C:\Users\sravani\ DatabaseProject> python main.py

Provided sample input Table:

EmployeeID SSN FirstName LastName JobTitle Manager ManagerEmail Department PhoneNumber Office

0 201 123-45-6789 Yamini Nagaram Engineer Kashif kashif@techm.com IT (555)-123-4567 R1

1 201 123-45-6789 Yamini Nagaram Lead Engineer Raju raju@infosys.com Research (555)-123-4567 R2

2 202 123-45-6700 Sandeep Kumar Manager Kashif kashif@techm.com IT (202) 555-0198 R1

3 203 123-45-6701 Rodger anes Engineer Kashif kashif@techm.com IT (415) 555-0147 R2

4 204 123-45-6702 Dave Brown Engineer Raju raju@infosys.com Design (702) 555-0116 D1

5 205 123-45-6707 Eve Green Analyst Kashif kashif@techm.com IT (617) 555-0132 R1

fds=

{('Office',): ['Department'], ('Manager',): ['ManagerEmail'], ('SSN',): ['EmployeeID', 'FirstName', 'LastName', 'JobTitle', 'Manager', 'ManagerEmail', 'Department', 'PhoneNumber', 'Office'], ('PhoneNumber',): ['EmployeeID']}

Please enter Multi-Valued Dependencies using format X ->> A, B or X ->> Y. After done, type exit or finish: EmployeeID ->> Department

Please enter Multi-Valued Dependencies using format X ->> A, B or X ->> Y. After done, type exit or finish: EmployeeID ->> JobTitle

Please enter Multi-Valued Dependencies using format X ->> A, B or X ->> Y. After done, type exit or finish: Manager ->> JobTitle

Please enter Multi-Valued Dependencies using format X ->> A, B or X ->> Y. After done, type exit or finish: Manager ->> Department

Please enter Multi-Valued Dependencies using format X ->> A, B or X ->> Y. After done, type exit or finish: exit

mvds = {

('EmployeeID'): ['Department', 'JobTitle'],

('Manager'): ['Department', 'JobTitle'],

}

Select the highest normal form that the table can achieve (1: 1NF, 2: 2NF, 3: 3NF, 4: BCNF, 5: 4NF, 6: 5NF): 5

Do you want to know the highest normal form of the input table? (1: Yes, 2: No): 1

Enter primary keys (for composite keys, separate them with commas): SSN

Given input table is already in 1NF.

Given input table is already in 2NF.

Relations after transforming into 3NF:

Office Department

0 R1 IT

1 R2 Research

2 R2 IT

3 D1 Design

Manager Office FirstName SSN LastName PhoneNumber EmployeeID JobTitle ManagerEmail

0 Kashif R1 Yamini 123-45-6789 Nagaram (555)-123-4567 201 Engineer kashif@techm.com

1 Raju R2 Yamini 123-45-6789 Nagaram (555)-123-4567 201 Lead Engineer raju@infosys.com

2 Kashif R1 Sandeep 123-45-6700 Kumar (202) 555-0198 202 Manager kashif@techm.com

3 Kashif R2 Rodger 123-45-6701 anes (415) 555-0147 203 Engineer kashif@techm.com

4 Raju D1 Dave 123-45-6702 Brown (702) 555-0116 204 Engineer raju@infosys.com

5 Kashif R1 Eve 123-45-6707 Green (617) 555-0132 205 Analyst kashif@techm.com

Transformed relations into BCNF:

	Office	Department
0	R1	IT
1	R2	Research
2	R2	IT
3	D1	Design

	Manager	ManagerEmail
0	Kashif	kashif@techm.com
1	Raju	raju@infosys.com

	PhoneNumber	EmployeeID
0	(555)-123-4567	201
2	(202) 555-0198	202
3	(415) 555-0147	203
4	(702) 555-0116	204
5	(617) 555-0132	205

	Manager	Office	FirstName	SSN	LastName	PhoneNumber	JobTitle
0	Kashif	R1	Yamini	123-45-6789	Nagaram	(555)-123-4567	Engineer
1	Raju	R2	Yamini	123-45-6789	Nagaram	(555)-123-4567	Lead Engineer

2 Kashif R1 Sandeep 123-45-6700 Kumar (202) 555-0198 Manager

3 Kashif R2 Rodger 123-45-6701 anes (415) 555-0147 Engineer

4 Raju D1 Dave 123-45-6702 Brown (702) 555-0116 Engineer

5 Kashif R1 Eve 123-45-6707 Green (617) 555-0132 Analyst

Multi-valued dependency violation: ('Manager',) ->> JobTitle

The relation after transforming into 4NF.

Generating output queries for 4NF-->

```
CREATE TABLE Office (
  Office VARCHAR(2) NOT NULL PRIMARY KEY,
  Department VARCHAR(8)
);
CREATE TABLE Manager (
  Manager VARCHAR(6) NOT NULL PRIMARY KEY,
  JobTitle VARCHAR(13)
);
CREATE TABLE PhoneNumber (
  PhoneNumber VARCHAR(14) NOT NULL PRIMARY KEY,
  EmployeeID VARCHAR(3)
);
CREATE TABLE SSN (
  Manager VARCHAR(6),
  Office VARCHAR(2),
  FirstName VARCHAR(7),
  SSN VARCHAR(11) NOT NULL PRIMARY KEY,
  LastName VARCHAR(7),
```

```
  PhoneNumber VARCHAR(14),
  FOREIGN KEY (Manager) REFERENCES Manager(Manager),
  FOREIGN KEY (PhoneNumber) REFERENCES PhoneNumber(PhoneNumber),
  FOREIGN KEY (Office) REFERENCES Office(Office)
);
```

The highest normal form achieved is: 2NF

Normalization process completed.