Lab 2 – Week 5

This Lab is based on lecture 5 (chapters 14).

○ Submit your *own work* on time. No credit will be given if the lab is submitted after the due date.
○ Note that the completed lab should be submitted in .doc, .docx, .rtf or .pdf format only.
○ If you think that your answer needs more explanation to get credit then please write it down.

Consider a relation with following attributes:

EmpNo: Employee Number
EmpName   : Employee Name
EmpEmail   : Employee Email
ProjNo : Project Number
ProjName   : Project Name
EmpGrade   : Employee Grade
HrlyRate   : Hourly rate of compensation
                Employees of the same grade receive the same hourly compensation
HrsWorked   : Hours a particular employee worked on a particular project

1. Create this table and sample data in SQL Server. There must be at least 10 rows. There must be 3 to 6 Employees and 3 to 6 projects. You need to add the screenshot of the table showing all the rows.

empNo	empName	empEmail	projNo	projName	empGrade	hrlyRate	hrlyWorked
1	Jimmy	jp@gmail.com	1	proj1	A+	40	40
2	Mar	m@gmail.com	1	proj1	B	15	34
3	Lucas	l@gmail.com	2	proj2	A-	20	50
4	Nena	n@gmail.com	2	proj2	A	50	40
5	Hector	h@gmail.com	3	proj3	A+	35	40
6	Emma	e@gmail.com	3	proj3	B+	30	40
7	Alejo	a@gmail.com	4	proj4	A+	40	40
8	Carlos	c@gmail.com	4	proj4	B+	35	55
9	Andre	an@gmail.com	5	proj5	A+	40	40
10	Len	l@gmail.com	5	proj5	A-	45	50

2. Find all functional dependencies.
   ANS:

   empNo => empName, empEmail, empGrade, hrlyRate

   empEmail => empName, empNo, empGrade, hrlyRate

projNo => projName

empNo, prjNo => hrlyWorked

3. Find all Candidate Keys.
   ANS:

   empNo, projNo

   empEmail, projNo

4. Find a Primary Key.
   ANS:

   empNo, projNo

5. Find all partial dependencies.
   ANS:

   empNo => empName, empEmail, empGrade, hrlyRate

   empEmail => empName, empNo, empGrade, hrlyRate

   projNo => projName

6. Normalize to 2NF.
   ANS:

   Table: Employee

   Columns: empNo, empName, empEmail, empGrade, hrlyRate

   Table: Project

   Columns: projNo, projName

   Table: Employee_project

   Columns: projNo, empNo, hrlyWorked

7. Show new tables after 2NF (based on the sample data you created in 1 above). Screenshots of all the tables are required.

empNo	empName	empEmail	empGrade	hrlyRate
1	Jimmy	jp@gmail.com	A+	40
2	Mar	m@gmail.com	B	15
3	Lucas	l@gmail.com	A-	20
4	Nena	n@gmail.com	A	50
5	Hector	h@gmail.com	A+	35
6	Emma	e@gmail.com	B+	30
7	Alejo	a@gmail.com	A+	40
8	Carlos	c@gmail.com	B+	35
9	Andre	an@gmail.com	A+	40
10	Len	l@gmail.com	A-	45

projNo	projName
1	proj1
1	proj1
2	proj2
2	proj2
3	proj3
3	proj3
4	proj4
4	proj4
5	proj5
5	proj5

empNo	projNo	hrlyWorked
1	1	40
2	1	34
3	2	50
4	2	40
5	3	40
6	3	40
7	4	40
8	4	55
9	5	40
10	5	50

8. Normalize to 3NF.
   ANS:

   empGrade => hrlyRate

   Table: Grade

   Columns: empGrade, hrlyRate

   Table: Employee

   empNo, empName, empEmail, empGrade

   Table: Project

   Columns: projNo, projName

   Table: Employee_project

Columns: projNo, empNo, HrlyWorked

9. Show new tables after 3NF (based on the sample data you created in 1 above). Screenshots of all the tables are required.

empGrade	hrlyRate
A+	40
B	15
A-	20
A	50
A+	35
B+	30
A+	40
B+	35
A+	40
A-	45

empNo	empName	empEmail	empGrade
1	Jimmy	jp@gmail.com	A+
2	Mar	m@gmail.com	B
3	Lucas	l@gmail.com	A-
4	Nena	n@gmail.com	A
5	Hector	h@gmail.com	A+
6	Emma	e@gmail.com	B+
7	Alejo	a@gmail.com	A+
8	Carlos	c@gmail.com	B+
9	Andre	an@gmail.com	A+
10	Len	l@gmail.com	A-

projNo	projName
1	proj1
1	proj1
2	proj2
2	proj2
3	proj3
3	proj3
4	proj4
4	proj4
5	proj5
5	proj5

projNo	empNo	hrlyWorked
1	1	40
1	2	34
2	3	50
2	4	40
3	5	40
3	6	40
4	7	40
4	8	55
5	9	40
5	10	50