Cracking
Mimmy The Mau Mau | excellent | Driving
Mimmy The Mau Mau | excellent | Planning
Sonny Genovese    | excellent | Explosives
Sonny Genovese    | excellent | Lock-Picking
Sonny Genovese    | excellent | Safe-Cracking
Waxey Gordon      | excellent | Gun-Shooting
Kid Cann          | good      | Money Counting
Mickey Cohen      | good      | Money Counting
Vito Genovese     | good      | Cooking
Vito Genovese     | good      | Eating
Vito Genovese     | good      | Scouting
Al Capone         | very good | Gun-Shooting
Al Capone         | very good | Planning
Al Capone         | very good | Preaching
Al Capone         | very good | Safe-Cracking
Anastazia         | very good | Guarding
King Solomon      | very good | Planning
Lepke Buchalter   | very good | Driving
Lepke Buchalter   | very good | Guarding
Longy Zwillman    | very good | Planning
Moe Dalitz        | very good | Safe-Cracking
Sonny Genovese    | very good | Explosives
Sonny Genovese    | very good | Lock-Picking
Sonny Genovese    | very good | Safe-Cracking
Al Capone         | weak      | Gun-Shooting
Al Capone         | weak      | Planning
Al Capone         | weak      | Preaching
Al Capone         | weak      | Safe-Cracking
Boo Boo Hoff      | weak      | Planning
Bugsy Siegel      | weak      | Driving
Bugsy Siegel      | weak      | Explosives
Bugsy Siegel      | weak      | Guarding
Clyde             | weak      | Lock-Picking
Clyde             | weak      | Planning
Clyde             | weak      | Scouting
Dutch Schulz      | weak      | Driving
Dutch Schulz      | weak      | Lock-Picking
Greasy Guzik      | weak      | Lock-Picking
Greasy Guzik      | weak      | Preaching
```

```
Lepke Buchalter   | weak      | Driving
Lepke Buchalter   | weak      | Guarding
Sonny Genovese    | weak      | Explosives
Sonny Genovese    | weak      | Lock-Picking
Sonny Genovese    | weak      | Safe-Cracking
Vito Genovese     | weak      | Cooking
Vito Genovese     | weak      | Eating
Vito Genovese     | weak      | Scouting
```