

Tenzin Tashi

Prof. Hesham A Auda

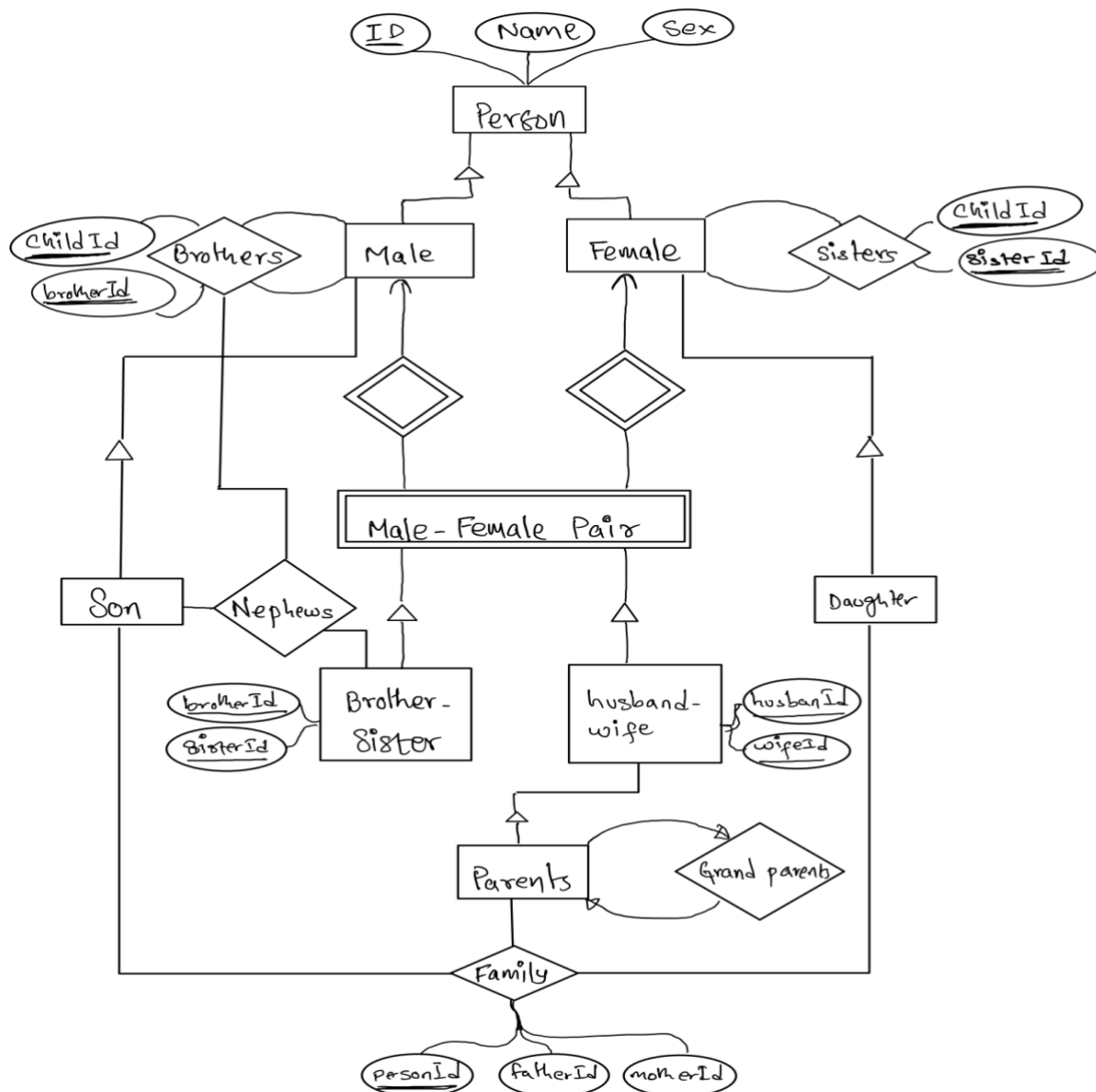
CSc - 33600 Introduction to Database Systems

March 7, 2021

Assignment 1

Family Relations – I

1. Provide the E-R diagram and corresponding relational database schema used for this assignment. Indicate for each relation: the key or keys, primary key, foreign keys, essential constraints, and any appropriate checks.



2. Provide appropriate data manipulation expressions to create the relations in a RDBS of your choice.

The data manipulation expressions are the statements that tell database how to store the data. The data manipulation expressions that I have used to create the relations in my DB are:

- INSERT INTO ... VALUES
- SELECT ... FROM ... WHERE

Person Table:

```
INSERT INTO Persons VALUES (1, 'Tenzin Tashi', 'M');
INSERT INTO Persons VALUES (2, 'Tsering Dolma', 'F');
INSERT INTO Persons VALUES (3, 'Jetsun Pema', 'F');
SELECT *
FROM PERSONS;
```

Family Table:

```
INSERT INTO Family VALUES (1, 11, 2);
INSERT INTO Family VALUES (3, 11, 2);
INSERT INTO Family VALUES (15, 11, 2);
INSERT INTO Family VALUES (4, 5, 6);
SELECT (SELECT Name FROM Persons WHERE Id = personId) AS 'Child',
       (SELECT Name FROM Persons WHERE Id = fatherId) AS 'Father',
       (SELECT Name FROM Persons WHERE Id = motherId) AS 'Mother'
FROM Family;
```

Spouses Table:

```
INSERT INTO Spouses VALUES (11, 2);
INSERT INTO Spouses VALUES (5, 6);
SELECT (SELECT Name FROM Persons WHERE Id = husbandId) AS 'Husband',
       (SELECT Name FROM Persons WHERE Id = wifeId) AS 'Wife'
FROM Spouses;
```

Brothers Table:

```
INSERT INTO Brothers
SELECT F1.personId AS 'childId', F2.personId AS 'brotherId'
FROM (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'M')) F1,
     (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'M')) F2
WHERE F1.fatherId = F2.fatherId AND F1.MotherId = F2.MotherId AND F1.personId <> F2.personId;

SELECT (SELECT Name FROM Persons WHERE Id = childId) AS 'Brother',
       (SELECT Name FROM Persons WHERE Id = brotherId) AS 'Brother'
FROM Brothers;
```

Sisters Table:

```
INSERT INTO Sisters
SELECT F1.personId AS 'childId', F2.personId AS 'sisterId'
FROM (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'F')) F1,
     (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'F')) F2
WHERE F1.fatherId = F2.fatherId AND F1.MotherId = F2.MotherId AND F1.personId <> F2.personId;

SELECT (SELECT Name FROM Persons WHERE Id = childId) AS 'Sister',
       (SELECT Name FROM Persons WHERE Id = sisterId) AS 'Sister'
FROM Sisters;
```

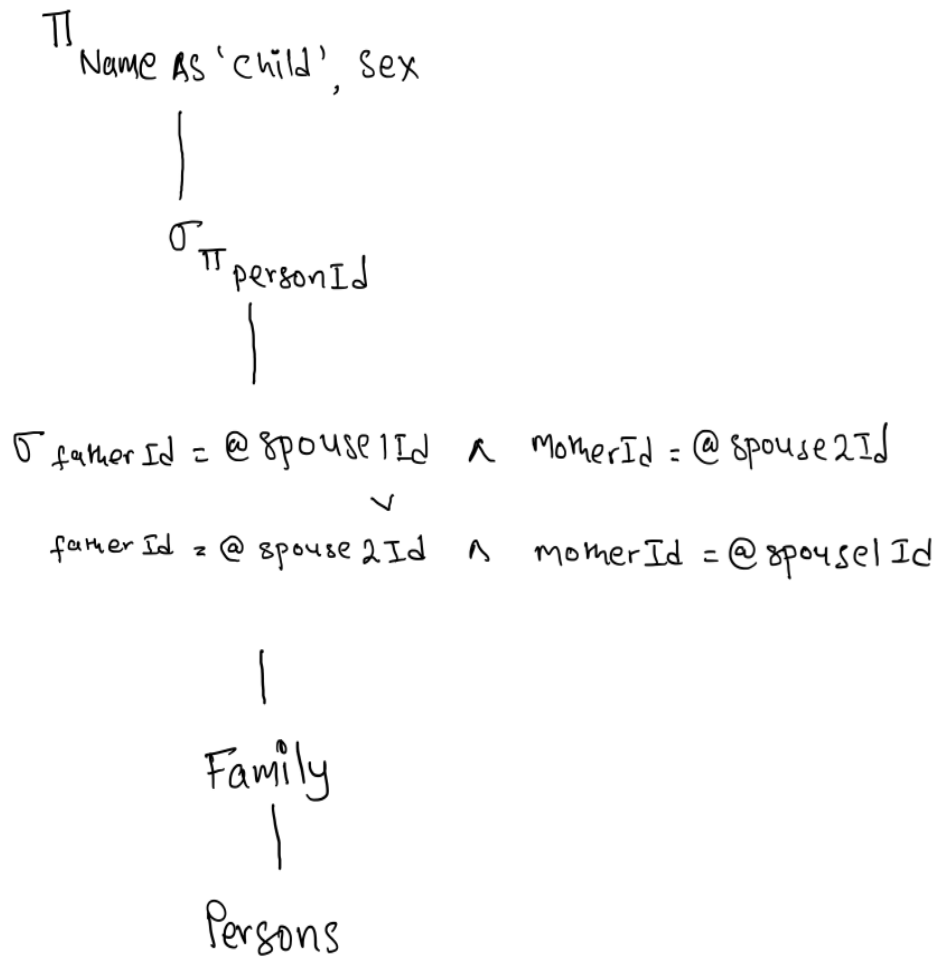
Brother-Sister Table:

```
INSERT INTO BrotherSisters
SELECT F1.personId AS 'brotherId', F2.personId AS 'sisterId'
FROM (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'M')) F1,
     (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'F')) F2
WHERE F1.fatherId = F2.fatherId AND F1.MotherId = F2.MotherId AND F1.personId <> F2.personId;

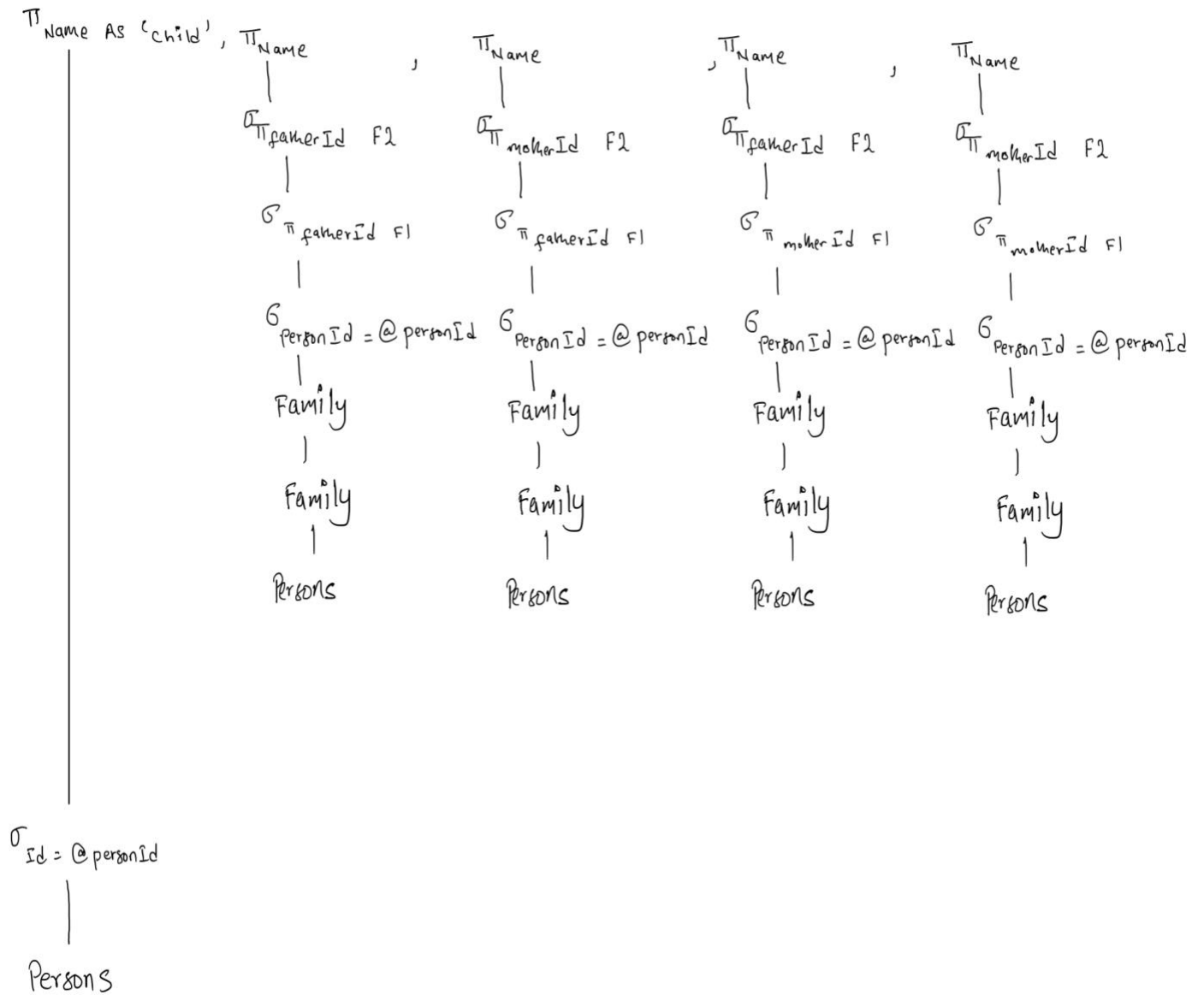
SELECT (SELECT Name FROM Persons WHERE Id = brotherId) AS 'Brother',
       (SELECT Name FROM Persons WHERE Id = sisterId) AS 'Sister'
FROM BrotherSisters;
```

3. Give appropriate relational algebra trees-expressions that return:

a. Children of a given couple.



b. Grandparents of a given person



c. Nephews -- sons of one's brother or sister -- of a given person.

$\pi_{\text{Name AS 'Nephews', Sex}}$
|
 $\sigma \pi_{\text{personId}}$
|
 $\sigma \pi_{\text{brotherId}}$
|
 $\sigma_{\text{childId} = @\text{personId} \wedge \text{sex} = 'M'}$
|
Brothers
|
Family
|
Persons

4. Provide appropriate SQL expressions for the queries in A3.

- Children of a given couple

```
/* Query: Children of a Couple */
DECLARE @spouse1Id INT = 11
DECLARE @spouse2Id INT = 2
SELECT Name AS 'Child', Sex
FROM Persons
WHERE Id IN (SELECT personId
             FROM Family
             WHERE fatherId = @spouse1Id AND motherId = @spouse2Id
             OR
             fatherId = @spouse2Id AND motherId = @spouse1Id);
```

- Grandparents of a given person

```
/* Query: Grandparents of a given person */
DECLARE @personId INT = 1
SELECT Name AS 'Child',
       (SELECT Name FROM Persons WHERE Id IN
        (SELECT fatherId FROM Family F2 WHERE F2.personId IN
         (SELECT fatherId FROM Family F1 WHERE F1.personId = @personId))) AS 'Paternal Grandfather',
       (SELECT Name FROM Persons WHERE Id IN
        (SELECT motherId FROM Family F2 WHERE F2.personId IN
         (SELECT fatherId FROM Family F1 WHERE F1.personId = @personId))) AS 'Paternal Grandmother',
       (SELECT Name FROM Persons WHERE Id IN
        (SELECT fatherId FROM Family F2 WHERE F2.personId IN
         (SELECT motherId FROM Family F1 WHERE F1.personId = @personId))) AS 'Maternal Grandfather',
       (SELECT Name FROM Persons WHERE Id IN
        (SELECT motherId FROM Family F2 WHERE F2.personId IN
         (SELECT motherId FROM Family F1 WHERE F1.personId = @personId))) AS 'Maternal Grandmother'
FROM Persons
WHERE Id = @personId;
```

- Nephews -- sons of one's brother or sister -- of a given person

```
/* Query: Nephews of a given person */
DECLARE @personId INT = 11
SELECT Name AS 'Nephew', Sex
FROM Persons
WHERE Id IN (SELECT personId
             FROM Family F1
             WHERE F1.fatherId IN (SELECT brotherId
                                   FROM Brothers F2
                                   WHERE F2.childId = @personId and Sex = 'M'));
```


B. Implement A above using your RDBS of choice. Test your implementation of the queries using an appropriate set of data. The data utilized must be representative and sufficient to demonstrate the validity of your queries.

CODE:

Link : <https://rextester.com/LVD59874>

```
1  --Sql Server 2014 Express Edition
2  --Batches are separated by 'go'
3
4  select @@version as 'sql server version'
5  Go
6
7  /* Function CheckPersonSex */
8  CREATE FUNCTION CheckPersonSex(@personId AS INT, @G as CHAR)
9      RETURNS BIT
10 AS
11 BEGIN
12     DECLARE @Flag BIT = 1
13     IF (@G NOT IN (SELECT Sex FROM Persons WHERE Id = @personId))
14         SET @Flag = 0
15     RETURN @Flag
16 END;
17 GO
18
19 /* Table Persons */
20 CREATE TABLE Persons(
21     Id INT PRIMARY KEY,
22     Name VARCHAR(64),
23     Sex CHAR CHECK(Sex = 'M' OR Sex = 'F'));
24
25 INSERT INTO Persons VALUES (1, 'Tenzin Tashi', 'M');
26 INSERT INTO Persons VALUES (2, 'Tsering Dolma', 'F');
27 INSERT INTO Persons VALUES (3, 'Jetsun Pema', 'F');
28 INSERT INTO Persons VALUES (4, 'Pasang Lama', 'M');
29 INSERT INTO Persons VALUES (5, 'James Lama', 'M');
30 INSERT INTO Persons VALUES (6, 'Jane Lama', 'F');
31 INSERT INTO Persons VALUES (7, 'Zoe Dunlap', 'F');
32 INSERT INTO Persons VALUES (8, 'Maximo Mingo', 'M');
33 INSERT INTO Persons VALUES (9, 'James Remo', 'M');
34 INSERT INTO Persons VALUES (10, 'Fletcher Copeland', 'M');
35 INSERT INTO Persons VALUES (11, 'Anula', 'M');
36 INSERT INTO Persons VALUES (12, 'Ten Lama', 'M');
37 INSERT INTO Persons VALUES (13, 'ROSE', 'F');
38 INSERT INTO Persons VALUES (14, 'WOOD', 'M');
```

```

39
40 /* Display Persons Tables */
41 SELECT *
42 FROM PERSONS;
43 GO
44
45 CREATE TABLE Family(
46     personId INT Primary Key REFERENCES Persons(Id),
47     fatherId INT REFERENCES Persons(Id),
48     motherId INT REFERENCES Persons(Id));
49
50 ALTER TABLE Family
51     ADD CONSTRAINT CheckFatherSex CHECK(dbo.CheckPersonSex(fatherId, 'M') = 1),
52     CONSTRAINT CheckMotherSex CHECK(dbo.CheckPersonSex(motherId, 'F') = 1);
53
54 INSERT INTO Family VALUES (1, 11, 2);
55 INSERT INTO Family VALUES (3, 11, 2);
56 INSERT INTO Family VALUES (4, 5, 6);
57 INSERT INTO Family VALUES (12, 5, 6);
58 INSERT INTO Family VALUES (11, 10, 7);
59 INSERT INTO Family VALUES (5, 10, 7);
60 INSERT INTO Family VALUES (2, 14, 13);
61
62 /* Display Family Table with Names */
63 SELECT (SELECT Name FROM Persons WHERE Id = personId) AS 'Child',
64         (SELECT Name FROM Persons WHERE Id = fatherId) AS 'Father',
65         (SELECT Name FROM Persons WHERE Id = motherId) AS 'Mother'
66 FROM Family;
67 GO
68
69 CREATE TABLE Spouses(
70     husbandId INT REFERENCES Persons(Id),
71     wifeId INT REFERENCES Persons(Id),
72     PRIMARY KEY (husbandId, wifeId));
73
74 ALTER TABLE Spouses
75     ADD CONSTRAINT CheckHusbandSex CHECK(dbo.CheckPersonSex(husbandId, 'M') = 1),
76     CONSTRAINT CheckWifeSex CHECK(dbo.CheckPersonSex(wifeId, 'F') = 1);
77

```

```

78 INSERT INTO Spouses VALUES (11, 2);
79 INSERT INTO Spouses VALUES (5, 6);
80 INSERT INTO Spouses VALUES (10, 7);
81 INSERT INTO Spouses VALUES (14, 13);
82
83 /* Display Spouses Table with names */
84 SELECT (SELECT Name FROM Persons WHERE Id = husbandId) AS 'Husband',
85        (SELECT Name FROM Persons WHERE Id = wifeId) AS 'Wife'
86 FROM Spouses;
87 GO
88
89
90
91
92 /* Table Brothers */
93 CREATE TABLE Brothers(
94     childId INT REFERENCES Persons(Id),
95     brotherId INT REFERENCES Persons(Id),
96     PRIMARY KEY (childId, brotherId));
97
98 INSERT INTO Brothers
99 SELECT F1.personId AS 'childId', F2.personId AS 'brotherId'
100 FROM (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'M')) F1,
101      (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'M')) F2
102 WHERE F1.fatherId = F2.fatherId AND F1.MotherId = F2.MotherId AND F1.personId <> F2.personId;
103
104 /* Display Brothers Table with Names */
105 SELECT (SELECT Name FROM Persons WHERE Id = childId) AS 'Brother',
106        (SELECT Name FROM Persons WHERE Id = brotherId) AS 'Brother'
107 FROM Brothers;
108 GO
109
110
111 /* Table Sisters */
112 CREATE TABLE Sisters(
113     childId INT REFERENCES Persons(Id),
114     sisterId INT REFERENCES Persons(Id),
115     PRIMARY KEY (childId, sisterId));
116

```

```

117 INSERT INTO Sisters
118 SELECT F1.personId AS 'childId', F2.personId AS 'sisterId'
119 FROM (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'F')) F1,
120      (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'F')) F2
121 WHERE F1.fatherId = F2.fatherId AND F1.MotherId = F2.MotherId AND F1.personId <> F2.personId;
122
123 /* Display Sisters Table with Names */
124 SELECT (SELECT Name FROM Persons WHERE Id = childId) AS 'Child',
125        (SELECT Name FROM Persons WHERE Id = sisterId) AS 'Sister'
126 FROM Sisters;
127 GO
128
129 /* Table Brother-Sisters */
130 CREATE TABLE BrotherSisters(
131     brotherId INT REFERENCES Persons(Id),
132     sisterId INT REFERENCES Persons(Id),
133     PRIMARY KEY (brotherId, sisterId));
134
135 INSERT INTO BrotherSisters
136 SELECT F1.personId AS 'brotherId', F2.personId AS 'sisterId'
137 FROM (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'M')) F1,
138      (SELECT * FROM Family WHERE personID IN (SELECT Id FROM Persons WHERE Sex = 'F')) F2
139 WHERE F1.fatherId = F2.fatherId AND F1.MotherId = F2.MotherId AND F1.personId <> F2.personId;
140
141 /* Display BrotherSisters Table with Names */
142 SELECT (SELECT Name FROM Persons WHERE Id = brotherId) AS 'Brother',
143        (SELECT Name FROM Persons WHERE Id = sisterId) AS 'Sister'
144 FROM BrotherSisters;
145 GO
146
147 /* Query: Children of a Couple */
148 DECLARE @spouse1Id INT = 11
149 DECLARE @spouse2Id INT = 2
150 SELECT Name AS 'Child', Sex
151 FROM Persons
152 WHERE Id IN (SELECT personId
153             FROM Family
154             WHERE fatherId = @spouse1Id AND motherId = @spouse2Id
155             OR

```



```

156         fatherId = @spouse2Id AND motherId = @spouse1Id);
157
158
159  /* Query: Nephews of a given person */
160  DECLARE @personId INT = 11
161  SELECT Name AS 'Nephew', Sex
162  FROM Persons
163  WHERE Id IN (SELECT personId
164              FROM Family F1
165              WHERE F1.fatherId IN (SELECT brotherId
166                                  FROM Brothers F2
167                                  WHERE F2.childId = @personId and Sex = 'M'));
168  GO
169
170  /* Query: Grandparents of a given person */
171  DECLARE @personId INT = 1
172  SELECT Name AS 'Child',
173         (SELECT Name FROM Persons WHERE Id IN
174          (SELECT fatherId FROM Family F2 WHERE F2.personId IN
175           (SELECT fatherId FROM Family F1 WHERE F1.personId = @personId))) AS 'Paternal Grandfather',
176         (SELECT Name FROM Persons WHERE Id IN
177          (SELECT motherId FROM Family F2 WHERE F2.personId IN
178           (SELECT fatherId FROM Family F1 WHERE F1.personId = @personId))) AS 'Paternal Grandmother',
179         (SELECT Name FROM Persons WHERE Id IN
180          (SELECT fatherId FROM Family F2 WHERE F2.personId IN
181           (SELECT motherId FROM Family F1 WHERE F1.personId = @personId))) AS 'Maternal Grandfather',
182         (SELECT Name FROM Persons WHERE Id IN
183          (SELECT motherId FROM Family F2 WHERE F2.personId IN
184           (SELECT motherId FROM Family F1 WHERE F1.personId = @personId))) AS 'Maternal Grandmother'
185
186  FROM Persons
187  WHERE Id = @personId;
188  GO
189
190
191
192
193
194

```

Output:

Persons Table

	Id	Name	Sex
1	1	Tenzin Tashi	M
2	2	Tsering Dolma	F
3	3	Jetsun Pema	F
4	4	Pasang Lama	F
5	5	James Lama	M
6	6	Jane Lama	F
7	7	Zoe Dunlap	F
8	8	Maximo Mingo	M
9	9	James Remo	M
10	10	Fletcher Copeland	M
11	11	Anula	M
12	12	Khanten Gurung	M
13	13	ROSE	F
14	14	WOOD	M
15	15	Ashley	F

Family Table

	Child	Father	Mother
1	Tenzin Tashi	Anula	Tsering Dolma
2	Tsering Dolma	WOOD	ROSE
3	Jetsun Pema	Anula	Tsering Dolma
4	Pasang Lama	James Lama	Jane Lama
5	James Lama	Fletcher Copeland	Zoe Dunlap
6	Anula	Fletcher Copeland	Zoe Dunlap
7	Khanten Gurung	Anula	Tsering Dolma
8	Ashley	Anula	Tsering Dolma

Husband-Wife Table

	Husband	Wife
1	James Lama	Jane Lama
2	Fletcher Copeland	Zoe Dunlap
3	Anula	Tsering Dolma
4	WOOD	ROSE

Brother-Brother Table

	Brother	Brother
1	Tenzin Tashi	Khanten Gurung
2	James Lama	Anula
3	Anula	James Lama
4	Khanten Gurung	Tenzin Tashi

Sister-Sister Table

	Sister	Sister
1	Jetsun Pema	Ashley
2	Ashley	Jetsun Pema

Brother-Sister Table

	Brother	Sister
1	Tenzin Tashi	Jetsun Pema
2	Tenzin Tashi	Ashley
3	Khanten Gurung	Jetsun Pema
4	Khanten Gurung	Ashley

Child of Given Couple Table

	Child	Sex
1	Tenzin Tashi	M
2	Jetsun Pema	F
3	Khanten Gurung	M
4	Ashley	F

Nephew of a given Person Table

	Nephew	Sex
1	Tenzin Tashi	M
2	Khanten Gurung	M

Grand-Parents of a given Person Table

	Child	Paternal Grandfather	Paternal Grandmother	Maternal Grandfather	Maternal Grandmother
1	Tenzin Tashi	Fletcher Copeland	Zoe Dunlap	WOOD	ROSE