

A CLIENT AND DONOR DATABASE SYSTEM

Course : Database Management Systems (CS4513)

Semester : Fall

Year : 2017

Instructor: Dr. Le Gruenwald

Author : Vinothini Rajasekaran

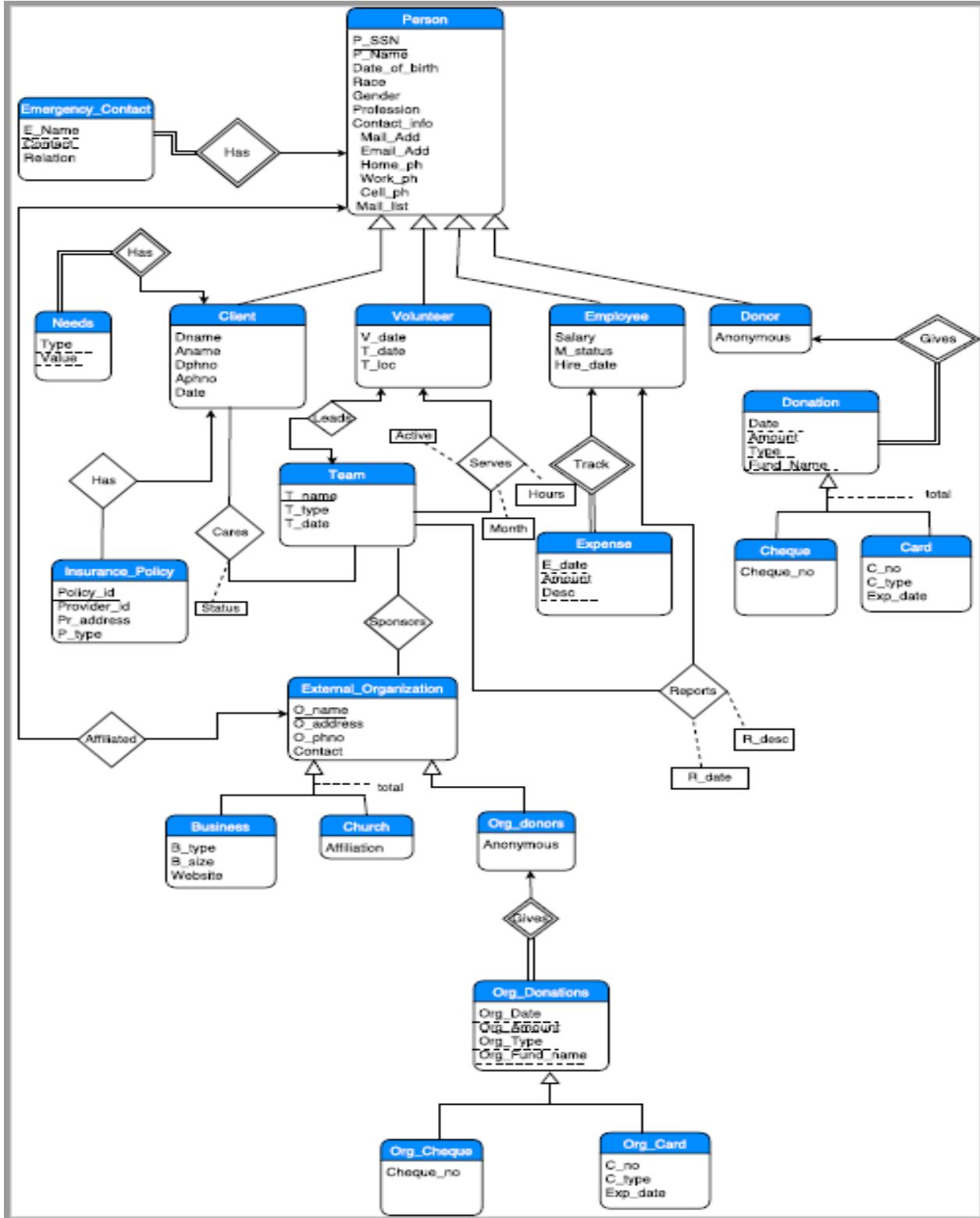
ID : 113423523

Email : vinothini.rajasekaran-1@ou.edu

TASKS PERFORMED	PAGE NUMBER
Task 1.	2-3
1.1 ER Diagram	2
1.2 Relational Database Schema	3
Task 2.	4-17
2.1 Data Dictionary Diagram	4
2.2 Data Dictionary Report	5-17
Task 3.	18-19
3.1 Discussion of Storage Structures for tables	18
3.2 Discussion of Storage Structures for tables (Oracle 12c)	19
Task 4. SQL and text files showing the creation of tables in Oracle 12 c	20-25
Task 5. Java Program	26-43
Task 6. Execution of Java Program	44-94
6.1 Script file showing the testing of query 1	44-45
6.2 Script file showing the testing of query 2	46-51
6.3 Script file showing the testing of query 3	52-57
6.4 Script file showing the testing of query 4	58-60
6.5 Script file showing the testing of query 5	61-66
6.6 Script file showing the testing of query 6	67-68
6.7 Script file showing the testing of query 7	69-71
6.8 Script file showing the testing of query 8	72-78
6.9 Script file showing the testing of query 9	78-82
6.10 Script file showing the testing of query 10	82-83
6.11 Script file showing the testing of query 11	83-84
6.12 Script file showing the testing of query 12	84-85
6.13 Script file showing the testing of query 13	85
6.14 Script file showing the testing of query 14	86
6.15 Script file showing the testing of query 15	86-87
6.16 Script file showing the testing of query 16	87
6.17 Script file showing the testing of query 17	88
6.18 Script file showing the testing of import and export options	88-90
6.19 Script file showing the testing of three types of errors	90-93
6.20 Script file showing the testing of the quit option	94

TASK 1

1.1 ER Diagram



1.2 Relational Database Schema

Person(P_SSN, P_Name, Dob, Race, Gender, Profession, Address, Email, Home_phno, Work_phno, Cell_phno, Mailing_list)

Client(P_SSN, Dname, Aname, Dphno, Aphno, Cdate)

Volunteer(P_SSN, V_date, T_date, T_loc)

Employee(P_SSN, Salary, M_status, Hire_date)

Donor(P_SSN, Anonymous)

Emergency_Contact(P_SSN, E_Name, Contact, Relation)

Needs(P_SSN, Type, Value)

Cheque(P_SSN, Ddate, Amount, Type, Fund_Name, Cheque_no)

Card(P_SSN, Ddate, Amount, Type, Fund_Name, C_no, C_type, Exp_date)

Team(T_name, T_type, T_date)

Report(T_name, P_SSN, R_desc, R_date)

Leads(P_SSN, T_name)

Serves(T_name, P_SSN, Hours, Month, Active)

Expense(P_SSN, E_date, E_desc, E_amount)

Cares(P_SSN, T_name, Status)

Insurance_Policy(Policy_id, P_SSN, Provider_id, Pr_address, P_type)

External_Organization(O_name, Address, Phno, Contact)

Business(O_name, B_type, B_size, Website)

Church(O_name, Affiliation)

Sponsors(O_name, T_name)

Affiliated(P_SSN, O_name)

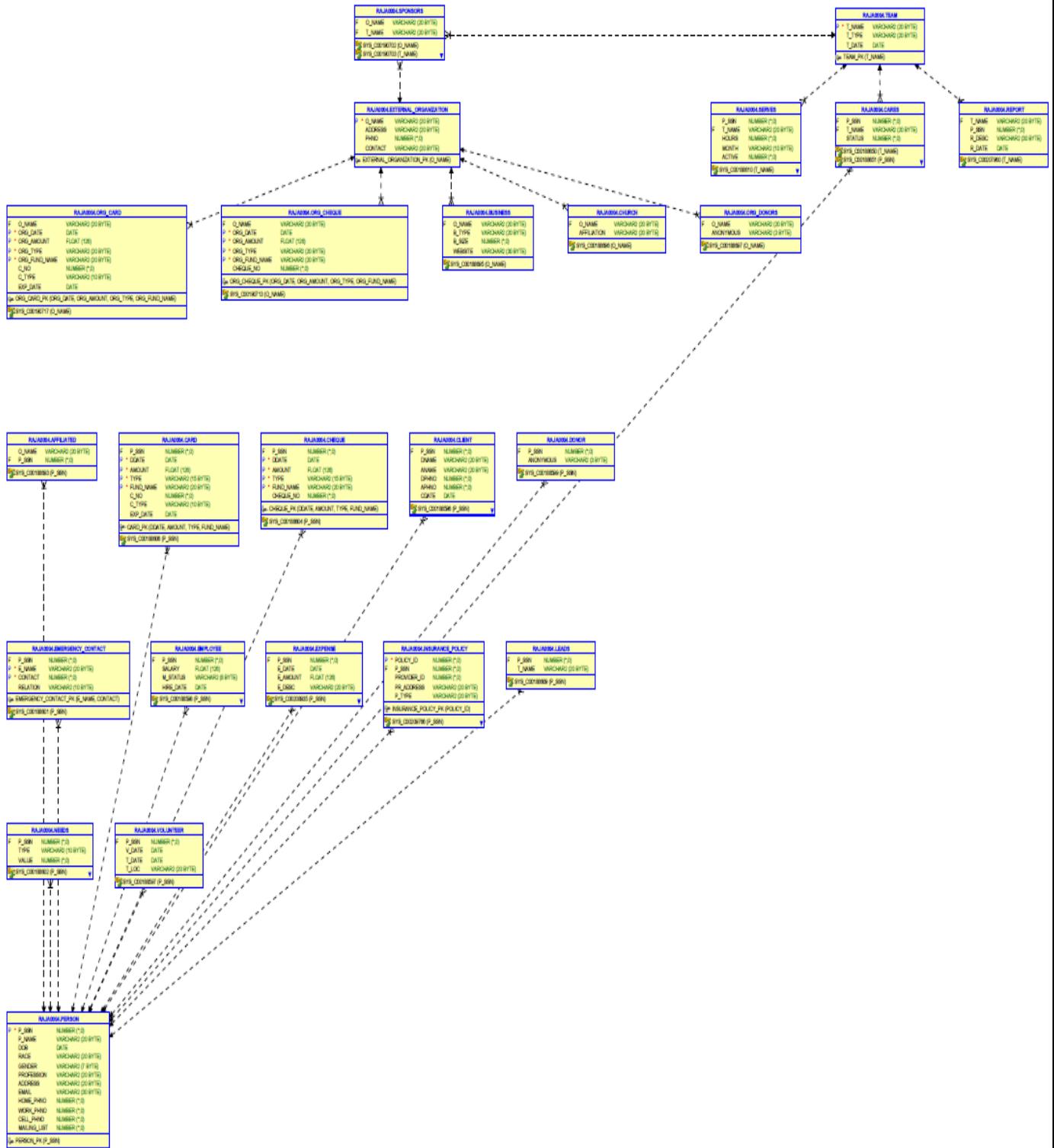
Org_donors(O_name, Anonymous)

Org_Cheque(O_name, Org_Date, Org_Amount, Org_Type, Org_Fund_name, Cheque_no)

Org_Card(O_name, Org_Date, Org_Amount, Org_Type, Org_Fund_name, C_no, C_type, Exp_date)

TASK 2

2.1 Data Dictionary Diagram



2.2 Data Dictionary Report

All Tables Details

Design Name	Untitled_1
Version Date	19.11.2017 12:27:08
Version Comment	
Model Name	Relational_1

Table Name	RAJA0004.AFFILIATED
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	2
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns										
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	O_NAME				VARCHAR (20 BYTE)	LT				
2	P_SSN		F		NUMERIC (*)	LT				

Foreign Keys (referring to)										
Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule			
SYS_C00188693	PERSON		Y		P_SSN	P_SSN				

Table Name	RAJA0004.BUSINESS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	4
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns										
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	O_NAME		F		VARCHAR (20 BYTE)	LT				
2	B_TYPE				VARCHAR (20 BYTE)	LT				
3	B_SIZE				NUMERIC (*)	LT				
4	WEBSITE				VARCHAR (30 BYTE)	LT				

Foreign Keys (referring to)										
Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule			
SYS_C00188695	EXTERNAL_ORGANIZATION		Y		O_NAME	O_NAME				

Table Name	RAJA0004.CARD
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	8
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	P_SSN		F		NUMERIC (*)	LT				
2	DDATE		P	Y	Date (7)	LT				
3	AMOUNT		P	Y	FLOAT (126)	LT				
4	TYPE		P	Y	VARCHAR (15 BYTE)	LT				
5	FUND_NAME		P	Y	VARCHAR (20 BYTE)	LT				
6	C_NO				NUMERIC (*)	LT				
7	C_TYPE				VARCHAR (10 BYTE)	LT				
8	EXP_DATE				Date (7)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
CARD_PK	PK				DDATE	ASC
					AMOUNT	ASC
					TYPE	ASC
					FUND_NAME	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00188606	PERSON			Y	P_SSN	P_SSN	

Table Name	RAJA0004.CARES
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	3
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	P_SSN		F		NUMERIC (*)	LT				
2	T_NAME		F		VARCHAR (20 BYTE)	LT				
3	STATUS				NUMERIC (*)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
C1					P_SSN	ASC
C2					T_NAME	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00188650	TEAM			Y	T_NAME	T_NAME	
SYS_C00188651	PERSON			Y	P_SSN	P_SSN	

Table Name	RAJA0004.CHEQUE
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	6
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	P_SSN		F		NUMERIC (*)	LT				
2	DDATE	P		Y	Date (7)	LT				
3	AMOUNT	P		Y	FLOAT (126)	LT				
4	TYPE	P		Y	VARCHAR (15 BYTE)	LT				
5	FUND_NAME	P		Y	VARCHAR (20 BYTE)	LT				
6	CHEQUE_NO				NUMERIC (*)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
CHEQUE_PK	PK				DDATE	ASC
					AMOUNT	ASC
					TYPE	ASC
					FUND_NAME	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00188604	PERSON		Y		P_SSN	P_SSN	

Table Name	RAJA0004.CHURCH
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	2
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	O_NAME		F		VARCHAR (20 BYTE)	LT				
2	AFFILIATION				VARCHAR (20 BYTE)	LT				

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00188696	EXTERNAL_ORGANIZATION		Y		O_NAME	O_NAME	

Table Name	RAJA0004.CLIENT									
Functional Name										
Abbreviation										
Classification Type Name										
Object Type Name										
Number Of Columns	6									
Number Of Rows Min.	0									
Number Of Rows Max.	9999999									
Expected Number Of Rows	0									
Expected Growth	0									
Growth Interval	Year									
<i>Columns</i>										
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	P_SSN		F		NUMERIC (*)	LT				
2	DNAME				VARCHAR (20 BYTE)	LT				
3	ANAME				VARCHAR (20 BYTE)	LT				
4	DPHNO				NUMERIC (*)	LT				
5	APHNO				NUMERIC (*)	LT				
6	CDATE				Date (7)	LT				
<i>Indexes</i>										
Index Name	State	Functional	Spatial	Expression			Column Name	Sort Order		
SSN							P_SSN			ASC
<i>Foreign Keys (referring to)</i>										
Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule			
SYS_C00188596	PERSON			Y	P_SSN	P_SSN				
<i>Table Name</i>										
Functional Name	RAJA0004.DONOR									
Abbreviation										
Classification Type Name										
Object Type Name										
Number Of Columns	2									
Number Of Rows Min.	0									
Number Of Rows Max.	9999999									
Expected Number Of Rows	0									
Expected Growth	0									
Growth Interval	Year									
<i>Columns</i>										
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	P_SSN		F		NUMERIC (*)	LT				
2	ANONYMOUS				VARCHAR (3 BYTE)	LT				
<i>Foreign Keys (referring to)</i>										
Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule			
SYS_C00188599	PERSON			Y	P_SSN	P_SSN				

RAJA0004.EMERGENCY_CONTACT															
Functional Name															
Abbreviation															
Classification Type Name															
Object Type Name															
Number Of Columns		4													
Number Of Rows Min.		0													
Number Of Rows Max.		9999999													
Expected Number Of Rows		0													
Expected Growth		0													
Growth Interval		Year													
<i>Columns</i>															
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation					
1	P_SSN		F		NUMERIC (*)	LT									
2	E_NAME		P	Y	VARCHAR (20 BYTE)	LT									
3	CONTACT		P	Y	NUMERIC (*)	LT									
4	RELATION				VARCHAR (10 BYTE)	LT									
<i>Indexes</i>															
Index Name			State	Functional	Spatial	Expression		Column Name	Sort Order						
EMERGENCY_CONTACT_PK			PK					E_NAME	ASC						
								CONTACT	ASC						
<i>Foreign Keys (referring to)</i>															
Name	Referring To			Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule						
SYS_C00188601	PERSON				Y		P_SSN	P_SSN							
RAJA0004.EMPLOYEE															
Functional Name															
Abbreviation															
Classification Type Name															
Object Type Name															
Number Of Columns		4													
Number Of Rows Min.		0													
Number Of Rows Max.		9999999													
Expected Number Of Rows		0													
Expected Growth		0													
Growth Interval		Year													
<i>Columns</i>															
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation					
1	P_SSN		F		NUMERIC (*)	LT									
2	SALARY				FLOAT (126)	LT									
3	M_STATUS				VARCHAR (8 BYTE)	LT									
4	HIRE_DATE				Date (7)	LT									
<i>Indexes</i>															
Index Name			State	Functional	Spatial	Expression		Column Name	Sort Order						
SSN1								P_SSN	ASC						
<i>Foreign Keys (referring to)</i>															
Name	Referring To			Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule						
SYS_C00188598	PERSON				Y		P_SSN	P_SSN							

Table Name	RAJA0004.EXPENSE								
Functional Name									
Abbreviation									
Classification Type Name									
Object Type Name									
Number Of Columns	4								
Number Of Rows Min.	0								
Number Of Rows Max.	9999999								
Expected Number Of Rows	0								
Expected Growth	0								
Growth Interval	Year								

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	P_SSN		F		NUMERIC (*)	LT				
2	E_DATE				Date (?)	LT				
3	E_AMOUNT				FLOAT (126)	LT				
4	E_DESC				VARCHAR (20 BYTE)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
EDATE					E_DATE	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00208935	PERSON			Y	P_SSN	P_SSN	

Table Name	RAJA0004.EXTERNAL_ORGANIZATION								
Functional Name									
Abbreviation									
Classification Type Name									
Object Type Name									

Number Of Columns	4								
Number Of Rows Min.	0								
Number Of Rows Max.	9999999								
Expected Number Of Rows	0								
Expected Growth	0								
Growth Interval	Year								

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	O_NAME	P		Y	VARCHAR (20 BYTE)	LT				
2	ADDRESS				VARCHAR (20 BYTE)	LT				
3	PHNO				NUMERIC (*)	LT				
4	CONTACT				VARCHAR (20 BYTE)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
EXTERNAL_ORGANIZATION_PK	PK				O_NAME	ASC

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00188695	BUSINESS			Y	O_NAME	O_NAME	
SYS_C00188696	CHURCH			Y	O_NAME	O_NAME	
SYS_C00188697	ORG_DONORS			Y	O_NAME	O_NAME	
SYS_C00190702	SPONSORS			Y	O_NAME	O_NAME	
SYS_C00190713	ORG_CHEQUE			Y	O_NAME	O_NAME	
SYS_C00190717	ORG_CARD			Y	O_NAME	O_NAME	

Table Name	RAJA0004.INSURANCE_POLICY									
Functional Name										
Abbreviation										
Classification Type Name										
Object Type Name										
Number Of Columns	5									
Number Of Rows Min.	0									
Number Of Rows Max.	9999999									
Expected Number Of Rows	0									
Expected Growth	0									
Growth Interval	Year									
<i>Columns</i>										
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	POLICY_ID	P		Y	NUMERIC (*)	LT				
2	P_SSN		F		NUMERIC (*)	LT				
3	PROVIDER_ID				NUMERIC (*)	LT				
4	PR_ADDRESS				VARCHAR (20 BYTE)	LT				
5	P_TYPE				VARCHAR (20 BYTE)	LT				
<i>Indexes</i>										
Index Name	State	Functional	Spatial	Expression			Column Name	Sort Order		
INSURANCE_POLICY_PK	PK						POLICY_ID		ASC	
INS_TYPE							P_TYPE		ASC	
<i>Foreign Keys (referring to)</i>										
Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule			
SYS_C00209766	PERSON		Y		P_SSN	P_SSN				

Table Name	RAJA0004.LEADS									
Functional Name										
Abbreviation										
Classification Type Name										
Object Type Name										
Number Of Columns	2									
Number Of Rows Min.	0									
Number Of Rows Max.	9999999									
Expected Number Of Rows	0									
Expected Growth	0									
Growth Interval	Year									
<i>Columns</i>										
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	P_SSN		F		NUMERIC (*)	LT				
2	T_NAME				VARCHAR (20 BYTE)	LT				
<i>Foreign Keys (referring to)</i>										
Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule			
SYS_C00188609	PERSON		Y		P_SSN	P_SSN				

Table Name	RAJA0004.NEEDS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	3
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	P_SSN		F		NUMERIC (*)	LT				
2	TYPE				VARCHAR (10 BYTE)	LT				
3	VALUE				NUMERIC (*)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
NEEDS					TYPE	ASC
					VALUE	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00188602	PERSON			Y	P_SSN	P_SSN	

Table Name	RAJA0004.ORG_CARD
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	8
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	O_NAME		F		VARCHAR (20 BYTE)	LT				
2	ORG_DATE	P		Y	Date (7)	LT				
3	ORG_AMOUNT	P		Y	FLOAT (126)	LT				
4	ORG_TYPE	P		Y	VARCHAR (20 BYTE)	LT				
5	ORG_FUND_NAME	P		Y	VARCHAR (20 BYTE)	LT				
6	C_NO				NUMERIC (*)	LT				
7	C_TYPE				VARCHAR (10 BYTE)	LT				
8	EXP_DATE				Date (7)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
ORG_CARD_PK	PK				ORG_DATE	ASC
					ORG_AMOUNT	ASC
					ORG_TYPE	ASC
					ORG_FUND_NAME	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00190717	EXTERNAL_ORGANIZATION			Y	O_NAME	O_NAME	

Table Name	RAJA0004.ORG_CHEQUE
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	
Number Of Columns	6
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	O_NAME		F		VARCHAR (20 BYTE)	LT				
2	ORG_DATE	P		Y	Date (7)	LT				
3	ORG_AMOUNT	P		Y	FLOAT (126)	LT				
4	ORG_TYPE	P		Y	VARCHAR (20 BYTE)	LT				
5	ORG_FUND_NAME	P		Y	VARCHAR (20 BYTE)	LT				
6	CHEQUE_NO				NUMERIC (*)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
ORG_CHEQUE_PK	PK				ORG_DATE	ASC
					ORG_AMOUNT	ASC
					ORG_TYPE	ASC
					ORG_FUND_NAME	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00190713	EXTERNAL_ORGANIZATION		Y		O_NAME	O_NAME	

Table Name	RAJA0004.ORG_DONORS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	
Number Of Columns	2
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	O_NAME		F		VARCHAR (20 BYTE)	LT				
2	ANONYMOUS				VARCHAR (3 BYTE)	LT				

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00188697	EXTERNAL_ORGANIZATION		Y		O_NAME	O_NAME	

Table Name	RAJA0004.PERSON
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	12
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	P_SSN	P		Y	NUMERIC (*)	LT				
2	P_NAME				VARCHAR (20 BYTE)	LT				
3	DOB				Date (?)	LT				
4	RACE				VARCHAR (20 BYTE)	LT				
5	GENDER				VARCHAR (7 BYTE)	LT				
6	PROFESSION				VARCHAR (20 BYTE)	LT				
7	ADDRESS				VARCHAR (20 BYTE)	LT				
8	EMAIL				VARCHAR (30 BYTE)	LT				
9	HOME_PHNO				NUMERIC (*)	LT				
10	WORK_PHNO				NUMERIC (*)	LT				
11	CELL_PHNO				NUMERIC (*)	LT				
12	MAILING_LIST				NUMERIC (*)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
PERSON_PK	PK				P_SSN	ASC

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00188596	CLIENT			Y	P_SSN	P_SSN	
SYS_C00188597	VOLUNTEER			Y	P_SSN	P_SSN	
SYS_C00188598	EMPLOYEE			Y	P_SSN	P_SSN	
SYS_C00188599	DONOR			Y	P_SSN	P_SSN	
SYS_C00188601	EMERGENCY_CONTACT			Y	P_SSN	P_SSN	
SYS_C00188602	NEEDS			Y	P_SSN	P_SSN	
SYS_C00188604	CHEQUE			Y	P_SSN	P_SSN	
SYS_C00188606	CARD			Y	P_SSN	P_SSN	
SYS_C00188609	LEADS			Y	P_SSN	P_SSN	
SYS_C00188651	CARES			Y	P_SSN	P_SSN	
SYS_C00188693	AFFILIATED			Y	P_SSN	P_SSN	
SYS_C00208935	EXPENSE			Y	P_SSN	P_SSN	
SYS_C00209766	INSURANCE_POLICY			Y	P_SSN	P_SSN	

Table Name	RAJA0004.REPORT
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	4
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	T_NAME		F		VARCHAR (20 BYTE)	LT				
2	P_SSN				NUMERIC (*)	LT				
3	R_DESC				VARCHAR (30 BYTE)	LT				
4	R_DATE				Date (?)	LT				

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00207960	TEAM			Y	T_NAME	T_NAME	

Table Name	RAJA0004.SERVES
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	5
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns										
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	P_SSN				NUMERIC (*)	LT				
2	T_NAME		F		VARCHAR (20 BYTE)	LT				
3	HOURS				NUMERIC (*)	LT				
4	MONTH				VARCHAR (10 BYTE)	LT				
5	ACTIVE				NUMERIC (*)	LT				

Indexes							
	Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
T1						T_NAME	ASC
T2						MONTH	ASC

Foreign Keys (referring to)							
Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00188610	TEAM		Y		T_NAME	T_NAME	

Table Name	RAJA0004.SPONSORS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Number Of Columns	2
Number Of Rows Min.	0
Number Of Rows Max.	9999999
Expected Number Of Rows	0
Expected Growth	0
Growth Interval	Year

Columns										
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	O_NAME		F		VARCHAR (20 BYTE)	LT				
2	T_NAME		F		VARCHAR (20 BYTE)	LT				

Indexes							
	Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
ONAME						O_NAME	ASC

Foreign Keys (referring to)							
Name	Referring To	Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule
SYS_C00190702	EXTERNAL_ORGANIZATION		Y		O_NAME	O_NAME	
SYS_C00190703	TEAM		Y		T_NAME	T_NAME	

Table Name	RAJA0004.TEAM											
Functional Name												
Abbreviation												
Classification Type Name												
Object Type Name												
Number Of Columns	3											
Number Of Rows Min.	0											
Number Of Rows Max.	9999999											
Expected Number Of Rows	0											
Expected Growth	0											
Growth Interval	Year											
<i>Columns</i>												
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation		
1	T_NAME	P		Y	VARCHAR (20 BYTE)	LT						
2	T_TYPE				VARCHAR (20 BYTE)	LT						
3	T_DATE				Date (?)	LT						
<i>Indexes</i>												
	Index Name	State	Functional	Spatial	Expression		Column Name		Sort Order			
	TEAM_PK	PK					T_NAME			ASC		
<i>Foreign Keys (referred from)</i>												
Name	Referred From			Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule			
SYS_C00188610	SERVES				Y		T_NAME	T_NAME				
SYS_C00188650	CARES				Y		T_NAME	T_NAME				
SYS_C00190703	SPONSORS				Y		T_NAME	T_NAME				
SYS_C00207960	REPORT				Y		T_NAME	T_NAME				
<i>Table Name</i>												
Table Name	RAJA0004.VOLUNTEER											
Functional Name												
Abbreviation												
Classification Type Name												
Object Type Name												
Number Of Columns	4											
Number Of Rows Min.	0											
Number Of Rows Max.	9999999											
Expected Number Of Rows	0											
Expected Growth	0											
Growth Interval	Year											
<i>Columns</i>												
No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation		
1	P_SSN		F		NUMERIC (*)	LT						
2	V_DATE				Date (?)	LT						
3	T_DATE				Date (?)	LT						
4	T_LOC				VARCHAR (20 BYTE)	LT						
<i>Foreign Keys (referring to)</i>												
Name	Referring To			Mandatory	Transferable	In Arc	Columns	Referred Columns	Delete Rule			
SYS_C00188597	PERSON				Y		P_SSN	P_SSN				

TASK 3

3.1 Discussion of Storage Structures for tables

Table Name	Type of Query	Search Key	Frequency	File Organization
Person	2.Insert 3.Insert 5.Insert 8.Insert 13.Random search 14.Random search 15.Random search	P_SSN P_SSN P_SSN	1/week 2/month 1/year 1/day 1/week 1/week 1/month	Dynamic Hashing using P_SSN as hash key
Client	2.Insert 10.Random Search 17.Random search	P_SSN P_SSN	1/week 1/week 4/year	Dynamic Hashing using P_SSN as hash key
Volunteer	3.Insert		2/month	Heap File
Employee	5.Insert 14.Random search	P_SSN	1/year 1/week	Indexed Seq File with search key as P_SSN as primary index
Donor	8.Insert		1/day	Heap File
Emergency_Contact				Heap File
Needs	17.Random search	Type Value	4/year	Dynamic Hashing with type as primary index and value as secondary index
Cheque	8.Insert		1/day	Heap File
Card	8.Insert		1/day	Heap File
Team	1.Insert		1/month	Heap File
Report	5.Insert		1/year	Heap File
Leads				Heap File
Serves	3.Insert 4.Insert 12.Random search 15.Range search	T_name Month	2/month 30/month 4/year 1/month	B+ Tree with search keys T_name and Month
Expense	6.Insert 11.Range search	E_date	1/day 1/month	B+ Tree with search key as E_date
Cares	2.Insert 12.Random search 13.Random search	P_SSN T_name	1/week 4/year 1/week	Dynamic Hashing with hash key on P_SSN as primary index and T_name as secondary index
Insurance_Policy	17.Random search	Type	4/year	Indexed Seq file with search key on Type as secondary index
External_Organization	7.Insert 9.Insert		2/week 1/day	Heap File
Business				Heap File
Church				Heap File
Sponsors	7.Insert 13.Range search	O_name	2/week 1/week	B+ Tree with search key as O_name
Affiliated				Heap File
Org_donors	9.Insert		1/day	Heap File
Org_Cheque	9.Insert		1/day	Heap File
Org_Card	9.Insert		1/day	Heap File

3.2 Discussion of Storage Structures for tables (Oracle 12c)

Table Name	Type of Query	Search Key	Frequency	File Organization
Person	2.Insert 3.Insert 5.Insert 8.Insert 13.Random search 14.Random search 15.Random search	P_SSN P_SSN P_SSN	1/week 2/month 1/year 1/day 1/week 1/week 1/month	No need of indexing because P_SSN is a primary key indexed by default
Client	2.Insert 10.Random Search 17.Random search	P_SSN P_SSN	1/week 1/week 4/year	Indexing on P_SSN
Volunteer	3.Insert		2/month	No indexing
Employee	5.Insert 14.Random search	P_SSN	1/year 1/week	Indexing on P_SSN
Donor	8.Insert		1/day	No indexing
Emergency_Contact				No indexing
Needs	17.Random search	Type Value	4/year	Indexing on Type and Value
Cheque	8.Insert		1/day	No indexing
Card	8.Insert		1/day	No indexing
Team	1.Insert		1/month	No indexing
Report	5.Insert		1/year	No indexing
Leads				No indexing
Serves	3.Insert 4.Insert 12.Random search 15.Range search	T_name Month	2/month 30/month 4/year 1/month	Indexing on T_name and Month
Expense	6.Insert 11.Range search	E_date	1/day 1/month	Indexing on E_date
Cares	2.Insert 12.Random search 13.Random search	P_SSN T_name	1/week 4/year 1/week	Indexing on P_SSN and T_name
Insurance_Policy	17.Random search	P_type	4/year	Indexing on P_type
External_Organization	7.Insert 9.Insert		2/week 1/day	No indexing
Business				No indexing
Church				No indexing
Sponsors	7.Insert 13.Range search	O_name	2/week 1/week	Indexing on O_name
Affiliated				No indexing
Org_donors	9.Insert		1/day	No indexing
Org_Cheque	9.Insert		1/day	No indexing
Org_Card	9.Insert		1/day	No indexing

TASK 4 - SQL and text files showing the creation of tables in Oracle 12c

//Creating table for Person

```
create table Person(P_SSN integer, P_Name varchar(20), Dob date, Race varchar(20), Gender
varchar(7), Profession varchar(20), Address varchar(20), Email varchar(30), Home_phno
integer, Work_phno integer, Cell_phno integer, Mailing_list boolean, PRIMARY KEY(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER(38,0)	No	(null)	1	(null)
2 P_NAME	VARCHAR2(20 BYTE)	Yes	(null)	2	(null)
3 DOB	DATE	Yes	(null)	3	(null)
4 RACE	VARCHAR2(20 BYTE)	Yes	(null)	4	(null)
5 GENDER	VARCHAR2(7 BYTE)	Yes	(null)	5	(null)
6 PROFESSION	VARCHAR2(20 BYTE)	Yes	(null)	6	(null)
7 ADDRESS	VARCHAR2(20 BYTE)	Yes	(null)	7	(null)
8 EMAIL	VARCHAR2(30 BYTE)	Yes	(null)	8	(null)
9 HOME_PHNO	NUMBER(38,0)	Yes	(null)	9	(null)
10 WORK_PHNO	NUMBER(38,0)	Yes	(null)	10	(null)
11 CELL_PHNO	NUMBER(38,0)	Yes	(null)	11	(null)
12 MAILING_LIST	NUMBER(38,0)	Yes	(null)	12	(null)

//Creating table for Client

```
create table Client(P_SSN integer, Dname varchar(20), Aname varchar(20), Dphno integer, Aphno
integer, Cdate date, FOREIGN KEY(P_SSN) REFERENCES Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER(38,0)	Yes	(null)	1	(null)
2 DNAME	VARCHAR2(20 BYTE)	Yes	(null)	2	(null)
3 ANAME	VARCHAR2(20 BYTE)	Yes	(null)	3	(null)
4 DPHNO	NUMBER(38,0)	Yes	(null)	4	(null)
5 APHNO	NUMBER(38,0)	Yes	(null)	5	(null)
6 CDATE	DATE	Yes	(null)	6	(null)

//Creating table for Volunteer

```
create table Volunteer(P_SSN integer, V_date date, T_date date, T_loc varchar(20), FOREIGN
KEY(P_SSN) REFERENCES Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER(38,0)	Yes	(null)	1	(null)
2 V_DATE	DATE	Yes	(null)	2	(null)
3 T_DATE	DATE	Yes	(null)	3	(null)
4 T_LOC	VARCHAR2(20 BYTE)	Yes	(null)	4	(null)

//Creating table for Employee

```
create table Employee(P_SSN integer, Salary float,M_status varchar(8),Hire_date date, FOREIGN KEY(P_SSN) REFERENCES Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER (38, 0)	Yes	(null)	1	(null)
2 SALARY	FLOAT	Yes	(null)	2	(null)
3 M_STATUS	VARCHAR2 (8 BYTE)	Yes	(null)	3	(null)

//Creating table for Donor

```
create table Donor(P_SSN integer, Anonymous varchar(3), FOREIGN KEY(P_SSN) REFERENCES Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER (38, 0)	Yes	(null)	1	(null)
2 ANONYMOUS	VARCHAR2 (3 BYTE)	Yes	(null)	2	(null)

//Creating table for Emergency_Contact

```
create table Emergency_Contact(P_SSN integer, E_Name varchar(20),Contact integer,Relation varchar(10), PRIMARY KEY(E_Name,Contact),FOREIGN KEY(P_SSN) REFERENCES Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER (38, 0)	Yes	(null)	1	(null)
2 E_NAME	VARCHAR2 (20 BYTE)	No	(null)	2	(null)
3 CONTACT	NUMBER (38, 0)	No	(null)	3	(null)
4 RELATION	VARCHAR2 (10 BYTE)	Yes	(null)	4	(null)

//Creating table for Needs

```
create table Needs(P_SSN integer, Type varchar(10),Value integer, FOREIGN KEY(P_SSN) REFERENCES Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER (38, 0)	Yes	(null)	1	(null)
2 VALUE	NUMBER (38, 0)	Yes	(null)	2	(null)
3 TYPE	VARCHAR2 (30 BYTE)	Yes	(null)	3	(null)

//Creating table for Cheque

```
create table Cheque(P_SSN integer, Ddate date,Amount float,Type varchar(15),Fund_Name varchar(20),Cheque_no integer,PRIMARY KEY(Ddate,Amount,Type,Fund_Name), FOREIGN KEY(P_SSN) REFERENCES Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER (38, 0)	Yes	(null)	1	(null)
2 DDATE	DATE	No	(null)	2	(null)
3 AMOUNT	FLOAT	No	(null)	3	(null)
4 TYPE	VARCHAR2 (15 BYTE)	No	(null)	4	(null)
5 FUND_NAME	VARCHAR2 (20 BYTE)	No	(null)	5	(null)
6 CHEQUE_NO	NUMBER (38, 0)	Yes	(null)	6	(null)

//Creating table for Card

```
create table Card(P_SSN integer, Ddate date,Amount float,Type varchar(15),Fund_Name  
varchar(20),C_no integer,C_type varchar(10),Exp_date date,PRIMARY  
KEY(Ddate,Amount,Type,Fund_Name), FOREIGN KEY(P_SSN) REFERENCES Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER (38, 0)	Yes	(null)	1	(null)
2 DDATE	DATE	No	(null)	2	(null)
3 AMOUNT	FLOAT	No	(null)	3	(null)
4 TYPE	VARCHAR2 (15 BYTE)	No	(null)	4	(null)
5 FUND_NAME	VARCHAR2 (20 BYTE)	No	(null)	5	(null)
6 C_NO	NUMBER (38, 0)	Yes	(null)	6	(null)
7 C_TYPE	VARCHAR2 (10 BYTE)	Yes	(null)	7	(null)
8 EXP_DATE	DATE	Yes	(null)	8	(null)

//Creating table for Team

```
create table Team( T_name varchar(20),T_type varchar(20),T_date date,R_desc varchar(20),R_date date,  
PRIMARY KEY(T_name));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 T_NAME	VARCHAR2 (20 BYTE)	No	(null)	1	(null)
2 T_TYPE	VARCHAR2 (20 BYTE)	Yes	(null)	2	(null)
3 T_DATE	DATE	Yes	(null)	3	(null)

//Creating table for Leads

```
create table Leads(P_SSN integer, T_name varchar(20),FOREIGN KEY(P_SSN) REFERENCES  
Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER (38, 0)	Yes	(null)	1	(null)
2 T_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	2	(null)

//Creating table for Serves

```
create table Serves(P_SSN integer, T_name varchar(20),Hours integer,Month varchar(10),Active  
integer,FOREIGN KEY(T_name) REFERENCES Team(T_name));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER (38, 0)	Yes	(null)	1	(null)
2 T_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	2	(null)
3 HOURS	NUMBER (38, 0)	Yes	(null)	3	(null)
4 ACTIVE	NUMBER (38, 0)	Yes	(null)	4	(null)
5 MONTH	VARCHAR2 (10 BYTE)	Yes	(null)	5	(null)

//Creating table for Cares

```
create table Cares(P_SSN integer, T_name varchar(20),status integer,FOREIGN KEY(T_name)  
REFERENCES Team(T_name),FOREIGN KEY(P_SSN) REFERENCES Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER (38, 0)	Yes	(null)	1	(null)
2 T_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	2	(null)
3 STATUS	NUMBER (38, 0)	Yes	(null)	3	(null)

//Creating table for External_Organization

```
create table External_Organization(O_name varchar(20),Address varchar(20),Phno integer,Contact
varchar(20), PRIMARY KEY(O_name));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 O_NAME	VARCHAR2 (20 BYTE)	No	(null)	1	(null)
2 ADDRESS	VARCHAR2 (20 BYTE)	Yes	(null)	2	(null)
3 PHNO	NUMBER (38, 0)	Yes	(null)	3	(null)
4 CONTACT	VARCHAR2 (20 BYTE)	Yes	(null)	4	(null)

//Creating table for Business

```
create table Business(O_name varchar(20),B_type varchar(20),B_size integer,Website
varchar(30),FOREIGN KEY(O_name) REFERENCES External_Organization(O_name));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 O_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	1	(null)
2 B_TYPE	VARCHAR2 (20 BYTE)	Yes	(null)	2	(null)
3 B_SIZE	NUMBER (38, 0)	Yes	(null)	3	(null)
4 WEBSITE	VARCHAR2 (30 BYTE)	Yes	(null)	4	(null)

//Creating table for Church

```
create table Church(O_name varchar(20),Affiliation varchar(20),FOREIGN KEY(O_name)
REFERENCES External_Organization(O_name));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 O_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	1	(null)
2 AFFILIATION	VARCHAR2 (20 BYTE)	Yes	(null)	2	(null)

//Creating table for Affiliated

```
create table Affiliated(O_name varchar(20),P_SSN integer,FOREIGN KEY(P_SSN) REFERENCES
Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 O_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	1	(null)
2 P_SSN	NUMBER (38, 0)	Yes	(null)	2	(null)

//Creating table for Org_donors

```
create table Org_donors(O_name varchar(20), Anonymous varchar(3),FOREIGN KEY(O_name)
REFERENCES External_organization(O_name));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 O_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	1	(null)
2 ANONYMOUS	VARCHAR2 (3 BYTE)	Yes	(null)	2	(null)

//Creating table for Sponsors

```
create table Sponsors(O_name varchar(20),T_name varchar(20),FOREIGN KEY(O_name)
REFERENCES External_Organization(O_name),FOREIGN KEY(T_name) REFERENCES
Team(T_name));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 O_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	1	(null)
2 T_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	2	(null)

//Creating table for Org_Cheque

```
create table Org_Cheque(O_name varchar(20), Org_Date date,Org_Amount float,Org_Type
varchar(20),Org_Fund_name varchar(20),Cheque_no integer,PRIMARY
KEY(Org_Date,Org_Amount,Org_Type,Org_Fund_name),FOREIGN KEY(O_name) REFERENCES
External_organization(O_name));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 O_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	1	(null)
2 ORG_DATE	DATE	No	(null)	2	(null)
3 ORG_AMOUNT	FLOAT	No	(null)	3	(null)
4 ORG_TYPE	VARCHAR2 (20 BYTE)	No	(null)	4	(null)
5 ORG_FUND_NAME	VARCHAR2 (20 BYTE)	No	(null)	5	(null)
6 CHEQUE_NO	NUMBER (38, 0)	Yes	(null)	6	(null)

//Creating table for Org_Card

```
create table Org_Card(O_name varchar(20), Org_Date date,Org_Amount float,Org_Type
varchar(20),Org_Fund_name varchar(20),C_no integer,C_type varchar(10),Exp_date date,PRIMARY
KEY(Org_Date,Org_Amount,Org_Type,Org_Fund_name),FOREIGN KEY(O_name) REFERENCES
External_organization(O_name));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 O_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	1	(null)
2 ORG_DATE	DATE	No	(null)	2	(null)
3 ORG_AMOUNT	FLOAT	No	(null)	3	(null)
4 ORG_TYPE	VARCHAR2 (20 BYTE)	No	(null)	4	(null)
5 ORG_FUND_NAME	VARCHAR2 (20 BYTE)	No	(null)	5	(null)
6 C_NO	NUMBER (38, 0)	Yes	(null)	6	(null)
7 C_TYPE	VARCHAR2 (10 BYTE)	Yes	(null)	7	(null)
8 EXP_DATE	DATE	Yes	(null)	8	(null)

//Creating table for Report

```
create table Report(T_name varchar(20),P_SSN integer,R_desc varchar(30),R_date date, FOREIGN
KEY(T_name) REFERENCES Team(T_name));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 T_NAME	VARCHAR2 (20 BYTE)	Yes	(null)	1	(null)
2 P_SSN	NUMBER (38, 0)	Yes	(null)	2	(null)
3 R_DESC	VARCHAR2 (30 BYTE)	Yes	(null)	3	(null)
4 R_DATE	DATE	Yes	(null)	4	(null)

//Creating table for Expense

```
create table Expense(P_SSN integer,E_date date,E_amount float,E_desc varchar(20), PRIMARY KEY(E_date,E_amount,E_desc),FOREIGN KEY(P_SSN) REFERENCES Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 P_SSN	NUMBER (38, 0)	Yes	(null)	1	(null)
2 E_DATE	DATE	No	(null)	2	(null)
3 E_AMOUNT	FLOAT	No	(null)	3	(null)
4 E_DESC	VARCHAR2 (20 BYTE)	No	(null)	4	(null)

//Creating table for Insurance_policy

```
create table Insurance_policy(Policy_id integer,P_SSN integer,Provider_id integer,Pr_address varchar(20),P_type varchar(20), PRIMARY KEY(Policy_id),FOREIGN KEY(P_SSN) REFERENCES Person(P_SSN));
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 POLICY_ID	NUMBER (38, 0)	No	(null)	1	(null)
2 P_SSN	NUMBER (38, 0)	Yes	(null)	2	(null)
3 PROVIDER_ID	NUMBER (38, 0)	Yes	(null)	3	(null)
4 PR_ADDRESS	VARCHAR2 (20 BYTE)	Yes	(null)	4	(null)
5 P_TYPE	VARCHAR2 (20 BYTE)	Yes	(null)	5	(null)

//Creating indices for tables

```
create index ssn on Client(P_SSN);
```

```
create index ssn1 on Employee(P_SSN);
```

```
create index t1 on Serves(T_name);
```

```
create index t2 on Serves(Month);
```

```
create index needs on Needs(Type,Value);
```

```
create index Edate on Expense(E_date);
```

```
create index c1 on Cares(P_SSN);
```

```
create index c2 on Cares(T_name);
```

```
create index ins_type on Insurance_Policy(P_type);
```

```
create index oname on Sponsors(O_name);
```

TASK 5 Java Program

//Java Program connecting to the database

```
package testjdbc;  
import java.io.BufferedReader;  
import java.io.BufferedWriter;  
import java.io.DataInputStream;  
import java.io.FileInputStream;  
import java.io.FileWriter;  
import java.io.IOException;  
import java.io.InputStreamReader;  
import java.sql.*;  
import java.util.Scanner;  
  
public class Ip {  
    //For reading the input from user for the function input  
    static Scanner sc=new Scanner(System.in);  
    public static int input() throws SQLException {  
        //Creating database connection  
        Connection dbc=DriverManager.getConnection("jdbc:oracle:thin:@//oracle.cs.ou.edu:1521/  
        pdborcl.cs.ou.edu","raja0004","IRsj1Xj1");  
        //Creating statement object  
        Statement st=dbc.createStatement();  
        //Reading the input from user for person table..  
        System.out.println("Enter the Person details first\n");  
        System.out.println("Enter the Person SSN");  
        int p1=sc.nextInt();  
        System.out.println("Enter the Person name");  
        String p2=sc.next();  
        System.out.println("Enter the Person DOB");
```

```

String p3=sc.next();
System.out.println("Enter the Person Race");
String p4=sc.next();
System.out.println("Enter the Person Gender");
String p5=sc.next();
System.out.println("Enter the Person Profession");
String p6=sc.next();
System.out.println("Enter the Person Contact details..\n");
System.out.println("Enter the Person mailing address");
String p7=sc.next();
System.out.println("Enter the Person Email");
String p8=sc.next();
System.out.println("Enter the Person Home_ph_no");
int p9=sc.nextInt();
System.out.println("Enter the Person Work_ph_no");
int p10=sc.nextInt();
System.out.println("Enter the Person Cell_ph_no");
int p11=sc.nextInt();
System.out.println("Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news
letter list ");
int p12=sc.nextInt();
//Inserting the values into person table
st.executeUpdate("insert into Person
values("+p1+","+p2+","+p3+","+p4+","+p5+","+p6+","+p7+","+p8+","+p9+","+p10+","+p11+
","+p12+");";
return p1;
}

public static void main(String[] args) throws IOException {
//To read the input from user at runtime
BufferedReader r=new BufferedReader(new InputStreamReader(System.in));
int n;

```

```
//Loading Oracle Driver

try {
    Class.forName("oracle.jdbc.OracleDriver");
    System.out.println("Connected");
}

catch(Exception e) {
    System.out.println("Unable to load driver class");
}

//Connecting to the Database

try {
    Connection
    dbc=DriverManager.getConnection("jdbc:oracle:thin:@//oracle.cs.ou.edu:1521/pdborcl.cs.ou.edu","raja0
004","IRsj1Xj1");

    //Creating Statement object

    Statement st=dbc.createStatement();
    System.out.println("WELCOME TO THE PAN CLIENT AND DONOR DATABASE
SYSTEM!!!\n"
+ "1.Enter a new team into the database\n"
+ "2.Enter a new client into the database and associate him or her with one or more teams\n"
+ "3.Enter a new volunteer into the database and associate him or her with one or more teams\n"
+ "4.Enter the number of hours a volunteer worked this month for a particular team\n"
+ "5.Enter a new employee into the database and associate him or her with one or more teams\n"
+ "6.Enter an expense charged by an employee\n"
+ "7.Enter a new organization and associate it to one or more PAN teams\n"
+ "8.Enter a new donor and associate him or her with several donations\n"
+ "9.Enter a new organization and associate it with several donations\n"
+ "10.Retrieve the name and phone number of the doctor of a particular client\n"
```

- + "11.Retrieve the total amount of expenses charged by each employee for a particular period of time. The list should be sorted by the total amount of expenses\n"
- + "12.Retrieve the list of volunteers that are members of teams that support a particular client\n"
- + "13.Retrieve the names and contact information of the clients that are supported by teams sponsored by an organization whose name starts with a letter between B and K. The client list should be sorted by name\n"
- + "14.Retrieve the name and total amount donated by donors that are also employees. The list should be sorted by the total amount of the donations, and indicate if each donor wishes to remain anonymous\n"
- + "15.For each team, retrieve the name and associated contact information of the volunteer that has worked the most total hours between March and June\n"
- + "16.Increase the salary by 10% of all employees to whom more than one team must report\n"
- + "17.Delete all clients who do not have health insurance and whose value of importance for transportation is less than 5\n"
- + "18.Import option-Entering Team values from a file\n"
- + "19.Export option-Retrieve all data from Team and write to a file\n"
- + "20.Exit from program");

```

do{
    System.out.println("Enter your option....");
    n=Integer.parseInt(r.readLine());
    switch(n) {
        //Entering a new team
        case 1: System.out.println("Enter the Team name");
        String s1=(r.readLine());
        System.out.println("Enter the Team type");
        String s2=(r.readLine());
        System.out.println("Enter the Team formed date");
        String s3=(r.readLine());
        st.executeUpdate("insert into Team values(\""+s1+"','"+s2+"','"+s3+"\")");
        break;
        case 2: //Entering a new client and associating clients to teams
        int ssn=input();
    }
}

```

```
System.out.println("Now Enter the Client details....");
System.out.println("Enter the Doctor name");
BufferedReader re=new BufferedReader(new InputStreamReader(System.in));
String c2=(re.readLine());
System.out.println("Enter the Attorney name");
String c3=(re.readLine());
System.out.println("Enter the Doctor phone number");
int c4=Integer.parseInt(re.readLine());
System.out.println("Enter the Attorney phone number");
int c5=Integer.parseInt(re.readLine());
System.out.println("Enter the date client was assigned to organization");
String c6=(re.readLine());
st.executeUpdate("insert into Client values
                ("+ssn+"','"+c2+"','"+c3+"','"+c4+"','"+c5+"','"+c6+"')");
System.out.println("Now Associating clients to teams");
System.out.println("Retrieving team names...");
ResultSet rs=st.executeQuery("select t_name from Team");
System.out.println("Tname\n");
while(rs.next()) {
    System.out.println(rs.getString(1));
}
System.out.println("Enter the number of teams to be linked to");
int co=Integer.parseInt(re.readLine());
while(co!=0) {
    System.out.println("Enter the team name to be associated with");
    String tn=(re.readLine());
    int str=0;
    st.executeUpdate("insert into Cares values("+ssn+","+tn+","+str+ ")");
    co--;
}
break;
```

```

//Entering a new volunteer and associating volunteer to teams

case 3: int vp=input();

        System.out.println("");
        System.out.println("Enter the Volunteer details..");
        System.out.println("Enter the Volunteer joining date");
        String v1=(r.readLine());
        System.out.println("Enter the Volunteer training date");
        String v2=(r.readLine());
        System.out.println("Enter the Volunteer training location");
        String v3=(r.readLine());
        st.executeUpdate("insert into Volunteer values(\"+vp+\";\"+v1+\";\"+v2+\";\"+v3+)");

        ResultSet rs1=st.executeQuery("select t_name from Team");

        System.out.println("Tname\n");
        while(rs1.next()) {
            System.out.println(rs1.getString(1));
        }

        System.out.println("Enter the number of teams to be linked to");
        int vo=Integer.parseInt(r.readLine());
        while(vo!=0) {
            System.out.println("Enter the team name to be associated with");
            String tt=(r.readLine());
            System.out.println("Enter the no of hours he worked");
            int t=Integer.parseInt(r.readLine());
            System.out.println("Enter the value for month");
            String tm=(r.readLine());
            st.executeUpdate("insert into Serves values(\"+vp+\";\"+tt+\";\"+t+\";1,\"+tm+)");

            vo--;
        }
        break;

//Entering no of hours for volunteer data for particular client

case 4: System.out.println("Displaying Volunteer details...");



```

```

ResultSet rs2=st.executeQuery("select P_SSN,P_Name from Person where
                                P_SSN in (select P_SSN from Volunteer)");
System.out.println("P_SSN\tName\n");
while(rs2.next()) {
    System.out.println(rs2.getString(1)+"\t"+rs2.getString(2));
}
System.out.println("Displaying team names... ");
ResultSet rs3=st.executeQuery("select t_name from Team");
System.out.println("Tname\n");
while(rs3.next()) {
    System.out.println(rs3.getString(1));
}
System.out.println("Enter the SSN");
int ps=Integer.parseInt(r.readLine());
System.out.println("Enter the team name to be associated with");
String an=(r.readLine());
System.out.println("Enter the no of hours he worked");
int ho=Integer.parseInt(r.readLine());
System.out.println("Enter the value for month");
String m=(r.readLine());
st.executeUpdate("insert into Serves values(\""+ps+"','"+an+"','"+ho+"",1,""+m+"")");
break;
//Entering a new employee and associating employees to teams
case 5: int es=input();
System.out.println("Enter the Employee details");
System.out.println("Enter the salary of Employee");
float sa=Float.parseFloat(r.readLine());
System.out.println("Enter the Marital_status of Employee");
String ms=(r.readLine());
System.out.println("Enter the hire_date of the Employee");

```

```

String hd=(r.readLine());
st.executeUpdate("insert into Employee values(\"+es\",""+sa+"\",""+ms+"\",""+hd+"")");
ResultSet rs4=st.executeQuery("select t_name from Team");
System.out.println("Tname\n");
while(rs4.next()) {
    System.out.println(rs4.getString(1));
}
System.out.println("Enter the number of teams to be linked to");
int no=Integer.parseInt(r.readLine());
while(no!=0) {
    System.out.println("Enter the team name to be associated with");
    String ro=(r.readLine());
    System.out.println("Enter the report description");
    String r1=(r.readLine());
    System.out.println("Enter the reporting date");
    String r2=(r.readLine());
    st.executeUpdate("insert into Report values(\"+ro\",""+es+"\",""+r1+"\",""+r2+"")");
    no--;
}
break;
//Entering the expenses charged by employee
case 6: ResultSet rs5=st.executeQuery("select P_SSN from Employee");
System.out.println("Employess SSN\n");
while(rs5.next()) {
    System.out.println(rs5.getString(1));
}
System.out.println("Enter the employee SSN for whom expense to be recorded");
String sn=(r.readLine());
System.out.println("Enter the expense date");
String ed=(r.readLine());

```

```

System.out.println("Enter the expense amount");
String ea=(r.readLine());
System.out.println("Enter the expense description");
String de=(r.readLine());
st.executeUpdate("insert into Expense values(\"+sn+\" , \"+ed+\" , \"+ea+\" , \"+de+)\")");
break;
//Entering a new organization and associating it to teams
case 7: System.out.println("Enter the organization name");
String on=(r.readLine());
System.out.println("Enter the organization address");
String oa=(r.readLine());
System.out.println("Enter the organization phone no");
String op=(r.readLine());
System.out.println("Enter the organization contact person");
String oc=(r.readLine());
st.executeUpdate("insert into External_Organization
values(\"+on+\" , \"+oa+\" , \"+op+\" , \"+oc+)\")");
ResultSet rs6=st.executeQuery("select t_name from Team");
System.out.println("Tname\n");
while(rs6.next()) {
System.out.println(rs6.getString(1));
}
System.out.println("Enter the number of teams to be linked to");
int o1=Integer.parseInt(r.readLine());
while(o1!=0) {
System.out.println("Enter the team name to be associated with");
String ot=(r.readLine());
st.executeUpdate("insert into Sponsors values(\"+on+\" , \"+ot+)\")");
o1--;
}
break;

```

```

//Entering a new donor and associating donors to donations

case 8: int ds=input();

        System.out.println("Donor details....");

        System.out.println("Enter whether the donor wishes to remain anonymous: yes or
                           no");

        String ano=(r.readLine());

        st.executeUpdate("insert into Donor values(\""+ds+"','"+ano+"\")");

        System.out.println("Donation details..");

        System.out.println("Enter the no of donations");

        int dn=Integer.parseInt(r.readLine());

        while(dn!=0) {

            System.out.println("Enter 1 for cheque and 2 for Card...");

            int ch=Integer.parseInt(r.readLine());

            if(ch==1) {

                System.out.println("Enter the date of donation");

                String ddo=(r.readLine());

                System.out.println("Enter the donation amount");

                float dda=Float.parseFloat(r.readLine());

                System.out.println("Enter the type of donation");

                String dt=(r.readLine());

                System.out.println("Enter the fund name of donation");

                String fn=(r.readLine());

                System.out.println("Enter the cheque no");

                int che=Integer.parseInt(r.readLine());

                st.executeUpdate("insert into Cheque values
                               (""+ds+"','"+ddo+"','"+dda+"','"+dt+"','"+fn+"','"+che+"")");

            }

            else {

                System.out.println("Enter the date of donation");

                String ddo=(r.readLine());

```

```

        System.out.println("Enter the donation amount");
        float dda=Float.parseFloat(r.readLine());
        System.out.println("Enter the type of donation");
        String dt=(r.readLine());
        System.out.println("Enter the fund name of donation");
        String fn=(r.readLine());
        System.out.println("Enter the card no");
        int car=Integer.parseInt(r.readLine());
        System.out.println("Enter the card type");
        String cty=(r.readLine());
        System.out.println("Enter the card expiry date");
        String exp=(r.readLine());
        st.executeUpdate("insert into Card values
        (""+ds+"",""+ddo+"",""+dda+"",""+dt+"",""+fn+"",""+car+"",""+cty+"",""+exp+"")");
        }dn--;
    }
    break;
    //Entering a new organization and associating it to donations
    case 9: System.out.println("Enter the organization name");
    String on1=(r.readLine());
    System.out.println("Enter the organization address");
    String oa1=(r.readLine());
    System.out.println("Enter the organization phone no");
    String op1=(r.readLine());
    System.out.println("Enter the organization contact person");
    String oc1=(r.readLine());
    st.executeUpdate("insert into External_Organization
    values(""+on1+"','"+oa1+"','"+op1+"','"+oc1+"')");
    System.out.println("Organization Donor details....");
    System.out.println("Enter whether the org_donor wishes to remain anonymous:
    yes or no");

```

```
String ano1=(r.readLine());
st.executeUpdate("insert into Org_donors values(\"+on1+\""+ano1+"\")");
System.out.println("Org_Donation details..");
System.out.println("Enter the no of donations");
int nd=Integer.parseInt(r.readLine());
while(nd!=0) {
    System.out.println("Enter 1 for cheque and 2 for Card...");
    int ch1=Integer.parseInt(r.readLine());
    if(ch1==1) {
        System.out.println("Enter the date of donation");
        String ddo=(r.readLine());
        System.out.println("Enter the donation amount");
        float dda=Float.parseFloat(r.readLine());
        System.out.println("Enter the type of donation");
        String dt=(r.readLine());
        System.out.println("Enter the fund name of donation");
        String fn=(r.readLine());
        System.out.println("Enter the cheque no");
        int che=Integer.parseInt(r.readLine());
        st.executeUpdate("insert into Org_Cheque
values(\"+on1+\""+ddo+"\""+dda+"\""+dt+"\""+fn+"\""+che+"\")");
    }
    else {
        System.out.println("Enter the date of donation");
        String ddo=(r.readLine());
        System.out.println("Enter the donation amount");
        float dda=Float.parseFloat(r.readLine());
        System.out.println("Enter the type of donation");
        String dt=(r.readLine());
        System.out.println("Enter the fund name of donation");
    }
}
```

```

        String fn=(r.readLine());
        System.out.println("Enter the card no");
        int car=Integer.parseInt(r.readLine());
        System.out.println("Enter the card type");
        String cty=(r.readLine());
        System.out.println("Enter the card expiry date");
        String exp=(r.readLine());
        st.executeUpdate("insert into Org_Card
values(""+on1+"','"+ddo+"','"+dda+"','"+dt+"','"+fn+"','"+car+"','"+cty+"','"+exp+"')");
    } nd--;
    break;
    //Retrieving the name and phoneno of the doctor for particular client
case 10: System.out.println("Displaying Client SSNs..");
    ResultSet rs7=st.executeQuery("select P_SSN from Client");
    System.out.println("Clients SSN\n");
    while(rs7.next()) {
        System.out.println(rs7.getString(1));
    }
    System.out.println("Enter the Client SSN to get the doctor details");
    int dssn=Integer.parseInt(r.readLine());
    ResultSet rs8=st.executeQuery("select Dname,Dphno from Client where
P_SSN='"+dssn+"'");
    System.out.println("Doctor Name"+'\t'+"Doctor phone_number");
    while(rs8.next()) {
        System.out.println(rs8.getString(1)+"\t"+rs8.getString(2));
    }
    break;
    //Retrieving the total expense charged by employee for a period of time
case 11: System.out.println("Enter the date range to retrieve the employee expense");
    System.out.println("Enter the starting date");
    String sdate=(r.readLine());

```

```

        System.out.println("Enter the ending date");
        String edate=(r.readLine());
        System.out.println("Employee_SSN"+'\t'+"Total expense");
        ResultSet rs9=st.executeQuery("select P_SSN,sum(E_amount) from expense
where E_date between "+sdate+" and "+edate+" group by P_SSN order by sum(E_amount) asc");
        while(rs9.next()) {
            System.out.println(rs9.getString(1)+"\t\t"+rs9.getString(2));
        }
        break;
//Retrieving the list of volunteers that are members of teams that support a particular client
        case 12: System.out.println("Displaying Client SSNs..");
        ResultSet rs10=st.executeQuery("select P_SSN from cares");
        while(rs10.next()) {
            System.out.println(rs10.getString(1));
        }
        System.out.println("Enter the Client SSN to retrieve the Volunteer details..");
        int rssn=Integer.parseInt(r.readLine());
        ResultSet rs11=st.executeQuery("select P_SSN,P_Name from Person where
P_SSN in (select P_SSN from serves where T_name in (select T_name from cares where
P_SSN='"+rssn+"'))");
        System.out.println("PSSN"\t+"Name");
        while(rs11.next()) {
            System.out.println(rs11.getString(1)+"\t"+rs11.getString(2));
        }
        break;
//Retrieving the names and contact information of the clients that are supported by
teams sponsored by an organization whose name starts with a letter between B and K. The client list
should be sorted by name
        case 13:
System.out.println("Name"\t+"Address"\t+"Email"\t+"Homephno"\t+"workphno"\t+
cellphno");

```

```

        ResultSet rs12=st.executeQuery("select P_Name,Address, Email, Home_phno,
Work_phno, Cell_phno from Person where P_SSN in(select P_SSN from cares where T_name in (select
T_name from Sponsors where O_name between 'B%' and 'T%')) order by P_Name asc");

        while(rs12.next()) {

System.out.println(rs12.getString(1)+"\t"+rs12.getString(2)+"\t"+rs12.getString(3)+"\t"+rs12.getString(
4)+"\t"+rs12.getString(5)+"\t"+rs12.getString(6));

    }

    break;

    case 14: System.out.println("Retrieving name and total amount donated by donors who
are also employees");

        ResultSet rs13=st.executeQuery("select p.P_Name,d.Anonymous,amount from
(select am.P_SSN ssn,max(am.Amount) amount from (select c.P_SSN,c.Amount from Cheque
c,Employee e where c.P_SSN=e.P_SSN union all select ca.P_SSN,ca.Amount from Card ca,Employee e
where ca.P_SSN=e.P_SSN)am group by am.P_SSN),donor d,Person p where d.P_SSN=ssn and
p.P_SSN=ssn");

        System.out.println("Name"\t+"Anonymous"\t+"Total Amount");

        while(rs13.next()) {

System.out.println(rs13.getString(1)+"\t"+rs13.getString(2)+"\t"+rs13.getString(3));

    }

    break;

    //Retrieving name and contact info of volunteers who worked max hours
between March and June

    case 15: System.out.println("Retrieving name and contact information of the
volunteers...");

        System.out.println("Name"\t+"Address"\t+"Email"\t+"Homephno"\t+"workphno"\t+
"cellphno");

        ResultSet rs14=st.executeQuery("select
P_Name,Address,Email,Home_phno,Work_phno,Cell_phno from person,serves where
person.P_SSN=serves.P_SSN and (serves.t_name,serves.hours) in (select T_name,max(hours) from
serves where Month in ('March','April','May','June') group by T_name);");

        while(rs14.next()) {

System.out.println(rs14.getString(1)+"\t"+rs14.getString(2)+"\t"+rs14.getString(3)+"\t"+rs14.getString(
4)+"\t"+rs14.getString(5)+"\t"+rs14.getString(6));

```

```

    }

break;

//Updating the employee salary to whom more than one team is reporting

case 16: st.executeQuery("update employee set salary=salary+(salary/10) where P_SSN
in (select P_SSN from report group by P_SSN having count(P_SSN)>1)");

System.out.println("Salary Updated...");

ResultSet rs15=st.executeQuery("Select * from employee");

System.out.println("EmployeeSSN"+'\t'+"Salary"+'\t'+"Marital_status"+'\t'+"Hire_date");

while(rs15.next()) {

System.out.println(rs15.getString(1)+"\t"+rs15.getString(2)+"\t"+rs15.getString(3)+"\t"+rs15.getString
(4));

}

break;

//Deleting client details who don't have health insurance and their need of
transportation has value less than 5

case 17: System.out.println("Before deletion...");

System.out.println("ClientSSN"+'\t'+"Dname"+'\t'+"Aname"+'\t'+"Dphno"+'\t'+"Aphno"+'\t'+"Date"
);

ResultSet rs16=st.executeQuery("Select * from client");

while(rs16.next()) {

System.out.println(rs16.getString(1)+"\t"+rs16.getString(2)+"\t"+rs16.getString(3)+"\t"+rs16.getString
(4)+"\t"+rs16.getString(5)+"\t"+rs16.getString(6));

}

st.executeQuery("delete from client where P_SSN not in (select P_SSN from
insurance_policy where P_type='health') and P_SSN in (select P_SSN from needs where
type='transportation' and value<5)");

System.out.println("Deleting Client data...");

System.out.println("After deletion...");

```

```

System.out.println("ClientSSN"+"\t"+"Dname"+"\t"+"Aname"+"\t"+"Dphno"+"\t"+"Aphno"+"\t"+"Date"
);

        ResultSet rs17=st.executeQuery("Select * from client");

        while(rs17.next()) {

System.out.println(rs17.getString(1)+"\t"+rs17.getString(2)+"\t"+rs17.getString(3)+"\t"+rs17.getString(
4)+"\t"+rs17.getString(5)+"\t"+rs17.getString(6));

        }

        break;

        //Importing data from a file into team table

        case 18: System.out.println("Enter import file name: ");

        String import_file_name=r.readLine();

        FileInputStream fstream = new FileInputStream("C:/Users/rvino/Desktop/"+import_file_name);
DataInputStream in = new DataInputStream(fstream);

        BufferedReader br18 = new BufferedReader(new InputStreamReader(in));

        String strLine;

        while ((strLine = br18.readLine()) != null)

        {

            String a[] = strLine.split(",");
String name = a[0];
String type = a[1];
String date=a[2];

            st.executeUpdate("insert into team values("+name+ "," +type+ ","+date+ ")");
System.out.println(" One row inserted succesfully");

        }

        System.out.println("File imported succesfully...");

        break;

        //Exporting team data to a file

        case 19: System.out.println("Exporting data to a file..");

        ResultSet rs18=st.executeQuery("select * from team");

        System.out.println("Enter output file name: ");

```

```

        String opt=r.readLine();

BufferedWriter output = new BufferedWriter(new FileWriter("C:/Users/rvino/Desktop/"+opt));

        while(rs18.next())
        {

            String str= rs18.getString("T_name");
            String st1= rs18.getString("T_Type");
            String st2 =rs18.getString("T_date");
            output.write("\n");
            output.write(str+"\t");
            output.write(st1+"\t");
            output.write(st2+"\t");
            output.write("\n");

            System.out.println(" One Record inserted in a file succesfully!");

        }

        System.out.println(" File exported succesfully...");
        output.close();

        break;

case 20: System.out.println("Exiting....");
        System.exit(0);
        break;
    }

}while(n<20);

dbc.close();

}

catch(SQLException e) {
    e.printStackTrace();
}

}
}

```

TASK 6 Execution of Java Program

TASK 6.1 Query 1 Output

```
Connected
WELCOME TO THE PAN CLIENT AND DONOR DATABASE SYSTEM!!!
1.Enter a new team into the database
2.Enter a new client into the database and associate him or her with one or more
teams
3.Enter a new volunteer into the database and associate him or her with one or more
teams
4.Enter the number of hours a volunteer worked this month for a particular team
5.Enter a new employee into the database and associate him or her with one or more
teams
6.Enter an expense charged by an employee
7.Enter a new organization and associate it to one or more PAN teams
8.Enter a new donor and associate him or her with several donations
9.Enter a new organization and associate it with several donations
10.Retrieve the name and phone number of the doctor of a particular client
11.Retrieve the total amount of expenses charged by each employee for a particular
period of time. The list should be sorted by the total amount of expenses
12.Retrieve the list of volunteers that are members of teams that support a
particular client
13.Retrieve the names and contact information of the clients that are supported by
teams sponsored by an organization whose name starts with a letter between B and K.
The client list should be sorted by name
14.Retrieve the name and total amount donated by donors that are also employees. The
list should be sorted by the total amount of the donations, and indicate if each
donor wishes to remain anonymous
15.For each team, retrieve the name and associated contact information of the
volunteer that has worked the most total hours between March and June
16.Increase the salary by 10% of all employees to whom more than one team must report
17.Delete all clients who do not have health insurance and whose value of importance
for transportation is less than 5
18.Import option-Entering Team values from a file
19.Export option-Retrieve all data from Team and write to a file
20.Exit from program
Enter your option....
1
Enter the Team name
Green
Enter the Team type
Admin
Enter the Team formed date
12-jul-1996
Enter your option....
1
Enter the Team name
Blue
Enter the Team type
Support
Enter the Team formed date
```

17-aug-2000

Enter your option....

1

Enter the Team name

Pink

Enter the Team type

Finance

Enter the Team formed date

20-sep-2002

Enter your option....

1

Enter the Team name

Yellow

Enter the Team type

Research

Enter the Team formed date

7-mar-1980

Enter your option....

1

Enter the Team name

Red

Enter the Team type

Testing

Enter the Team formed date

9-apr-1960

TEAM TABLE:

T_NAME	T_TYPE	T_DATE
1 Pink	Finance	20-SEP-02
2 Yellow	Research	07-MAR-80
3 Red	Testing	09-APR-60
4 Green	Admin	12-JUL-96
5 Blue	Support	17-AUG-00

eclipse-workspace - testJDBC/src/testJDBC/p.java - Eclipse

```
File Edit Source Refactor Navigate Search Project Run Window Help
```

Conn.java p.java

```
78     + "7.Enter a new organization and associate it to one or more PAN teams\n"
79     + "8.Enter a new donor and associate him or her with several donations\n"
80     + "9.Enter a new organization and associate it with several donations\n"
81     + "10.Retrieve the names and contact information of the clients that are supported by an organization whose name starts with a letter between B and K.\n"
82     + "11.Retrieve the total amount of expenses charged by each employee for a particular period of time. The list should be sorted by the total amount of the donation.\n"
83     + "12.Retrieve the list of volunteers that are members of teams that support a particular client\n"
84     + "13.Retrieve the names and contact information of the clients that are supported by an organization whose name starts with a letter between B and K.\n"
85     + "14.Retrieve the names and contact information of the clients that are supported by an organization whose name starts with a letter between B and K. The list should be sorted by the total amount of the donation.\n"
86     + "15.For each team, retrieve the name and associated contact information of the volunteer that has worked the most total hours between March and June.\n"
87     + "16.Increase the salary by 10% of all employees to whom more than one team must report\n"
88     + "17.Delete all clients who do not have health insurance and whose value of importance for transportation is less than 5\n"
89     + "18.Import option-Entering Team values from a file\n"
90     + "19.Export option-Retrieve all data from Team and write to a file\n"
91
92     n=Integer.parseInt(readline());
93     switch(n) {
94         case 1: System.out.println("Enter the Team name");
95         String s1=(r.readLine());
96         System.out.println("Enter the Team type");
97         String s2=(r.readLine());
```

Console

```
13. Retrieve the names and contact information of the clients that are supported by teams sponsored by an organization whose name starts with a letter between B and K. The list should be sorted by the total amount of the donation.\n14. Retrieve the name and total amount donated by donors that are also employees. The list should be sorted by the total amount of the donations, and indicate if each donor is an employee or not.\n15. For each team, retrieve the name and associated contact information of the volunteer that has worked the most total hours between March and June.\n16. Increase the salary by 10% of all employees to whom more than one team must report.\n17. Delete all clients who do not have health insurance and whose value of importance for transportation is less than 5.\n18. Import option-Entering Team values from a file.\n19. Export option-Retrieve all data from Team and write to a file.\n20. Exit from program.\n1\nEnter the Team name\nGreen
```

Type here to search

TASK 6.2 Query 2 Output

Enter your option....

2

Enter the Person details first

Enter the Person SSN

1876

Enter the Person name

Tharun

Enter the Person DOB

7-dec-1994

Enter the Person Race

Asian

Enter the Person Gender

Male

Enter the Person Profession

Student

Enter the Person Contact details..

Enter the Person mailing address

1121-H

Enter the Person Email

tharun@gmail.com

Enter the Person Home_ph_no

5673214

Enter the Person Work_ph_no

4321297

Enter the Person Cell_ph_no

5649325

Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list

1

Now Enter the Client details....

Enter the Doctor name

John

Enter the Attorney name

Jack

Enter the Doctor phone number

8764322

Enter the Attorney phone number

4563297

Enter the date client was assigned to organization

10-mar-2000

Now Associating clients to teams

Retrieving team names...

Tname

Blue

Green

Pink

Red

Yellow

Enter the number of teams to be linked to

1

Enter the team name to be associated with
Green
Enter your option....
2
Enter the Person details first

Enter the Person SSN
1996
Enter the Person name
James
Enter the Person DOB
5-jan-1996
Enter the Person Race
American
Enter the Person Gender
Male
Enter the Person Profession
Engineer
Enter the Person Contact details..

Enter the Person mailing address
1189-B
Enter the Person Email
james@gmail.com
Enter the Person Home_ph_no
3452987
Enter the Person Work_ph_no
9658437
Enter the Person Cell_ph_no
5439977
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
0

Now Enter the Client details....
Enter the Doctor name
Williams
Enter the Attorney name
Amen
Enter the Doctor phone number
7820604
Enter the Attorney phone number
5617820
Enter the date client was assigned to organization
11-apr-2007
Now Associating clients to teams
Retrieving team names...
Tname

Blue
Green
Pink
Red
Yellow
Enter the number of teams to be linked to

2
Enter the team name to be associated with
Blue
Enter the team name to be associated with
Yellow
Enter your option....
2
Enter the Person details first

Enter the Person SSN
4321
Enter the Person name
Adams
Enter the Person DOB
12-sep-1960
Enter the Person Race
American
Enter the Person Gender
Male
Enter the Person Profession
Professor
Enter the Person Contact details..

Enter the Person mailing address
1017-H
Enter the Person Email
adams@gmail.com
Enter the Person Home_ph_no
4568876
Enter the Person Work_ph_no
5619089
Enter the Person Cell_ph_no
5619041
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
1
Now Enter the Client details....
Enter the Doctor name
John
Enter the Attorney name
Jordan
Enter the Doctor phone number
8017820
Enter the Attorney phone number
5619056
Enter the date client was assigned to organization
20-aug-2003
Now Associating clients to teams
Retrieving team names...
Tname

Blue
Green
Pink
Red

Yellow
Enter the number of teams to be linked to
1
Enter the team name to be associated with
Yellow

Enter your option....
2
Enter the Person details first

Enter the Person SSN
6832
Enter the Person name
Sharon
Enter the Person DOB
28-feb-1995
Enter the Person Race
Asian
Enter the Person Gender
Female
Enter the Person Profession
Professor
Enter the Person Contact details..

Enter the Person mailing address
1161-B
Enter the Person Email
sharon@gmail.com
Enter the Person Home_ph_no
4097820
Enter the Person Work_ph_no
4056155
Enter the Person Cell_ph_no
7890543
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
1
Now Enter the Client details....
Enter the Doctor name
Williams
Enter the Attorney name
Amen
Enter the Doctor phone number
7820604
Enter the Attorney phone number
5617820
Enter the date client was assigned to organization
11-apr-2007
Now Associating clients to teams
Retrieving team names...
Tname

Blue
Green
Pink

Red
Yellow
Enter the number of teams to be linked to
1
Enter the team name to be associated with
Pink
Enter your option....
2
Enter the Person details first

Enter the Person SSN
7195
Enter the Person name
Joe
Enter the Person DOB
17-aug-1995
Enter the Person Race
Asian
Enter the Person Gender
Female
Enter the Person Profession
Director
Enter the Person Contact details..

Enter the Person mailing address
1009-C
Enter the Person Email
joe@gmail.com
Enter the Person Home_ph_no
5617889
Enter the Person Work_ph_no
7945561
Enter the Person Cell_ph_no
9066432
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
1
Now Enter the Client details....
Enter the Doctor name
John
Enter the Attorney name
Jack
Enter the Doctor phone number
8764322
Enter the Attorney phone number
4563297
Enter the date client was assigned to organization
10-mar-2000
Now Associating clients to teams
Retrieving team names...
Tname

Blue
Green
Pink

Red

Yellow

Enter the number of teams to be linked to

1

Enter the team name to be associated with

Green

PERSON TABLE

	P_SSN	P_NAME	DOB	RACE	GENDER	PROFESSION	ADDRESS	EMAIL	HOME_PHNO	WORK_PHNO	CELL_PHNO	MAILING_LIST
1	7195	Joe	17-AUG-95	Asian	Female	Director	1009-C	joe@gmail.com	5617889	7945561	9066432	1
2	1996	James	05-JAN-96	American	Male	Engineer	1189-B	james@gmail.com	3452987	9655437	5439977	0
3	6832	Sharon	28-FEB-95	Asian	Female	Professor	1161-B	sharon@gmail.com	4097820	4056155	7890543	1
4	1876	Tharun	07-DEC-94	Asian	Male	Student	1121-H	tharun@gmail.com	5673214	4321297	5649325	1
5	4321	Adams	12-SEP-60	American	Male	Professor	1017-H	adams@gmail.com	4568876	5619089	5619041	1

CLIENT TABLE

	P_SSN	DNAME	ANAME	DPHNO	APHNO	CDATE
1	4321	John	Jordan	8017820	5619056	20-AUG-03
2	6832	Williams	Amen	7820604	5617820	11-APR-07
3	1876	John	Jack	8764322	4563297	10-MAR-00
4	1996	Williams	Amen	7820604	5617820	11-APR-07
5	7195	John	Jack	8764322	4563297	10-MAR-00

CARES TABLE

	P_SSN	T_NAME	STATUS
1	1996	Yellow	0
2	1876	Green	0
3	1996	Blue	0
4	4321	Yellow	0
5	6832	Pink	0
6	7195	Green	0

```
eclipse-workspace - testjdbc/src/testjdbc/lp.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Connjava lp.java
78 + "7.Enter a new organization and associate it to one or more PAN teams\n"
79 + "8.Enter a new donor and associate him or her with several donations\n"
80 + "9.Enter a new organization and associate it with several donations\n"
81 + "10.Retrieve the name and phone number of the doctor of a particular client\n"
82 + "11.Retrieve the total amount of expenses charged by each employee for a particular period of time. The list should be sorted by the total amount
83 + "12.Retrieve the list of volunteers that are members of teams that support a particular client\n"
84 + "13.Retrieve the names and contact information of the clients that are supported by teams sponsored by an organization whose name starts with a
85 + "14.Retrieve the name and total amount donated by donors that are also employees. The list should be sorted by the total amount of the donation
86 + "15.For each team, retrieve the name and associated contact information of the volunteer that has worked the most total hours between March and
87 + "16.Increase the salary by 10% of all employees to whom more than one team must report\n"
88 + "17.Delete all clients who do not have health insurance and whose value of importance for transportation is less than 5\n"
89 + "18.Import option-Entering Team values from a file\n"
90 + "19.Export option-Retrieve all data from Team and write to a file\n"
91 + "20.Exit from program";
92 n=Integer.parseInt(r.readLine());
93 switch(n) {
94 case 1: System.out.println("Enter the Team name");
95 String s1=(r.readLine());
96 System.out.println("Enter the Team type");
97 String s2=(r.readLine());
```

Console

```
lp [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 10:56:17 AM)
18.Import option-Entering Team values from a file
19.Export option-Retrieve all data from Team and write to a file
20.Exit from program
2
Enter the Person details first
Enter the Person SSN
6832
Enter the Person name
Sharon
Enter the Person DOB
28-feb-1995|
```

Type here to search

TASK 6.3 Query 3 Output

Enter your option....

3

Enter the Person details first

Enter the Person SSN

5823

Enter the Person name

Ria

Enter the Person DOB

12-jan-1995

Enter the Person Race

Asian

Enter the Person Gender

Female

Enter the Person Profession

Student

Enter the Person Contact details..

Enter the Person mailing address

1106-A

Enter the Person Email

ria@gmail.com

Enter the Person Home_ph_no

8019076

Enter the Person Work_ph_no

5614563

Enter the Person Cell_ph_no

7832561

Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list

1

Enter the Volunteer details..

Enter the Volunteer joining date

10-mar-2015

Enter the Volunteer training date

16-apr-2016

Enter the Volunteer training location

Norman

Tname

Blue

Green

Pink

Red

Yellow

Enter the number of teams to be linked to

1

Enter the team name to be associated with

Green

Enter the no of hours he worked

6

Enter the value for month
July

Enter your option....
3

Enter the Person details first

Enter the Person SSN
6178

Enter the Person name
Tina

Enter the Person DOB
14-mar-1996

Enter the Person Race
Asian

Enter the Person Gender
Female

Enter the Person Profession
Student

Enter the Person Contact details..

Enter the Person mailing address
1071-C

Enter the Person Email
tina@gmail.com

Enter the Person Home_ph_no
5637832

Enter the Person Work_ph_no
6784321

Enter the Person Cell_ph_no
5014563

Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
0

Enter the Volunteer details..

Enter the Volunteer joining date
11-apr-2016

Enter the Volunteer training date
20-aug-2017

Enter the Volunteer training location
Olc

Tname

Blue
Green
Pink
Red
Yellow

Enter the number of teams to be linked to
2

Enter the team name to be associated with
Blue

Enter the no of hours he worked
5

Enter the value for month

May

Enter the team name to be associated with

Red

Enter the no of hours he worked

8

Enter the value for month

April

Enter your option....

3

Enter the Person details first

Enter the Person SSN

1296

Enter the Person name

Priya

Enter the Person DOB

28-aug-1993

Enter the Person Race

Asian

Enter the Person Gender

Female

Enter the Person Profession

Professor

Enter the Person Contact details..

Enter the Person mailing address

1110-C

Enter the Person Email

priya@gmail.com

Enter the Person Home_ph_no

5609076

Enter the Person Work_ph_no

8015634

Enter the Person Cell_ph_no

5619077

Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list

1

Enter the Volunteer details..

Enter the Volunteer joining date

7-dec-2014

Enter the Volunteer training date

4-sep-2015

Enter the Volunteer training location

Dallas

Tname

Blue

Green

Pink

Red

Yellow
Enter the number of teams to be linked to
1
Enter the team name to be associated with
Red
Enter the no of hours he worked
4
Enter the value for month
October
Enter your option....
3
Enter the Person details first

Enter the Person SSN
2807
Enter the Person name
Simon
Enter the Person DOB
5-apr-1994
Enter the Person Race
American
Enter the Person Gender
Male
Enter the Person Profession
Professor
Enter the Person Contact details..

Enter the Person mailing address
1104-H
Enter the Person Email
simon@gmail.com
Enter the Person Home_ph_no
7820561
Enter the Person Work_ph_no
5690891
Enter the Person Cell_ph_no
7801820
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
0

Enter the Volunteer details..
Enter the Volunteer joining date
8-jul-2014
Enter the Volunteer training date
17-sep-2015
Enter the Volunteer training location
Kansas
Tname

Blue
Green
Pink
Red
Yellow

Enter the number of teams to be linked to
1
Enter the team name to be associated with
Yellow
Enter the no of hours he worked
6
Enter the value for month
March
Enter your option....
3
Enter the Person details first

Enter the Person SSN
3459
Enter the Person name
Karma
Enter the Person DOB
16-sep-1996
Enter the Person Race
American
Enter the Person Gender
Male
Enter the Person Profession
Student
Enter the Person Contact details..

Enter the Person mailing address
1017-B
Enter the Person Email
karma@gmail.com
Enter the Person Home_ph_no
5557893
Enter the Person Work_ph_no
6734597
Enter the Person Cell_ph_no
4098754
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
1

Enter the Volunteer details..
Enter the Volunteer joining date
20-sep-2013
Enter the Volunteer training date
9-may-2015
Enter the Volunteer training location
Colorado
Tname

Blue
Green
Pink
Red
Yellow
Enter the number of teams to be linked to

1

Enter the team name to be associated with
Pink

Enter the no of hours he worked

8

Enter the value for month

June

PERSON TABLE

	P_SSN	P_NAME	DOB	RACE	GENDER	PROFESSION	ADDRESS	EMAIL	HOME_PNO	WORK_PNO	CELL_PNO	MAILING_LIST
1	7195	Joe	17-AUG-95	Asian	Female	Director	1009-C	joe@gmail.com	5617889	7945561	9066432	1
2	1996	James	05-JAN-96	American	Male	Engineer	1189-B	james@gmail.com	3452987	9658437	5439977	0
3	6832	Sharon	28-FEB-95	Asian	Female	Professor	1161-B	sharon@gmail.com	4097820	4056155	7890543	1
4	1876	Tharun	07-DEC-94	Asian	Male	Student	1121-H	tharun@gmail.com	5673214	4321297	5649325	1
5	4321	Adams	12-SEP-60	American	Male	Professor	1017-H	adams@gmail.com	4568876	5619089	5619041	1
6	3459	Karma	16-SEP-96	American	Male	Student	1017-B	karma@gmail.com	5557893	6734597	4098754	1
7	5823	Ria	12-JAN-95	Asian	Female	Student	1106-A	ria@gmail.com	8019076	5614563	7832561	1
8	6178	Tina	14-MAR-96	Asian	Female	Student	1071-C	tina@gmail.com	5637832	6784321	5014563	0
9	1296	Priya	28-AUG-93	Asian	Female	Professor	1110-C	priya@gmail.com	5609076	8015634	5619077	1
10	2807	Simon	05-APR-94	American	Male	Professor	1104-H	simon@gmail.com	7820561	5690891	7801820	0

VOLUNTEER TABLE

	P_SSN	V_DATE	T_DATE	T_LOC
1	3459	20-SEP-13	09-MAY-15	Colorado
2	1296	07-DEC-14	04-SEP-15	Dallas
3	2807	08-JUL-14	17-SEP-15	Kansas
4	5823	10-MAR-15	16-APR-16	Norman
5	6178	11-APR-16	20-AUG-17	Okc

SERVES TABLE

	P_SSN	T_NAME	HOURS	ACTIVE	MONTH
1	3459	Pink	8	1	June
2	1296	Red	4	1	October
3	2807	Yellow	6	1	March
4	5823	Green	7	1	July
5	6178	Blue	5	1	May
6	6178	Red	8	1	April

eclipse-workspace - testJdbc/src/testJdbc/lp.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Conn.java lp.java
144     System.out.println("Enter the team name to be associated with");
145     while(rs1.next()) {
146         System.out.println(rs1.getString(1));
147     }
148     System.out.println("Enter the number of teams to be linked to");
149     int vo=Integer.parseInt(r.readLine());
150     while(vo!=0) {
151         System.out.println("Enter the team name to be associated with");
152         String tm=(r.readLine());
153         System.out.println("Enter the no of hours he worked");
154         int tt=Integer.parseInt(r.readLine());
155         System.out.println("Enter the value for month");
156         String tm=(r.readLine());
157         st.executeUpdate("insert into Serves values('"+vp+"','"+tt+"','"+tm+"',1,'"+tm+"')");
158         vo--;
159     }
160     case 4: System.out.println("Displaying Volunteer details...");
161     ResultSet rs2=st.executeQuery("select P_SSN,P_Name from Person where P_SSN in (select P_SSN from Volunteer) ");
162     System.out.println("P_SSN\tName\n");
163     while(rs2.next()) {
164         System.out.println(rs2.getString(1)+"\t"+rs2.getString(2));
165     }
    
```

Console

```

lp [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 11:39:55 AM)
18. Import option-Entering Team values from a file
19. Export option-Retrieve all data from Team and write to a file
20. Exit from program
3
Enter the Person details first

Enter the Person SSN
3459
Enter the Person name
Karma
Enter the Person DOB
16-sep-1996
<
```

Windows Taskbar:

- Type here to search
- File Explorer
- Google Chrome
- Mail
- Edge
- OneDrive
- PowerPoint
- Word
- Excel
- Snipping Tool
- Calculator
- File History
- Task View
- System
- 11:57 AM 11/19/2017

TASK 6.4 Query 4 Output

Enter your option....

4

Displaying Volunteer details...

P_SSN Name

3459 Karma

5823 Ria

6178 Tina

1296 Priya

2807 Simon

Displaying team names...

Tname

Blue

Green

Pink

Red

Yellow

Enter the SSN

3459

Enter the team name to be associated with

Green

Enter the no of hours he worked

6

Enter the value for month

March

Enter your option....

4

Displaying Volunteer details...

P_SSN Name

3459 Karma

5823 Ria

6178 Tina

1296 Priya

2807 Simon

Displaying team names...

Tname

Blue

Green

Pink

Red

Yellow

Enter the SSN

6178

Enter the team name to be associated with

Blue

Enter the no of hours he worked

4

Enter the value for month

June

```
Enter your option....  
4  
Displaying Volunteer details...  
P_SSN      Name  
  
3459  Karma  
5823  Ria  
6178  Tina  
1296  Priya  
2807  Simon  
Displaying team names...  
Tname  
  
Blue  
Green  
Pink  
Red  
Yellow  
Enter the SSN  
1296  
Enter the team name to be associated with  
Yellow  
Enter the no of hours he worked  
3  
Enter the value for month  
January  
Enter your option....  
4  
Displaying Volunteer details...  
P_SSN      Name  
  
3459  Karma  
5823  Ria  
6178  Tina  
1296  Priya  
2807  Simon  
Displaying team names...  
Tname  
  
Blue  
Green  
Pink  
Red  
Yellow  
Enter the SSN  
2807  
Enter the team name to be associated with  
Pink  
Enter the no of hours he worked  
2  
Enter the value for month  
February  
Enter your option....  
4  
Displaying Volunteer details...
```

P_SSN	Name
3459	Karma
5823	Ria
6178	Tina
1296	Priya
2807	Simon
Displaying team names...	
Tname	
Blue	
Green	
Pink	
Red	
Yellow	
Enter the SSN	
5823	
Enter the team name to be associated with	
Pink	
Enter the no of hours he worked	
8	
Enter the value for month	
November	

SERVES TABLE

P_SSN	T_NAME	HOURS	ACTIVE	MONTH
1	3459 Pink	8	1	June
2	1296 Red	4	1	October
3	2807 Yellow	6	1	March
4	1296 Yellow	3	1	January
5	5823 Pink	8	1	November
6	3459 Green	6	1	March
7	5823 Green	7	1	July
8	6178 Blue	5	1	May
9	6178 Red	8	1	April
10	2807 Pink	2	1	February
11	6178 Blue	4	1	June

eclipse-workspace - testJdbc/src/testJdbc/lp.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Conn.java lp.java
162     while(rs2.next()) {
163         System.out.println(rs2.getString(1)+"\t"+rs2.getString(2));
164     }
165     System.out.println("Displaying team names... ");
166     ResultSet rs3=st.executeQuery("select t_name from Team");
167     System.out.println("Tname\n");
168     while(rs3.next()) {
169         System.out.println(rs3.getString(1));
170     }
171     System.out.println("Enter the SSN");
172     int ps=Integer.parseInt(r.readLine());
173     System.out.println("Enter the team name to be associated with");
174     String an=r.readLine();
175     System.out.println("Enter the no of hours he worked");
176     int ho=Integer.parseInt(r.readLine());
177     System.out.println("Enter the value for month");
178     String m=r.readLine();
179     st.executeUpdate("insert into Serves values('"+ps+"','"+an+"','"+ho+"',1,'"+m+"')");
180     break;
181 case 5: int es=input();

```

Console

```

<terminated> lp [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 12:14:06 PM)
4
Displaying Volunteer details...
P_SSN Name
3459 Karma
5823 Ria
6178 Tina
1296 Priya
2807 Simon
Displaying team names...
Tname

```

12:22 PM 11/19/2017

TASK 6.5 Query 5 Output

Enter your option....

5

Enter the Person details first

Enter the Person SSN

1492

Enter the Person name

Evelin

Enter the Person DOB

28-dec-1995

Enter the Person Race

Asian

Enter the Person Gender

Female

Enter the Person Profession

Nurse

Enter the Person Contact details..

Enter the Person mailing address

1002-H

Enter the Person Email

eve@gmail.com

Enter the Person Home_ph_no

4670986

Enter the Person Work_ph_no

8905432

Enter the Person Cell_ph_no

5439655

Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list

1

Enter the Employee details

Enter the salary of Employee

4500

Enter the Marital_status of Employee

Single

Enter the hire_date of the Employee

20-jan-2017

Tname

Blue

Green

Pink

Red

Yellow

Enter the number of teams to be linked to

2

Enter the team name to be associated with

Green

Enter the report description

PatientA

Enter the reporting date

26-oct-2017

Enter the team name to be associated with
Red
Enter the report description
PatientB
Enter the reporting date
10-nov-2017
Enter your option....
5
Enter the Person details first

Enter the Person SSN
2409
Enter the Person name
Janani
Enter the Person DOB
19-mar-1996
Enter the Person Race
Asian
Enter the Person Gender
Female
Enter the Person Profession
Assistant
Enter the Person Contact details..

Enter the Person mailing address
1165-A
Enter the Person Email
janani@gmail.com
Enter the Person Home_ph_no
4356733
Enter the Person Work_ph_no
7654998
Enter the Person Cell_ph_no
4556784
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
0
Enter the Employee details
Enter the salary of Employee
2205.73
Enter the Marital_status of Employee
Single
Enter the hire_date of the Employee
20-apr-2016
Tname

Blue
Green
Pink
Red
Yellow
Enter the number of teams to be linked to
1
Enter the team name to be associated with
Blue

Enter the report description
Client_data
Enter the reporting date
7-jun-2017
Enter your option....
5
Enter the Person details first

Enter the Person SSN
4054
Enter the Person name
Shalini
Enter the Person DOB
31-mar-1996
Enter the Person Race
Asian
Enter the Person Gender
Female
Enter the Person Profession
Counselor
Enter the Person Contact details..

Enter the Person mailing address
1115-B
Enter the Person Email
shal@gmail.com
Enter the Person Home_ph_no
4567880
Enter the Person Work_ph_no
5619090
Enter the Person Cell_ph_no
8017898
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
1
Enter the Employee details
Enter the salary of Employee
3814.67
Enter the Marital_status of Employee
Single
Enter the hire_date of the Employee
7-aug-2016
Tname

Blue
Green
Pink
Red
Yellow
Enter the number of teams to be linked to
1
Enter the team name to be associated with
Green
Enter the report description
Patient_Counseling

Enter the reporting date
12-jul-2017
Enter your option....
5
Enter the Person details first

Enter the Person SSN
5897
Enter the Person name
Lavanya
Enter the Person DOB
30-dec-1995
Enter the Person Race
American
Enter the Person Gender
Female
Enter the Person Profession
PA
Enter the Person Contact details..

Enter the Person mailing address
1110-H
Enter the Person Email
lav@gmail.com
Enter the Person Home_ph_no
7884950
Enter the Person Work_ph_no
4056778
Enter the Person Cell_ph_no
8015619
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
1
Enter the Employee details
Enter the salary of Employee
4900
Enter the Marital_status of Employee
Single
Enter the hire_date of the Employee
20-oct-2015
Tname

Blue
Green
Pink
Red
Yellow
Enter the number of teams to be linked to
1
Enter the team name to be associated with
Pink
Enter the report description
General
Enter the reporting date
22-nov-2016

Enter your option....
5
Enter the Person details first

Enter the Person SSN
6413
Enter the Person name
Saron
Enter the Person DOB
28-aug-1995
Enter the Person Race
Asian
Enter the Person Gender
Female
Enter the Person Profession
Assistant
Enter the Person Contact details..
Enter the Person mailing address
1173-B
Enter the Person Email
saron@gmail.com
Enter the Person Home_ph_no
4566321
Enter the Person Work_ph_no
4098799
Enter the Person Cell_ph_no
4097765
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
0
Enter the Employee details
Enter the salary of Employee
3500
Enter the Marital_status of Employee
Married
Enter the hire_date of the Employee
18-mar-2015
Tname

Blue
Green
Pink
Red
Yellow
Enter the number of teams to be linked to
1
Enter the team name to be associated with
Yellow
Enter the report description
Employee_Data
Enter the reporting date
17-jul-2017

PERSON TABLE

	P_SSN	P_NAME	DOB	RACE	GENDER	PROFESSION	ADDRESS	EMAIL	HOME_PHNO	WORK_PHNO	CELL_PHNO	MAILING_LIST
1	7195	Joe	17-AUG-95	Asian	Female	Director	1009-C	joe@gmail.com	5617889	7945561	9066432	1
2	1996	James	05-JAN-96	American	Male	Engineer	1189-B	james@gmail.com	3452987	9658437	5439977	0
3	6832	Sharon	28-FEB-95	Asian	Female	Professor	1161-B	sharon@gmail.com	4097820	4056155	7890543	1
4	4054	Shalini	31-MAR-96	Asian	Female	Counselor	1115-B	shal@gmail.com	4567880	5619090	8017898	1
5	5897	Lavanya	30-DEC-95	American	Female	PA	1110-H	lav@gmail.com	7884950	4056778	8015619	1
6	1876	Tharun	07-DEC-94	Asian	Male	Student	1121-H	tharun@gmail.com	5673214	4321297	5649325	1
7	4321	Adams	12-SEP-60	American	Male	Professor	1017-H	adams@gmail.com	4568876	5619089	5619041	1
8	3459	Karma	16-SEP-96	American	Male	Student	1017-B	karma@gmail.com	5557893	6734597	4098754	1
9	1492	Evelin	28-DEC-95	Asian	Female	Nurse	1002-H	eve@gmail.com	4670986	8905432	5439655	1
10	2409	Janani	19-MAR-96	Asian	Female	Assistant	1165-A	janani@gmail.com	4356733	7654998	4556784	0
11	5823	Ria	12-JAN-95	Asian	Female	Student	1106-A	ria@gmail.com	8019076	5614563	7832561	1
12	6178	Tina	14-MAR-96	Asian	Female	Student	1071-C	tina@gmail.com	5637832	6784321	5014563	0
13	1296	Priya	28-AUG-93	Asian	Female	Professor	1110-C	priya@gmail.com	5609076	8015634	5619077	1
14	2807	Simon	05-APR-94	American	Male	Professor	1104-H	simon@gmail.com	7820561	5690891	7801820	0
15	6413	Saron	28-AUG-95	Asian	Female	Assistant	1173-B	saron@gmail.com	4566321	4098799	4097765	0

EMPLOYEE TABLE

	P_SSN	SALARY	M_STATUS	HIRE_DATE
1	4054	3814.67	Single	07-AUG-16
2	1492	4500	Single	20-JAN-17
3	2409	2205.73	Single	20-APR-16
4	5897	4900	Single	20-OCT-15
5	6413	3500	Married	18-MAR-15

REPORT TABLE

	T_NAME	P_SSN	R_DESC	R_DATE
1	Blue	2409	Client_data	07-JUN-17
2	Green	1492	PatientA	26-OCT-17
3	Red	1492	PatientB	10-NOV-17
4	Green	4054	Patient_Counseling	12-JUL-17
5	Pink	5897	General	22-NOV-16
6	Yellow	6413	Employee_Data	17-JUL-17

eclipse-workspace - testjdbc/src/testjdbc/lp.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Conn.java lp.java Quick Access
1 Conn.java ① lp.java
195 System.out.println("Enter the number of teams to be linked to");
196 int no=Integer.parseInt(r.readLine());
197 while(no!=0) {
198     System.out.println("Enter the team name to be associated with");
199     String ro=r.readLine();
200     System.out.println("Enter the report description");
201     String r1=r.readLine();
202     System.out.println("Enter the reporting date");
203     String r2=r.readLine();
204     st.executeUpdate("insert into Report values('"+ro+"','"+es+"','"+r1+"','"+r2+"')");
205     no--;
206 }
207 break;
case 6: ResultSet rs5=st.executeQuery("select P_SSN from Employee");
209 System.out.println("Employees SSN\n");
210 while(rs5.next()) {
211     System.out.println(rs5.getString(1));
212 }
213 System.out.println("Enter the employee SSN for whom expense to be recorded");
214 String sn=(r.readLine());

```

Console

```

<terminated> lp [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 12:23:51 PM)
4670986
Enter the Person Work_ph_no
8905432
Enter the Person Cell_ph_no
5439655
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
1
Enter the Employee details
Enter the salary of Employee
4500
Enter the Marital_status of Employee
Single

```

TASK 6.6 Query 6 Output

Enter your option....

6

Employess SSN

4054

1492

2409

5897

6413

Enter the employee SSN for whom expense to be recorded

1492

Enter the expense date

10-mar-2017

Enter the expense amount

500

Enter the expense description

Rent

Enter your option....

6

Employess SSN

4054

1492

2409

5897

6413

Enter the employee SSN for whom expense to be recorded

2409

Enter the expense date

7-apr-2017

Enter the expense amount

79.22

Enter the expense description

OGE

Enter your option....

6

Employess SSN

4054

1492

2409

5897

6413

Enter the employee SSN for whom expense to be recorded

4054

Enter the expense date

22-jun-2017

Enter the expense amount

1000

Enter the expense description

Savings

Enter your option....

6

Employee's SSN

4054
1492
2409
5897
6413

Enter the employee SSN for whom expense to be recorded

6413

Enter the expense date

16-oct-2017

Enter the expense amount

115.70

Enter the expense description

Wifi

Enter your option....

6

Employee's SSN

4054
1492
2409
5897
6413

Enter the employee SSN for whom expense to be recorded

4054

Enter the expense date

25-jul-2017

Enter the expense amount

478.07

Enter the expense description

Food

EXPENSE TABLE

	P_SSN	E_DATE	E_AMOUNT	E_DESC
1	1492	10-MAR-17	500	Rent
2	2409	07-APR-17	79.22	OGE
3	4054	22-JUN-17	1000	Savings
4	6413	16-OCT-17	115.7	Wifi
5	4054	25-JUL-17	478.07	Food

The screenshot shows the Eclipse IDE interface with a Java file named 'Ip.java' open. The code implements a menu-driven application for managing expenses. It includes a main loop with numbered options for various operations like retrieving data, increasing salary, deleting clients, and exporting to a file. The user interacts with the program by entering their choice and providing specific details like dates and amounts. The terminal window at the bottom shows the command-line interface where the user runs the application, selects options, and views the resulting expense table.

```
public class Ip {
    public static void main(String[] args) {
        System.out.println("Enter your option....");
        int n = Integer.parseInt(r.readLine());
        switch(n) {
            case 1:
                System.out.println("Enter the Team name");
                String s1=r.readLine();
                System.out.println("Enter the Team type");
                String s2=r.readLine();
                System.out.println("Enter the Team Formed date");
                break;
            default:
                System.out.println("Invalid Option");
        }
    }
}
```

Console:

```
Ip [Java Application] C:\Program Files\Java\jre8.0_144\bin\javaw.exe (Nov 19, 2017, 1:13:14 PM)
Enter your option...
6
Employee's SSN
4054
1492
2409
5897
6413
Enter the employee SSN for whom expense to be recorded
6413
Enter the expense date
16-oct-2017
Enter the expense amount
115.70
Enter the expense description
Wifi
Enter your option....
```

TASK 6.7 Query 7 Output

```
Enter your option....  
7  
Enter the organization name  
Krish  
Enter the organization address  
1009-F  
Enter the organization phone no  
5619101  
Enter the organization contact person  
Kim  
Tname  
  
Blue  
Green  
Pink  
Red  
Yellow  
Enter the number of teams to be linked to  
2  
Enter the team name to be associated with  
Green  
Enter the team name to be associated with  
Red  
Enter your option....  
7  
Enter the organization name  
Jerry  
Enter the organization address  
1156-B  
Enter the organization phone no  
5619088  
Enter the organization contact person  
Praveen  
Tname  
  
Blue  
Green  
Pink  
Red  
Yellow  
Enter the number of teams to be linked to  
1  
Enter the team name to be associated with  
Blue  
Enter your option....  
7  
Enter the organization name  
Tom  
Enter the organization address  
1008-H  
Enter the organization phone no  
4568333  
Enter the organization contact person
```

Solomon

Tname

Blue

Green

Pink

Red

Yellow

Enter the number of teams to be linked to

1

Enter the team name to be associated with

Yellow

Enter your option....

7

Enter the organization name

Mike

Enter the organization address

1122-B

Enter the organization phone no

4566678

Enter the organization contact person

Sweta

Tname

Blue

Green

Pink

Red

Yellow

Enter the number of teams to be linked to

1

Enter the team name to be associated with

Pink

Enter your option....

7

Enter the organization name

Allen

Enter the organization address

1199-H

Enter the organization phone no

4558769

Enter the organization contact person

Muthu

Tname

Blue

Green

Pink

Red

Yellow

Enter the number of teams to be linked to

1

Enter the team name to be associated with

Green

EXTERNAL_ORGANIZATION TABLE

	O_NAME	ADDRESS	PHNO	CONTACT
1	Krish	1009-F	5619101	Kim
2	Jerry	1156-B	5619088	Praveen
3	Tom	1008-H	4568333	Solomon
4	Mike	1122-B	4566678	Sweta
5	Allen	1199-H	4558769	Muthu

SPONSORS TABLE

	O_NAME	T_NAME
1	Jerry	Blue
2	Krish	Green
3	Krish	Red
4	Tom	Yellow
5	Mike	Pink
6	Allen	Green

eclipse-workspace - testjdbc/src/testjdbc/lp.java - Eclipse

File Edit Source Refactor Navigate Search Project Run Window Help

Conn.java lp.java

```
228     System.out.println("Enter the organization phone no");
229     String op=r.readLine();
230     System.out.println("Enter the organization contact person");
231     String ocp=r.readLine();
232     st.executeUpdate("insert into External_Organization values('"+on+"','"+oa+"','"+op+"','"+ocp+"')");
233     ResultSet rs6=st.executeQuery("select t_name from Team");
234     System.out.println("Tname\n");
235     while(rs6.next()) {
236         System.out.println(rs6.getString(1));
237     }
238     System.out.println("Enter the number of teams to be linked to");
239     int o1=Integer.parseInt(r.readLine());
240     while(o1>0) {
241         System.out.println("Enter the team name to be associated with");
242         String ot=(r.readLine());
243         st.executeUpdate("insert into Sponsors values('"+on+"','"+ot+"')");
244         o1--;
245     }
246     break;
247 case 8: int ds=input();
    System.out.println("Donor details....");
```

Console

```
ip [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 1:49:12 PM)
Enter your option....
7
Enter the organization name
Allen
Enter the organization address
1199-H
Enter the organization phone no
4558769
Enter the organization contact person
Muthu
Tname
```

Type here to search

1:50 PM
11/19/2017

TASK 6.8 Query 8 Output

Enter your option....

8

Enter the Person details first

Enter the Person SSN

9871

Enter the Person name

Saranya

Enter the Person DOB

9-dec-1997

Enter the Person Race

Asian

Enter the Person Gender

Female

Enter the Person Profession

Student

Enter the Person Contact details..

Enter the Person mailing address

1013-B

Enter the Person Email

sara@gmail.com

Enter the Person Home_ph_no

5689643

Enter the Person Work_ph_no

4056732

Enter the Person Cell_ph_no

5497654

Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list

1

Donor details....

Enter whether the donor wishes to remain anonymous: yes or no

yes

Donation details..

Enter the no of donations

1

Enter 1 for cheque and 2 for Card...

1

Enter the date of donation

20-aug-2017

Enter the donation amount

20000

Enter the type of donation

Food

Enter the fund name of donation

Ou

Enter the cheque no

18896

Enter your option....

8

Enter the Person details first

Enter the Person SSN
8645
Enter the Person name
vishal
Enter the Person DOB
3-apr-1997
Enter the Person Race
American
Enter the Person Gender
Male
Enter the Person Profession
Sports
Enter the Person Contact details..

Enter the Person mailing address
1134-C
Enter the Person Email
vishal@gmail.com
Enter the Person Home_ph_no
4058764
Enter the Person Work_ph_no
4096753
Enter the Person Cell_ph_no
4097865
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
1
Donor details....
Enter whether the donor wishes to remain anonymous: yes or no
no
Donation details..
Enter the no of donations
2
Enter 1 for cheque and 2 for Card...
1
Enter the date of donation
10-jul-2017
Enter the donation amount
1000
Enter the type of donation
Food
Enter the fund name of donation
Osu
Enter the cheque no
19876
Enter 1 for cheque and 2 for Card...
2
Enter the date of donation
19-sep-2017
Enter the donation amount
3600
Enter the type of donation
Organ
Enter the fund name of donation
Osu

Enter the card no
456784
Enter the card type
Credit
Enter the card expiry date
28-dec-2022
Enter your option....
8
Enter the Person details first

Enter the Person SSN
3818
Enter the Person name
Doris
Enter the Person DOB
12-apr-1995
Enter the Person Race
Asian
Enter the Person Gender
Female
Enter the Person Profession
Professor
Enter the Person Contact details..

Enter the Person mailing address
1109-A
Enter the Person Email
doris@gmail.com
Enter the Person Home_ph_no
4056773
Enter the Person Work_ph_no
4098997
Enter the Person Cell_ph_no
4056532
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
1
Donor details....
Enter whether the donor wishes to remain anonymous: yes or no
yes
Donation details..
Enter the no of donations
1
Enter 1 for cheque and 2 for Card...
1
Enter the date of donation
15-aug-2017
Enter the donation amount
5000
Enter the type of donation
Blood
Enter the fund name of donation
Csu
Enter the cheque no
34215

Enter your option....
8
Enter the Person details first

Enter the Person SSN
4599
Enter the Person name
Shruthi
Enter the Person DOB
16-oct-1994
Enter the Person Race
Asian
Enter the Person Gender
Female
Enter the Person Profession
Professor
Enter the Person Contact details..

Enter the Person mailing address
1173-H
Enter the Person Email
shru@gmail.com
Enter the Person Home_ph_no
5466754
Enter the Person Work_ph_no
4056780
Enter the Person Cell_ph_no
4097780
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
1
Donor details....
Enter whether the donor wishes to remain anonymous: yes or no
no
Donation details..
Enter the no of donations
1
Enter 1 for cheque and 2 for Card...
1
Enter the date of donation
11-jan-2017
Enter the donation amount
5433
Enter the type of donation
Organ
Enter the fund name of donation
Asu
Enter the cheque no
45322
Enter your option....
8
Enter the Person details first

Enter the Person SSN
5331

Enter the Person name
Dev
Enter the Person DOB
24-jul-1996
Enter the Person Race
Asian
Enter the Person Gender
Male
Enter the Person Profession
Professor
Enter the Person Contact details..

Enter the Person mailing address
1199-C
Enter the Person Email
dev@gmail.com
Enter the Person Home_ph_no
4056655
Enter the Person Work_ph_no
4098976
Enter the Person Cell_ph_no
4055612
Enter 1 for Yes 0 for No whether the person wishes to be in the monthly news letter list
1
Donor details....
Enter whether the donor wishes to remain anonymous: yes or no
yes
Donation details..
Enter the no of donations
2
Enter 1 for cheque and 2 for Card...
2
Enter the date of donation
22-sep-2017
Enter the donation amount
2000
Enter the type of donation
Food
Enter the fund name of donation
Csu
Enter the card no
234566
Enter the card type
Debit
Enter the card expiry date
22-nov-2027
Enter 1 for cheque and 2 for Card...
2
Enter the date of donation
29-oct-2017
Enter the donation amount
3000
Enter the type of donation
Organ

Enter the fund name of donation

Csu

Enter the card no

234566

Enter the card type

Debit

Enter the card expiry date

22-nov-2027

PERSON TABLE

	P_SSN	P_NAME	DOB	RACE	GENDER	PROFESSION	ADDRESS	EMAIL	HOME_PHNO	WORK_PHNO	CELL_PHNO	MAILING_LIST
1	7195	Joe	17-AUG-95	Asian	Female	Director	1009-C	joe@gmail.com	5617889	7945561	9066432	1
2	5331	Dev	24-JUL-96	Asian	Male	Professor	1199-C	dev@gmail.com	4056655	4098976	4055612	1
3	1996	James	05-JAN-96	American	Male	Engineer	1189-B	james@gmail.com	3452987	9658437	5439977	0
4	6832	Sharon	28-FEB-95	Asian	Female	Professor	1161-B	sharon@gmail.com	4097820	4056155	7890543	1
5	4054	Shalini	31-MAR-96	Asian	Female	Counselor	1115-B	shal@gmail.com	4567880	5619090	8017898	1
6	5897	Lavanya	30-DEC-95	American	Female	PA	1110-H	lav@gmail.com	7884950	4056778	8015619	1
7	1876	Tharun	07-DEC-94	Asian	Male	Student	1121-H	tharun@gmail.com	5673214	4321297	5649325	1
8	4321	Adams	12-SEP-60	American	Male	Professor	1017-H	adams@gmail.com	4568876	5619089	5619041	1
9	3459	Karma	16-SEP-96	American	Male	Student	1017-B	karma@gmail.com	5557893	6734597	4098754	1
10	1492	Evelin	28-DEC-95	Asian	Female	Nurse	1002-H	eve@gmail.com	4670986	8905432	5439655	1
11	2409	Janani	19-MAR-96	Asian	Female	Assistant	1165-A	janani@gmail.com	4356733	7654998	4556784	0
12	4599	Shruthi	16-OCT-94	Asian	Female	Professor	1173-H	shru@gmail.com	5466754	4056780	4097780	1
13	5823	Ria	12-JAN-95	Asian	Female	Student	1106-A	ria@gmail.com	8019076	5614563	7832561	1
14	6178	Tina	14-MAR-96	Asian	Female	Student	1071-C	tina@gmail.com	5637832	6784321	5014563	0
15	9871	Saranya	09-DEC-97	Asian	female	Student	1013-B	sara@gmail.com	5689643	4056732	5497654	1
16	8645	vishal	03-APR-97	American	Male	Sports	1134-C	vishal@gmail.com	4058764	4096753	4097865	1
17	3818	Doris	12-APR-95	Asian	Female	Professor	1109-A	doris@gmail.com	4056773	4098997	4056532	1
18	1296	Priya	28-AUG-93	Asian	Female	Professor	1110-C	priya@gmail.com	5609076	8015634	5619077	1
19	2807	Simon	05-APR-94	American	Male	Professor	1104-H	simon@gmail.com	7820561	5690891	7801820	0
20	6413	Saron	28-AUG-95	Asian	Female	Assistant	1173-B	saron@gmail.com	4566321	4098799	4097765	0

DONOR TABLE

	P_SSN	ANONYMOUS
1	9871	yes
2	8645	no
3	3818	yes
4	4599	no
5	5331	yes

CHEQUE TABLE

	P_SSN	DDATE	AMOUNT	TYPE	FUND_NAME	CHEQUE_NO
1	9871	20-AUG-17	20000	Food	Ou	18896
2	8645	10-JUL-17	1000	Food	Osu	19876
3	3818	15-AUG-17	5000	Blood	Csu	34215
4	4599	11-JAN-17	5433	Organ	Asu	45322

CARD TABLE

	P_SSN	DDATE	AMOUNT	TYPE	FUND_NAME	C_NO	C_TYPE	EXP_DATE
1	5331	29-OCT-17	3000	Organ	Csu	234566	Debit	22-NOV-27
2	8645	19-SEP-17	3600	Organ	Osu	456784	Credit	28-DEC-22
3	5331	22-SEP-17	2000	Food	Csu	234566	Debit	22-NOV-27

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** eclipse-workspace - testjdbc/src/testjdbc/lp.java - Eclipse
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Toolbars:** Standard, Java, Database, Java Persistence API, Java Swing, Java SWT, Java XML, Java JBoss Seam, Java JBoss Seam 2, Java JBoss Seam 3, Java JBoss Seam 4, Java JBoss Seam 5, Java JBoss Seam 6, Java JBoss Seam 7, Java JBoss Seam 8, Java JBoss Seam 9, Java JBoss Seam 10, Java JBoss Seam 11, Java JBoss Seam 12, Java JBoss Seam 13, Java JBoss Seam 14, Java JBoss Seam 15, Java JBoss Seam 16, Java JBoss Seam 17, Java JBoss Seam 18, Java JBoss Seam 19, Java JBoss Seam 20, Java JBoss Seam 21, Java JBoss Seam 22, Java JBoss Seam 23, Java JBoss Seam 24, Java JBoss Seam 25, Java JBoss Seam 26, Java JBoss Seam 27, Java JBoss Seam 28, Java JBoss Seam 29, Java JBoss Seam 30, Java JBoss Seam 31, Java JBoss Seam 32, Java JBoss Seam 33, Java JBoss Seam 34, Java JBoss Seam 35, Java JBoss Seam 36, Java JBoss Seam 37, Java JBoss Seam 38, Java JBoss Seam 39, Java JBoss Seam 40, Java JBoss Seam 41, Java JBoss Seam 42, Java JBoss Seam 43, Java JBoss Seam 44, Java JBoss Seam 45, Java JBoss Seam 46, Java JBoss Seam 47, Java JBoss Seam 48, Java JBoss Seam 49, Java JBoss Seam 50, Java JBoss Seam 51, Java JBoss Seam 52, Java JBoss Seam 53, Java JBoss Seam 54, Java JBoss Seam 55, Java JBoss Seam 56, Java JBoss Seam 57, Java JBoss Seam 58, Java JBoss Seam 59, Java JBoss Seam 60, Java JBoss Seam 61, Java JBoss Seam 62, Java JBoss Seam 63, Java JBoss Seam 64, Java JBoss Seam 65, Java JBoss Seam 66, Java JBoss Seam 67, Java JBoss Seam 68, Java JBoss Seam 69, Java JBoss Seam 70, Java JBoss Seam 71, Java JBoss Seam 72, Java JBoss Seam 73, Java JBoss Seam 74, Java JBoss Seam 75, Java JBoss Seam 76, Java JBoss Seam 77, Java JBoss Seam 78, Java JBoss Seam 79, Java JBoss Seam 80, Java JBoss Seam 81, Java JBoss Seam 82, Java JBoss Seam 83, Java JBoss Seam 84, Java JBoss Seam 85, Java JBoss Seam 86, Java JBoss Seam 87, Java JBoss Seam 88, Java JBoss Seam 89, Java JBoss Seam 90, Java JBoss Seam 91, Java JBoss Seam 92, Java JBoss Seam 93, Java JBoss Seam 94, Java JBoss Seam 95, Java JBoss Seam 96, Java JBoss Seam 97, Java JBoss Seam 98, Java JBoss Seam 99, Java JBoss Seam 100, Java JBoss Seam 101, Java JBoss Seam 102, Java JBoss Seam 103, Java JBoss Seam 104, Java JBoss Seam 105, Java JBoss Seam 106, Java JBoss Seam 107, Java JBoss Seam 108, Java JBoss Seam 109, Java JBoss Seam 110, Java JBoss Seam 111, Java JBoss Seam 112, Java JBoss Seam 113, Java JBoss Seam 114, Java JBoss Seam 115, Java JBoss Seam 116, Java JBoss Seam 117, Java JBoss Seam 118, Java JBoss Seam 119, Java JBoss Seam 120, Java JBoss Seam 121, Java JBoss Seam 122, Java JBoss Seam 123, Java JBoss Seam 124, Java JBoss Seam 125, Java JBoss Seam 126, Java JBoss Seam 127, Java JBoss Seam 128, Java JBoss Seam 129, Java JBoss Seam 130, Java JBoss Seam 131, Java JBoss Seam 132, Java JBoss Seam 133, Java JBoss Seam 134, Java JBoss Seam 135, Java JBoss Seam 136, Java JBoss Seam 137, Java JBoss Seam 138, Java JBoss Seam 139, Java JBoss Seam 140, Java JBoss Seam 141, Java JBoss Seam 142, Java JBoss Seam 143, Java JBoss Seam 144, Java JBoss Seam 145, Java JBoss Seam 146, Java JBoss Seam 147, Java JBoss Seam 148, Java JBoss Seam 149, Java JBoss Seam 150, Java JBoss Seam 151, Java JBoss Seam 152, Java JBoss Seam 153, Java JBoss Seam 154, Java JBoss Seam 155, Java JBoss Seam 156, Java JBoss Seam 157, Java JBoss Seam 158, Java JBoss Seam 159, Java JBoss Seam 160, Java JBoss Seam 161, Java JBoss Seam 162, Java JBoss Seam 163, Java JBoss Seam 164, Java JBoss Seam 165, Java JBoss Seam 166, Java JBoss Seam 167, Java JBoss Seam 168, Java JBoss Seam 169, Java JBoss Seam 170, Java JBoss Seam 171, Java JBoss Seam 172, Java JBoss Seam 173, Java JBoss Seam 174, Java JBoss Seam 175, Java JBoss Seam 176, Java JBoss Seam 177, Java JBoss Seam 178, Java JBoss Seam 179, Java JBoss Seam 180, Java JBoss Seam 181, Java JBoss Seam 182, Java JBoss Seam 183, Java JBoss Seam 184, Java JBoss Seam 185, Java JBoss Seam 186, Java JBoss Seam 187, Java JBoss Seam 188, Java JBoss Seam 189, Java JBoss Seam 190, Java JBoss Seam 191, Java JBoss Seam 192, Java JBoss Seam 193, Java JBoss Seam 194, Java JBoss Seam 195, Java JBoss Seam 196, Java JBoss Seam 197, Java JBoss Seam 198, Java JBoss Seam 199, Java JBoss Seam 200, Java JBoss Seam 201, Java JBoss Seam 202, Java JBoss Seam 203, Java JBoss Seam 204, Java JBoss Seam 205, Java JBoss Seam 206, Java JBoss Seam 207, Java JBoss Seam 208, Java JBoss Seam 209, Java JBoss Seam 210, Java JBoss Seam 211, Java JBoss Seam 212, Java JBoss Seam 213, Java JBoss Seam 214, Java JBoss Seam 215, Java JBoss Seam 216, Java JBoss Seam 217, Java JBoss Seam 218, Java JBoss Seam 219, Java JBoss Seam 220, Java JBoss Seam 221, Java JBoss Seam 222, Java JBoss Seam 223, Java JBoss Seam 224, Java JBoss Seam 225, Java JBoss Seam 226, Java JBoss Seam 227, Java JBoss Seam 228, Java JBoss Seam 229, Java JBoss Seam 230, Java JBoss Seam 231, Java JBoss Seam 232, Java JBoss Seam 233, Java JBoss Seam 234, Java JBoss Seam 235, Java JBoss Seam 236, Java JBoss Seam 237, Java JBoss Seam 238, Java JBoss Seam 239, Java JBoss Seam 240, Java JBoss Seam 241, Java JBoss Seam 242, Java JBoss Seam 243, Java JBoss Seam 244, Java JBoss Seam 245, Java JBoss Seam 246, Java JBoss Seam 247, Java JBoss Seam 248, Java JBoss Seam 249, Java JBoss Seam 250, Java JBoss Seam 251, Java JBoss Seam 252, Java JBoss Seam 253, Java JBoss Seam 254, Java JBoss Seam 255, Java JBoss Seam 256, Java JBoss Seam 257, Java JBoss Seam 258, Java JBoss Seam 259, Java JBoss Seam 260, Java JBoss Seam 261, Java JBoss Seam 262, Java JBoss Seam 263, Java JBoss Seam 264.
- Console Output:**

```
Ip [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 1:49:12 PM)
Green
Enter your option...
8
Enter the Person details first

Enter the Person SSN
9871
Enter the Person name
Saranya
Enter the Person DOB
9-dec-1997
Enter the Person Race
<
```

TASK 6.9 Query 9 Output

```
Enter your option....
9
Enter the organization name
Bob
Enter the organization address
1111-H
Enter the organization phone no
4553217
Enter the organization contact person
Amen
Organization Donor details....
Enter whether the org_donor wishes to remain anonymous: yes or no
yes
Org_Donation details..
Enter the no of donations
1
Enter 1 for cheque and 2 for Card...
2
Enter the date of donation
23-aug-2016
Enter the donation amount
3455
Enter the type of donation
Food
Enter the fund name of donation
Ou
```

Enter the card no
234566

Enter the card type
Credit

Enter the card expiry date
23-dec-2029

Enter your option....
9

Enter the organization name
Alice

Enter the organization address
1222-H

Enter the organization phone no
4589098

Enter the organization contact person
Joseph

Organization Donor details....

Enter whether the org_donor wishes to remain anonymous: yes or no
no

Org_Donation details..

Enter the no of donations
1

Enter 1 for cheque and 2 for Card...
1

Enter the date of donation
1-mar-2017

Enter the donation amount
20000

Enter the type of donation
Organ

Enter the fund name of donation
Osu

Enter the cheque no
345778

Enter your option....
9

Enter the organization name
Catrina

Enter the organization address
1209-A

Enter the organization phone no
5609822

Enter the organization contact person
Jenae

Organization Donor details....

Enter whether the org_donor wishes to remain anonymous: yes or no
yes

Org_Donation details..

Enter the no of donations
1

Enter 1 for cheque and 2 for Card...
1

Enter the date of donation
30-sep-2016

Enter the donation amount

5678
Enter the type of donation
Blood
Enter the fund name of donation
Csu
Enter the cheque no
765432
Enter your option....
9
Enter the organization name
Emily
Enter the organization address
1176-B
Enter the organization phone no
4509052
Enter the organization contact person
Bryan
Organization Donor details....
Enter whether the org_donor wishes to remain anonymous: yes or no
yes
Org_Donation details..
Enter the no of donations
1
Enter 1 for cheque and 2 for Card...
1
Enter the date of donation
22-jul-2015
Enter the donation amount
4800
Enter the type of donation
Blood
Enter the fund name of donation
Osu
Enter the cheque no
765889
Enter your option....
9
Enter the organization name
Edward
Enter the organization address
1121-F
Enter the organization phone no
4056678
Enter the organization contact person
Anna
Organization Donor details....
Enter whether the org_donor wishes to remain anonymous: yes or no
yes
Org_Donation details..
Enter the no of donations
1
Enter 1 for cheque and 2 for Card...
1
Enter the date of donation
18-mar-2016

Enter the donation amount

5600

Enter the type of donation

Food

Enter the fund name of donation

Ou

Enter the cheque no

345556

EXTERNAL_ORGANIZATION TABLE

	O_NAME	ADDRESS	PHNO	CONTACT
1	Krish	1009-F	5619101	Kim
2	Jerry	1156-B	5619088	Praveen
3	Tom	1008-H	4568333	Solomon
4	Mike	1122-B	4566678	Sweta
5	Allen	1199-H	4558769	Muthu
6	Bob	1111-H	4553217	Amen
7	Alice	1222-H	4589098	Joseph
8	Catrina	1209-A	5609822	Jenae
9	Emily	1176-B	4509052	Bryan
10	Edward	1121-F	4056678	Anna

ORG_DONORS TABLE

	O_NAME	ANONYMOUS
1	Bob	yes
2	Alice	no
3	Catrina	yes
4	Emily	yes
5	Edward	yes

ORG_CHEQUE TABLE

	O_NAME	ORG_DATE	ORG_AMOUNT	ORG_TYPE	ORG_FUND_NAME	CHEQUE_NO
1	Alice	01-MAR-17	20000	Organ	Osu	345778
2	Catrina	30-SEP-16	5678	Blood	Csu	765432
3	Emily	22-JUL-15	4800	Blood	Osu	765889
4	Edward	18-MAR-16	5600	Food	Ou	345556

ORG_CARD TABLE

	O_NAME	ORG_DATE	ORG_AMOUNT	ORG_TYPE	ORG_FUND_NAME	C_NO	C_TYPE	EXP_DATE
1	Bob	23-AUG-16	3455	Food	Ou	234566	Credit	23-DEC-29

The screenshot shows the Eclipse IDE interface with the following details:

- File Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Quick Access:** A search bar at the top right.
- Conn.java:** An open Java file containing code for a JDBC application. It includes imports for java.sql.* and javax.sql.*. The code handles various user inputs and database queries related to card expiry, client SSNs, and doctor details.
- Ip.java:** Another open Java file containing code. The current line of interest is "case 10: System.out.println("Displaying Client SSNs..");".
- Console:** A terminal window showing the execution of the application. The output includes prompts for organization details, donor information, and donation counts. It then displays "Clients SSN" followed by a list of SSNs: 4321, 6832, 1876, 1996, 7195. Finally, it asks for a Client SSN to get doctor details, with the input "4321" shown.
- Taskbar:** Shows the Windows taskbar with various pinned icons like File Explorer, Edge, and File Manager.
- System Tray:** Shows the date and time as 3:44 PM on 11/19/2017.

TASK 6.10 Query 10 Output

Enter your option....

10

Displaying Client SSNs..

Clients SSN

4321

6832

1876

1996

7195

Enter the Client SSN to get the doctor details

4321

Doctor Name Doctor phone_number

John 8017820

Enter your option....

10

Displaying Client SSNs..

Clients SSN

4321

6832

1876

1996

7195

Enter the Client SSN to get the doctor details

7195

Doctor Name Doctor phone_number
John 8764322

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** eclipse-workspace - testjdbc/src/testjdbc/lp.java - Eclipse
- Left Margin:** Shows line numbers 333 through 352.
- Code Editor:** Displays Java code for a JDBC application. The code includes printing expiry dates, executing insertions into the Org_Card table, displaying Client SSNs, and retrieving doctor details by Client SSN. It also handles date ranges for employee expenses.
- Console:** Shows the output of the application:

```
lp [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 1:49:12 PM)
345556
Enter your option...
10
Displaying Client SSNs..
Clients SSN
4321
6832
1876
1996
7195
Enter the Client SSN to get the doctor details
```
- Windows Taskbar:** Shows the Windows taskbar with various pinned icons like File Explorer, Edge, and File History.

TASK 6.11 Query 11 Output

Enter your option....

11

Enter the date range to retrieve the employee expense

Enter the starting date

10-jan-2017

Enter the ending date

20-jul-2017

Employee_SSN Total expense

2409 79.22

1492 500

4054 1000

Enter your option....

11

Enter the date range to retrieve the employee expense

Enter the starting date

20-apr-2017

Enter the ending date

12-sep-2017

Employee_SSN Total expense

4054 1478.07

The screenshot shows the Eclipse IDE interface with the following details:

- File Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Quick Access:** A search bar at the top right.
- Conn.java:** An open Java file containing code for a JDBC application. The code includes various SQL queries for inserting data into tables like Org_Card and retrieving data from tables like Client and Doctor.
- Ip.java:** Another open Java file containing the main logic of the application. It handles user input for card expiry date, client SSNs, doctor details, and employee expenses.
- Console:** A terminal window showing the execution of the Java application. The output includes:
 - Java Application C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 1:49:12 PM)
 - 1800
 - Enter your option....
 - 11
 - Enter the date range to retrieve the employee expense
 - Enter the starting date
 - 28-apr-2017
 - Enter the ending date
 - 12-sep-2017
 - Employee_SSN Total expense
 - 4054 1478.07
 - Enter your option....
- Taskbar:** Shows the Windows taskbar with various pinned icons and the system tray indicating the date and time (11/19/2017, 3:55 PM).

TASK 6.12 Query 12 Output

Enter your option....
12
Displaying Client SSNs..
1996
1876
1996
4321
6832
7195
Enter the Client SSN to retrieve the Volunteer details..
1876
PSSN Name
5823 Ria
3459 Karma
Enter your option....
12
Displaying Client SSNs..
1996
1876
1996
4321
6832
7195
Enter the Client SSN to retrieve the Volunteer details..
4321
PSSN Name
2807 Simon

1296 Priya

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** eclipse-workspace - testjdbc/src/testjdbc/lp.java - Eclipse
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Left Sidebar:** Conn.java, lp.java
- Code Editor (lp.java):**

```
356     String edate=(r.readLine());
357     System.out.println("Employee_SSN"+ "\t"+ "Total expense");
358     ResultSet rs9=st.executeQuery("select P_SSN,sum(E_amount) from expense where E_date between '"+sdate+"' and '"+edate+"' group by P_SSN order b
359     while(rs9.next()) {
360         System.out.println(rs9.getString(1)+"\t"+rs9.getString(2));
361     }
362     break;
363     case 12: System.out.println("Displaying Client SSNs..");
364     ResultSet rs10=st.executeQuery("select P_SSN from cares");
365     while(rs10.next()) {
366         System.out.println(rs10.getString(1));
367     }
368     System.out.println("Enter the Client SSN to retrieve the Volunteer details..");
369     int rsn=Integer.parseInt(r.readLine());
370     ResultSet rs11=st.executeQuery("select P_SSN,P_Name from Person where P_SSN in (select P_SSN from serves where T_name in (select T_name from c
371     System.out.println("PSSN"+ "\t"+ "Name");
372     while(rs11.next()) {
373         System.out.println(rs11.getString(1)+"\t"+rs11.getString(2));
374     }
375 }
```
- Console Output:**

```
lp [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 1:49:12 PM)
Enter your option....
12
Displaying Client SSNs..
1996
1876
1996
4321
6832
7195
Enter the Client SSN to retrieve the Volunteer details..
1876
PSSN    Name
<          >
```
- System Tray:** Type here to search, Task View, Start button, Taskbar icons, Date/Time: 4:01 PM, 11/19/2017.

TASK 6.13 Query 13 Output

Enter your option....

13

Name	Address	Email	Homephno	workphno	cellphno
James	1189-B	james@gmail.com	3452987	9658437	5439977
Joe	1009-C	joe@gmail.com	5617889	7945561	9066432
Sharon	1161-B	sharon@gmail.com	4097820	4056155	7890543
Tharun	1121-H	tharun@gmail.com	5673214	4321297	5649325

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** eclipse-workspace - testjdbc/src/testjdbc/lp.java - Eclipse
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Left Sidebar:** Conn.java, lp.java
- Code Editor (lp.java):**

```
368     System.out.println("Enter the Client SSN to retrieve the Volunteer details..");
369     int rsn=Integer.parseInt(r.readLine());
370     ResultSet rs11=st.executeQuery("select P_SSN,P_Name from Person where P_SSN in (select P_SSN from serves where T_name in (select T_name from c
371     System.out.println("PSSN"+ "\t"+ "Name");
372     while(rs11.next()) {
373         System.out.println(rs11.getString(1)+"\t"+rs11.getString(2));
374     }
375 }
376     break;
377     case 13: System.out.println("Name"+ "\t"+ "Address"+ "\t"+ "Email"+ "\t"+ "Homephno"+ "\t"+ "workphno"+ "\t"+ "cellphno");
378     ResultSet rs12=st.executeQuery("select P_Name,Address, Email, Home_phno, Work_phno, Cell_pno from Person where P_SSN in(select P_SSN from car
379     while(rs12.next()) {
380         System.out.println(rs12.getString(1)+"\t"+rs12.getString(2)+"\t"+rs12.getString(3)+"\t"+rs12.getString(4)+"\t"+rs12.getString(5)+"\t"
381     }
382     break;
383     case 14: CallableStatement cs1=dbc.prepareCall("{call input}");
384     cs1.execute();
385     cs1.close();
386     break;
387 }
```
- Console Output:**

```
lp [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 1:49:12 PM)
2007  Simon
1296  Priya
Enter your option....
13
Name    Address      Email      Homephno    workphno    cellphno
James   1189-B   james@gmail.com  3452987    9658437    5439977
Joe     1009-C   joe@gmail.com   5617889    7945561    9066432
Sharon  1161-B   sharon@gmail.com 4097820    4056155    7890543
Tharun  1121-H   tharun@gmail.com 5673214    4321297    5649325
Enter your option....
```
- System Tray:** Type here to search, Task View, Start button, Taskbar icons, Date/Time: 4:03 PM, 11/19/2017.

TASK 6.14 Query 14 Output

Enter your option....

14

Retrieving name and total amount donated by donors who are also employees

Name	Anonymous	Total Amount
vishal	no	3600

The screenshot shows the Eclipse IDE interface with two tabs open: Conn.java and lp.java. The lp.java tab contains Java code for a switch statement. Case 13 prints names, addresses, emails, home phones, work phones, and cell phones. Case 14 prints names and total amounts for donors who are also employees. Case 15 prints names, addresses, emails, and max hours between March and June. The code uses ResultSet objects to execute queries and print results. The lp.java tab has line numbers 387 to 480.

```
387
388     while(rs11.next()) {
389         System.out.println(rs11.getString(1)+"\t"+rs11.getString(2));
390     }
391     break;
392     //Retrieving the names and contact information of the clients that are supported by teams sponsored by an organization whose name starts with
393     case 13: System.out.println("Name"+ "\t"+ "Address"+ "\t"+ "Email"+ "\t"+ "Homephno"+ "\t"+ "workphno"+ "\t"+ "cellphno");
394     ResultSet rs12=st.executeQuery("select P_Name,Address, Email, Home_phno, Work_phno, Cell_phno from Person where P_SSN in(select P_SSN from car
395     while(rs12.next()) {
396         System.out.println(rs12.getString(1)+"\t"+rs12.getString(2)+"\t"+rs12.getString(3)+"\t"+rs12.getString(4)+"\t"+rs12.getString(5)+"\t");
397     }
398     break;
399     case 14: System.out.println("Retrieving name and total amount donated by donors who are also employees");
400     ResultSet rs13=st.executeQuery("select p.P_Name,d.Anonymous,amount from (select am.P_SSN ssn,max(am.Amount) amount from (select c.P_SSN,c.Amou
401     System.out.println("Name"+ "\t"+ "Anonymous"+ "\t"+ "Total Amount");
402     while(rs13.next()) {
403         System.out.println(rs13.getString(1)+"\t"+rs13.getString(2)+"\t"+rs13.getString(3));
404     }
405     break;
406     //Retrieving name and contact info of volunteers who worked max hours between March and June
407     case 15: System.out.println("Retrieving name and contact information of the volunteers...");
408     System.out.println("Name"+ "\t"+ "Address"+ "\t"+ "Email"+ "\t"+ "Homephno"+ "\t"+ "workphno"+ "\t"+ "cellphno").
```

Console []

```
lp [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 20, 2017, 12:23:44 AM)
19. Export option-Retrieve all data from Team and write to a file
20. Exit from program
Enter your option...
14
Retrieving name and total amount donated by donors who are also employees
Name      Anonymous      Total Amount
vishal    no            3600
Enter your option...
```

Type here to search 12:23 AM 11/20/2017

TASK 6.15 Query 15 Output

Enter your option....

15

Retrieving name and contact information of the volunteers...

Name	Address	Email	Homephno	workphno	cellphno
Karma	1017-B	karma@gmail.com	5557893	6734597	4098754
Karma	1017-B	karma@gmail.com	5557893	6734597	4098754
Ria	1106-A	ria@gmail.com	8019076	5614563	7832561
Tina	1071-C	tina@gmail.com	5637832	6784321	5014563
Tina	1071-C	tina@gmail.com	5637832	6784321	5014563
Simon	1104-H	simon@gmail.com	7820561	5690891	7801820

eclipse-workspace - testjdbc/src/testjdbc/lp.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Conn.java lp.java Quick Access
397
398
399
400
401
402
403 person_P_SSN=serves_P_SSN and (serves.t_name,serves.hours) in (select T_name,max(hours) from serves where Month in ('March','April','May','June') group by T_name");
404
405 "\t\t"+rs14.getString(5)+"\t\t"+rs14.getString(6);
406
407
408
409
410 SSN having count(P_SSN)>1");
411
412
413
414
415 );
416
417
418 <

```

Console ::

```

lp [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 7:45:20 PM)
Enter your option....
15
Retrieving name and contact information of the volunteers...
Name Address Email Homephno workphno cellphno
Karma 1071-A karma@gmail.com 5557893 674597 4098754
Karma 1071-B karma@gmail.com 5557893 6734597 4098754
Ria 1106-A ria@gmail.com 8015976 5614563 7832561
Tina 1071-C tina@gmail.com 5637832 6784321 5014563
Tina 1071-C tina@gmail.com 5637832 6784321 5014563
Simon 1104-H simon@gmail.com 7820561 5690891 7801820
<
```

Type here to search 7:48 PM 11/19/2017

TASK 6.16 Query 16 Output

Enter your option....

16

Salary Updated...

EmployeeSSN	Salary	Marital_status	Hire_date
4054	3814.67	Single	2016-08-07
1492	4950	Single	2017-01-20
2409	2205.73	Single	2016-04-20
5897	4900	Single	2015-10-20
6413	3500	Married	2015-03-18

eclipse-workspace - testjdbc/src/testjdbc/lp.java - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
Conn.java lp.java Quick Access
417 case 18: System.out.println("Enter import file name: ");
418     String import_file_name=readline();
419     FileInputStream fstream = new FileInputStream("C:/Users/rvino/Desktop/"+import_file_name);
420     DataInputStream in = new DataInputStream(fstream);
421     BufferedReader br18 = new BufferedReader(new InputStreamReader(in));
422     String strLine;
423     while ((strLine = br18.readLine()) != null)
424     {
425         String a[] = strLine.split(",");
426         String name = a[0];
427         String type = a[1];
428         String date=a[2];
429         st.executeUpdate("insert into team values('"+name+"','"+type+"','"+date+"')");
430         System.out.println(" One row inserted successfully");
431     }
432     System.out.println("File imported successfully...");
433     break;
434     case 19: System.out.println("Exporting data to a file..");
435     ResultSet rs18=st.executeQuery("select * from team");
436     System.out.println("Enter output file name: ");

```

Console ::

```

lp [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 4:44:37 PM)
Tina 1071-C tina@gmail.com 5637832 6784321 5014563
Enter your option....
16
Salary Updated...
EmployeeSSN Salary Marital_status Hire_date
4054 3814.67 Single 2016-08-07 00:00:00.0
1492 4950 Single 2017-01-20 00:00:00.0
2409 2205.73 Single 2016-04-20 00:00:00.0
5897 4900 Single 2015-10-20 00:00:00.0
6413 3500 Married 2015-03-18 00:00:00.0
Enter your option....

```

Type here to search 4:51 PM 11/19/2017

TASK 6.17 Query 17 Output

Enter your option....

17

Before deletion...

ClientSSN	Dname	Aname	Dphno	Aphno	Date
4321	John	Jordan	8017820	5619056	2003-08-20
6832	Williams	Amen	7820604	5617820	2007-04-11
1876	John	Jack	8764322	4563297	2000-03-10
1996	Williams	Amen	7820604	5617820	2007-04-11
7195	John	Jack	8764322	4563297	2000-03-10

Deleting Client data...

After deletion...

ClientSSN	Dname	Aname	Dphno	Aphno	Date
6832	Williams	Amen	7820604	5617820	2007-04-11
1876	John	Jack	8764322	4563297	2000-03-10
1996	Williams	Amen	7820604	5617820	2007-04-11
7195	John	Jack	8764322	4563297	2000-03-10

The screenshot shows the Eclipse IDE interface with two open windows. The top window is titled 'Ip.java' and contains Java code for handling file imports and exports. The bottom window is titled 'Console' and shows the command-line output of the application running in a terminal window.

```
case 18: System.out.println("Enter import file name: ");
        String import_file_name=reader.readLine();
        FileInputStream in = new FileInputStream("C:/Users/rvino/Desktop/" + import_file_name);
        DataInputStream in1 = new DataInputStream(in);
        BufferedReader br18 = new BufferedReader(new InputStreamReader(in));
        String strLine;
        while ((strLine = br18.readLine()) != null)
        {
            String a[] = strLine.split(",");
            String name = a[0];
            String type = a[1];
            String date=a[2];
            strLine="insert into team values('"+name+"','"+type+"','"+date+"')";
            System.out.println(" One row inserted successfully");
        }
        System.out.println("File imported successfully...");
        break;
    case 19: System.out.println("Exporting data to a file..");
        ResultSet rs19st=stmt.executeQuery("select * from team");
        System.out.println("Enter output file name: ");
        <
```

```
ip [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 4:44:37 PM)
Enter your option...
17
Before deletion...
ClientSSN      Dname      Aname      Dphno      Aphno      Date
4321          John       Jordan     8017820    5619056    2003-08-20
6832          Williams   Amen      7820604    5617820    2007-04-11
1876          John       Jack      8764322    4563297    2000-03-10
1996          Williams   Amen      7820604    5617820    2007-04-11
7195          John       Jack      8764322    4563297    2000-03-10
Deleting Client data...
ClientSSN      Dname      Aname      Dphno      Aphno      Date
6832          Williams   Amen      7820604    5617820    2007-04-11
1876          John       Jack      8764322    4563297    2000-03-10
1996          Williams   Amen      7820604    5617820    2007-04-11
7195          John       Jack      8764322    4563297    2000-03-10
```

TASK 6.18 Query 18 and 19 (Import and Export options) Output

//Importing data from a file

Enter your option....

18

Enter import file name:

new.txt

One row inserted successfully
One row inserted successfully
File imported successfully...

new.txt



TEAM TABLE

	T_NAME	T_TYPE	T_DATE
1	Pink	Finance	20-SEP-02
2	Yellow	Research	07-MAR-80
3	Red	Testing	09-APR-60
4	Green	Admin	12-JUL-96
5	Blue	Support	17-AUG-00
6	CSU	Research	20-JAN-17
7	UNC	finance	12-JUL-00

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** eclipse-workspace - testjdbc/src/testjdbc/lp.java - Eclipse
- Menu Bar:** File Edit Source Refactor Navigate Project Run Window Help
- Toolbar:** Standard Eclipse toolbar with icons for file operations, search, and navigation.
- Quick Access:** A small window in the top right corner.
- Left Margin:** Shows line numbers from 352 to 571.
- Code Editor:** The lp.java file contains Java code for a JDBC application. It includes cases for displaying employee expense ranges, client SSNs, and volunteer details, along with database query execution and result printing.
- Console:** Shows the command-line interaction with the Java application:
 - Input: lp [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 4:44:37 PM)
 - Output:

```
Enter your option...
18
Enter import file name:
new.txt
One row inserted successfully
One row inserted successfully
File imported successfully...
Enter your option...
19
Exporting data to a file..
Enter output file name:
new5.txt
```
- Bottom:** Windows taskbar with various pinned icons.

//Exporting data to a file

TASK 6.19 Queries for testing three types of errors

1) Violating unique constraint : By inserting a record into organization table with o_name that already exists in the table

Enter your option....

7

Enter the organization name

Krish

Enter the organization address

1123-C

Enter the organization phone no

4563214

Enter the organization contact person

Andrew

[java.sql.SQLIntegrityConstraintViolationException](#): ORA-00001: unique constraint (RAJA0004.SYS_C00188694) violated

```
at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:450)
at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:399)
at oracle.jdbc.driver.T4C80all.processError(T4C80all.java:1059)
at oracle.jdbc.driver.T4CTTIfun.receive(T4CTTIfun.java:522)
at oracle.jdbc.driver.T4CTTIfun.doRPC(T4CTTIfun.java:257)
at oracle.jdbc.driver.T4C80all.doOALL(T4C80all.java:587)
```

```

at oracle.jdbc.driver.T4CStatement.doOall8(T4CStatement.java:210)
at oracle.jdbc.driver.T4CStatement.doOall8(T4CStatement.java:30)
at oracle.jdbc.driver.T4CStatement.executeForRows(T4CStatement.java:931)
at
oracle.jdbc.driver.OracleStatement.doExecuteWithTimeout(OracleStatement.java:1150)
at
oracle.jdbc.driver.OracleStatement.executeUpdateInternal(OracleStatement.java:1707)
at oracle.jdbc.driver.OracleStatement.executeUpdate(OracleStatement.java:1670)
at
oracle.jdbc.driver.OracleStatementWrapper.executeUpdate(OracleStatementWrapper.java:310)
at testjdbc.Ip.main(Ip.java:232)

```

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** eclipse-workspace - testjdbc/src/testjdbc/Ip.java - Eclipse
- Left Sidebar:** Shows files Conn.java and Ip.java.
- Code Editor (Ip.java):**

```

281     int Cn=(Integer.parseInt(r.readLine()));
282     System.out.println("Enter the card type");
283     String cty=(r.readLine());
284     System.out.println("Enter the card expiry date");
285     String exp=(r.readLine());
286     st.executeUpdate("insert into Card values('"+ds+"','"+ddo+"','"+ddat+"','"+dt+"','"+fn+"','"+car+"','"+cty+"','"+exp+"')");
287     }dn-;}
288     break;
289   case 9: System.out.println("Enter the organization name");
290   String on=(r.readLine());
291   System.out.println("Enter the organization address");
292   String oad=(r.readLine());
293   System.out.println("Enter the organization phone no");
294   String op=(r.readLine());
295   System.out.println("Enter the organization contact person");

```
- Console Tab:**

```

<terminated> Ip [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 5:26:14 PM)
Enter your option...
7
Enter the organization name
Krish
Enter the organization address
1123-C
Enter the organization phone no
4563214
Enter the organization contact person
Andrew
java.sql.SQLIntegrityConstraintViolationException: ORA-00001: unique constraint (RAJA0004.SYS_C00188694) violated

```
- Bottom Status Bar:** Shows the taskbar with various icons and the system clock indicating 5:27 PM on 11/19/2017.

2) Literal does not match format string error – entering the date value in a different format which is not identified as valid date

Enter your option....

1

Enter the Team name

Unc

Enter the Team type

Admin

Enter the Team formed date

23-sept-2016

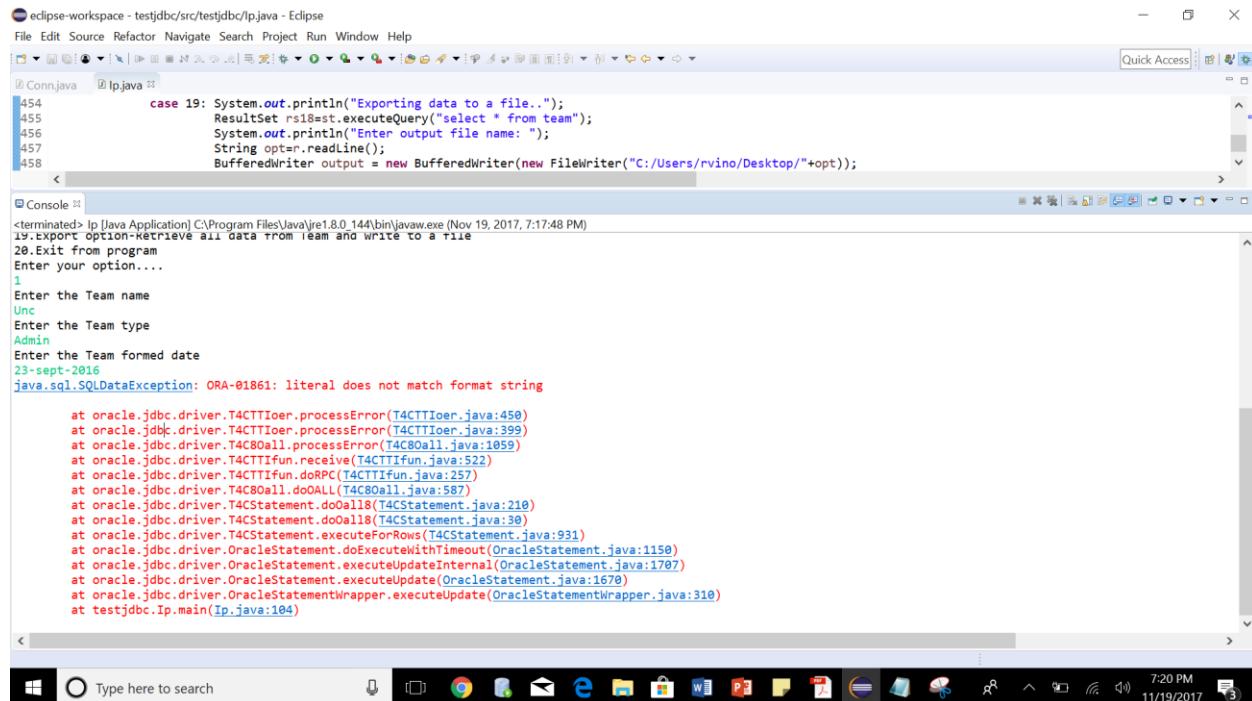
[java.sql.SQLDataException: ORA-01861: literal does not match format string](#)

```

at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:450)
at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:399)
at oracle.jdbc.driver.T4C80all.processError(T4C80all.java:1059)

```

```
at oracle.jdbc.driver.T4CTTIfun.receive(T4CTTIfun.java:522)
at oracle.jdbc.driver.T4CTTIfun.doRPC(T4CTTIfun.java:257)
at oracle.jdbc.driver.T4C8Oall.doOALL(T4C8Oall.java:587)
at oracle.jdbc.driver.T4CStatement.dooall18(T4CStatement.java:210)
at oracle.jdbc.driver.T4CStatement.dooall18(T4CStatement.java:30)
at oracle.jdbc.driver.T4CStatement.executeForRows(T4CStatement.java:931)
at
oracle.jdbc.driver.OracleStatement.doExecuteWithTimeout(OracleStatement.java:1150)
at
oracle.jdbc.driver.OracleStatement.executeUpdateInternal(OracleStatement.java:1707)
at oracle.jdbc.driver.OracleStatement.executeUpdate(OracleStatement.java:1670)
at
oracle.jdbc.driver.OracleStatementWrapper.executeUpdate(OracleStatementWrapper.java:3
10)
at testjdbc.Ip.main(Ip.java:104)
```



3) Primary key violation – Trying insert null values into a primary key attribute

Enter your option....

1

Enter the Team name

Enter the Team type

ou

Enter the Team formed date

22-aug-1998

java.sql.SQLIntegrityConstraintViolationException
("RAJA0004"."TEAM"."T_NAME")

at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:450)

Red dots indicate the positions of the four reference points used to align the two images.

```
at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:399)
at oracle.jdbc.driver.T4C80all.processError(T4C80all.java:1059)
at oracle.jdbc.driver.T4CTTIfun.receive(T4CTTIfun.java:522)
at oracle.jdbc.driver.T4CTTIfun.doRPC(T4CTTIfun.java:257)
at oracle.jdbc.driver.T4C80all.doOALL(T4C80all.java:587)
at oracle.jdbc.driver.T4CStatement.doOall8(T4CStatement.java:210)
at oracle.jdbc.driver.T4CStatement.doOall8(T4CStatement.java:30)
at oracle.jdbc.driver.T4CStatement.executeForRows(T4CStatement.java:931)
at
oracle.jdbc.driver.OracleStatement.doExecuteWithTimeout(OracleStatement.java:1150)
at
oracle.jdbc.driver.OracleStatement.executeUpdateInternal(OracleStatement.java:1707)
at oracle.jdbc.driver.OracleStatement.executeUpdate(OracleStatement.java:1670)
at
oracle.jdbc.driver.OracleStatementWrapper.executeUpdate(OracleStatementWrapper.java:3
10)
at testjdbc.Ip.main(Ip.java:105)
```

eclipse-workspace - testjdbc/src/testjdbc/Ip.java - Eclipse

File Edit Source Refactor Navigate Search Project Run Window Help

Conn.java Ip.java

```
380     while(rs10.next()) {
381         System.out.println(rs10.getString(1));
382     }
383     System.out.println("Enter the Client SSN to retrieve the Volunteer details..");
384     int rssn=Integer.parseInt(br.readLine());
385     ResultSet rs11=st.executeQuery("select P_SSN,P_Name from Person where P_SSN in (select P_SSN from serves where T_name in (select T_name from c
386     System.out.println("PSSN"+rssn+"Name").
```

Console

```
<terminated> Ip [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 20, 2017, 12:23:44 AM)
Enter your option...
1
Enter the Team name

Enter the Team type
ou
Enter the Team formed date
22-aug-1998
java.sql.IntegrityConstraintViolationException: ORA-01400: cannot insert NULL into ("RAJA0004"."TEAM"."T_NAME")

at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:450)
at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:399)
at oracle.jdbc.driver.T4C8Oall.processError(T4C8Oall.java:1059)
at oracle.jdbc.driver.T4CTTIfun.receive(T4CTTIfun.java:522)
at oracle.jdbc.driver.T4CTTIfun.doRPC(T4CTTIfun.java:257)
at oracle.jdbc.driver.T4C8Oall.doALL(T4C8Oall.java:587)
at oracle.jdbc.driver.T4CStatement.doAll18(T4CStatement.java:210)
at oracle.jdbc.driver.T4CStatement.doAll18(T4CStatement.java:30)
at oracle.jdbc.driver.T4CStatement.executeForRows(T4CStatement.java:931)
at oracle.jdbc.driver.OracleStatement.executeUpdateWithTimeout(OracleStatement.java:1158)
at oracle.jdbc.driver.OracleStatement.executeUpdateInternal(OracleStatement.java:1707)
at oracle.jdbc.driver.OracleStatement.executeUpdate(OracleStatement.java:1670)
at oracle.jdbc.driver.OracleStatementWrapper.executeUpdate(OracleStatementWrapper.java:310)
at testjdbc.Ip.main(Ip.java:105)
```

TASK 6.20 Testing the Quit Option

Enter your option....

20

Exiting....

The screenshot shows the Eclipse IDE interface with two open files: Conn.java and Ip.java. The Ip.java file contains Java code for a menu-driven application. The code includes cases for displaying employee expense, displaying client SSNs, and retrieving volunteer details. It uses JDBC to execute SQL queries and print results to the console. The terminal window below shows the application's output, indicating successful record insertions and file exports. The task bar at the bottom shows various application icons.

```
Conn.java
Ip.java
case 11: System.out.println("Enter the date range to retrieve the employee expense");
System.out.println("Enter the starting date");
String sdate=r.readLine();
System.out.println("Enter the ending date");
String edate=r.readLine();
System.out.println("Employee_SSN"+ "\t" +"Total expense");
ResultSet rs9=st.executeQuery("select P_SSN,sum(E_amount) from expense where E_date between '"+sdate+"' and '"+edate+"' group by P_SSN order by P_SSN");
while(rs9.next()) {
    System.out.println(rs9.getString(1)+"\t\t"+rs9.getString(2));
}
break;
case 12: System.out.println("Displaying Client SSNs..");
ResultSet rs10=st.executeQuery("select P_SSN from cares");
while(rs10.next()) {
    System.out.println(rs10.getString(1));
}
System.out.println("Enter the Client SSN to retrieve the Volunteer details..");
int rsn=Integer.parseInt(r.readLine());
ResultSet rs11=st.executeQuery("select P_SSN,P_Name from Person where P_SSN in (select P_SSN from serves where T_name in (select T_name from categories where T_SSN='"+rsn+"'))");
System.out.println("PSSN"+ "\t" +"Name");

```

Console

```
<terminated> [p [Java Application] C:\Program Files\Java\jre1.8.0_144\bin\javaw.exe (Nov 19, 2017, 4:44:37 PM)
One Record inserted in a file successfully!
One Record inserted in a file successfully...
File exported successfully...
Enter your option....
20
Exiting....
```