CSE462 – Project 2 - Database Concepts – Project 2 – Relational Design and ER model

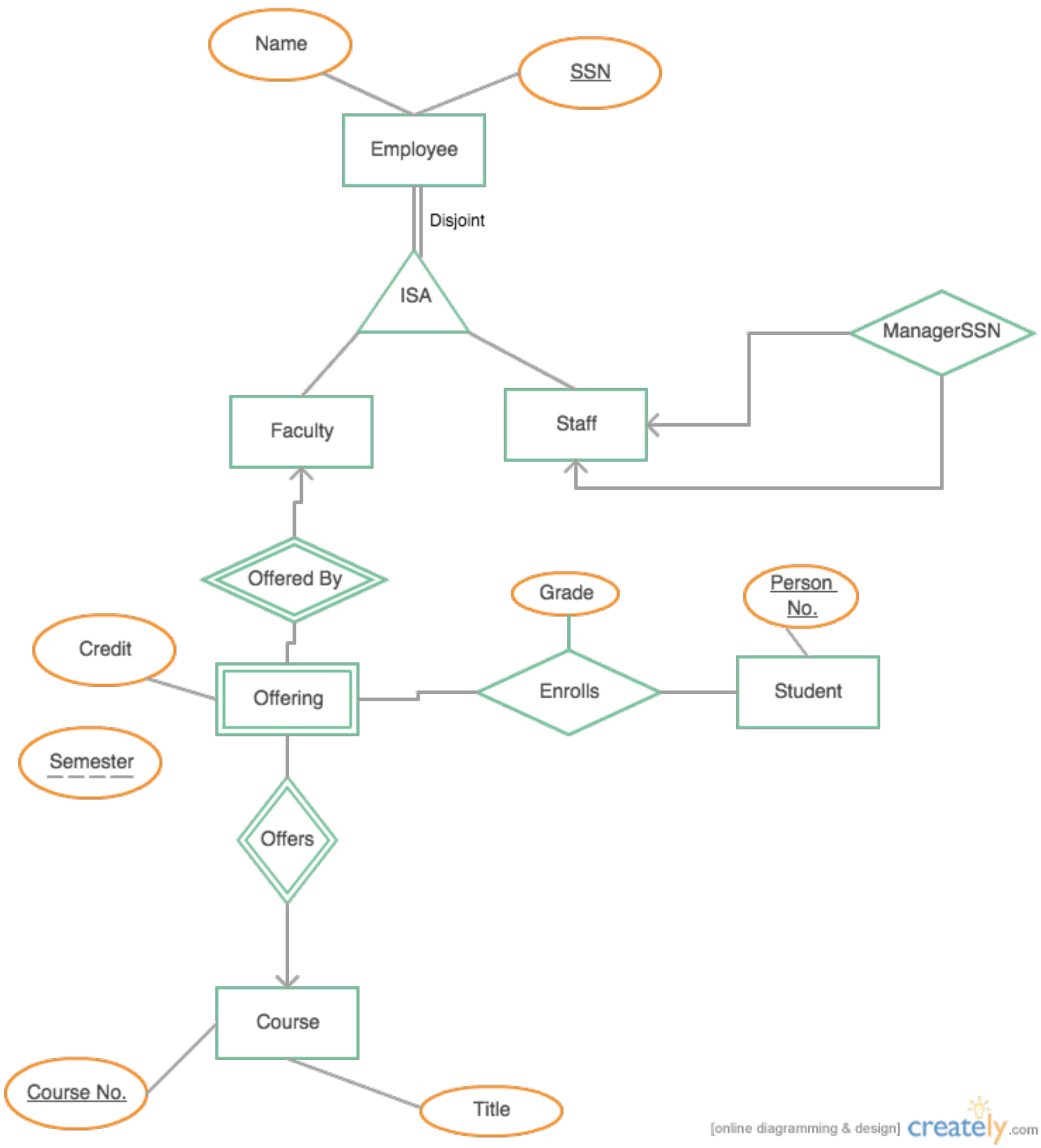
Name – Amey Sanjay Mahajan Person Number – 50090016

UBIT Name - ameysanj

Answer 1 – In the ER diagram, following are the assumptions

- Every Faculty is an employee. Similarly, every staff member is an employee
- Also every employee entity can either be a staff or a faculty member, because of which we have a complete participation as well as disjoint in the diagram.
- One faculty can offer only one offering of same course number and same semester whereas, one offering can be offered by many Faculty members; so we have a many-to-one identifying relationship between offering and faculty.
- Also Enrolls is a strong relationship between Students and Offering. It is a many-to-many relationship because one student can be enrolled in many offerings and one offering can have many students enrolled in it.
- We also have an identifying relationship Offers in the diagram which connects Offering and Course. It is a many-to-one from Offering to Course as one course can have multiple offerings but one offering with unique course number and semester results in only one instance of that specific course.

Diagram on the following page-


Name
SSN
Employee
Disjoint
ISA
ManagerSSN
Faculty
Staff
Offered By
Grade
Person No.
Credit
Offering
Enrolls
Student
Semester
Offers
Course
Course No.
Title
[online diagramming & design] creately.com

Answer 2 –

1. Representing the mentioned data in single R-

➔ R(Fac_SSN , Fac_Name , S_PNo , S_Name , Course_No , Course_Title , Semester , Credit , Grade)

Key - Fac_SSN, S_PNo,Course_No,Semester

➔ This can be proved as following –
{Fac_SSN, S_PNo,Course_No,Semester}$^+$ = { Fac_SSN , Fac_Name , S_PNo , S_Name , Course_No , Course_Title , Semester , Credit , Grade }

The Closure of Fac_SSN, S_PNo,Course_No,Semester gives us all the attributes in R. Hence it is a superkey.
Now, all the subsets of Fac_SSN, S_PNo,Course_No,Semester do not give us all the attributes.
Thus, both the conditions to be a key are satisfied.

Functional Dependencies

➔ Fac_SSN -> Fac_Name
➔ S_PNo -> S_Name
➔ S_PNo Course_No -> Grade
➔ Course_No -> Course_Title
➔ Course_No -> Credit
➔ Course_Title Semester ->Course_Number
➔ Semester Course_Title ->Credit

2. As we see, as none of the dependencies have individual superkeys on the LHS, the resulting schema R is **not** in BCNF.

3. Lossless Decomposition of R into – { Fac_SSN, S_PNo ,Semester,Course_Title}, {Fac_SSN , Fac_Name},{S_PNo,S_Name}, {S_PNo,Course_No, Grade},{Course_Number,Course_Title,Credit}, {Semester,Course_Title,Course_Number}

This can be proved by Chase-

Fac_SSN	Fac_Name	S_PNo	S_Name	Course_No	Course_Title	Semester	Credit	Grade
fs	fn1	sp	sn1	cn1	ct	s	c1	g1
fs	fn	sp1	sn2	cn2	ct1	s1	c2	g2
fs1	fn2	sp	sn	cn3	ct2	s2	c3	g3
fs2	fn3	sp	sn3	cn	ct3	s3	c4	g
fs3	fn4	sp2	sn4	cn	ct	s4	c	g4
fs4	fn5	sp3	sn5	cn4	ct4	s	c5	g5

After Applying all the dependencies mentioned above, we get the following table –

Fac_SSN	Fac_Name	S_PNo	S_Name	Course_No	Course_Title	Semester	Credit	Grade
fs	fn	sp	sn	cn	ct	s	c	g
fs	fn	sp1	sn2	cn2	ct1	s1	c2	g2
fs1	fn2	sp	sn	cn3	ct2	s2	c3	g3
fs2	fn3	sp	sn3	cn	ct3	s3	c4	g
fs3	fn4	sp2	sn4	cn	ct	s4	c	g4
fs4	fn5	sp3	sn5	cn4	ct4	s	c5	g5

The highlighted tuple, justifies that the decomposition is lossless and it preserves all the dependencies.

Answer 3 – Extra Credit-

1. If A->->B and A->->C then A->->BC

Answer - 'True' - justified by chase –

A	B	C	D
a	b1	c1	d1
a	b2	c2	d2

Tableau setup for A->->BC

A	B	C	D
a	b2	c1	d1
a	b1	c2	d2

Applying A->->B

A	B	C	D
a	b2	c2	d1
a	b1	c1	d2

Applying A->->C

Both the tuples we get after applying A->->C justifies that A->->BC holds, as per the definition mentioned.

2. If A->->BC, then A->->B

Answer – 'False' – justified by chase –

A	B	C	D
a	b1	c1	d1
a	b2	c2	d2

Tableau setup for A->->B

A	B	C	D
a	b2	c2	d1
a	b1	c1	d2

Applying A->->BC

As there are no other dependencies left to apply and we don't obtain tuples which satisfy the dependency A->->B, the answer is False.

3. If AB->C, then A->C

Answer – ‘False’ – justified by chase –

A	B	C	D
a	b1	c1	d1
a	b2	c2	d2

Tableau Setup for A->C

For Chase algorithm to start, it is necessary that LHS of dependencies have same values. Here to apply AB->C, we need values of A and B to be same in both tuples; but we see though A has same values, B has different. Hence we cannot achieve common/same values in C. Therefore, it is ‘False’.