COMP 3005 Assignment 3
Fall 2018
Abdul-Malik Marikar 101042166

1-1: p[name,areacode,officecode,stationcode](subscribers x lines)

1-2: p[name,address,portid](subscribers x (s(s.code= “Call Forward Busy”,service= “CFB”)(services)))

1-3: p[portid](trunks x trunk_channels(s(state != “IDLE”)))

1-4: p[call_id](trunks x s[orig = portid or term =portid](trunks))

1-5: p[portid](lines x s[orig = portid](calls) x s[tcode = “BUSY”](treatments))

1-6: p[portid](trunks* [state = “IDLE”](trunk_channels))

1-7:

2-1: Candidate keys- U,Z U,V
    1st Normal form: Satisfied (the all-attributes relation has a key)
    2nd Normal form: Satisfied (no non-prime attribute is functionally determined by
        the proper subset of a key)
    3rd Normal form: not satisfied because in dependency V,W->Y LHS is not a superkey AND
        RHS does not consist of only prime attributes
    Boyce-Codd Normal form: Not Satisfied because 3rd Normal Form isn't.

Loss-less join decomposition:This collection of tables preserves all attributes and functional dependencies and can be joined without loss into a single relation consisting of all the attributes.

[Primary Key Attributes | Non-Primary Key Attributes]

[U,V | W,X,Z]
[V,W | Y]
[Z | V]

Removing Redundant (Subsumed) table: [Z | V]

With Redundant (subsumed) Tables Removed:

[U,V | W,X,Z]
[V,W | Y]

2-2: The following set of attributes forms one possible candidate (minimal) key for a universal table consisting of all attributes with respect to the functional dependencies: U,V

1st Normal form: Satisfied (the all-attributes relation has a key)

2nd Normal form: Satisfied (no non-prime attribute is functionally determined by the proper subset of a key)

3rd Normal form: Not satisfied: because in dependency X->W LHS is not a superkey AND RHS does not consist of only prime attributes and because in dependency W->Z LHS is not a superkey AND RHS does not consist of only prime attributes

Boyce-Codd Normal form: Not Satisfied because 3rd Normal Form isn't.

Loss-less join decomposition:This collection of tables preserves all attributes and functional dependencies and can be joined without loss into a single relation consisting of all the attributes.

[Primary Key Attributes | Non-Primary Key Attributes]

[U,V | X,Y]
[X | W]
[W | Z]

2-3: Candidate key: V

1st Normal form: Satisfied (the all-attributes relation has a key)

2nd Normal form: Satisfied (no non-prime attribute is functionally determined by the proper subset of a key)

3rd Normal form: Not Satisfied because in dependency X->Z LHS is not a superkey AND RHS does not consist of only prime attributes

Boyce-Codd Normal form: Not Satisfied because 3rd Normal Form isn't.

Loss-less join decomposition:This collection of tables preserves all attributes and functional dependencies and can be joined without loss into a single relation consisting of all the attributes.

[Primary Key Attributes | Non-Primary Key Attributes]

[V | W,X,U,Y]
[X | Z]

2-4:

A) F1= A,B➝C,D,E,F,G
A➝C
B➝D,E,F
E➝F

AB➝BC (Argumentation)
AB➝BCG (union)

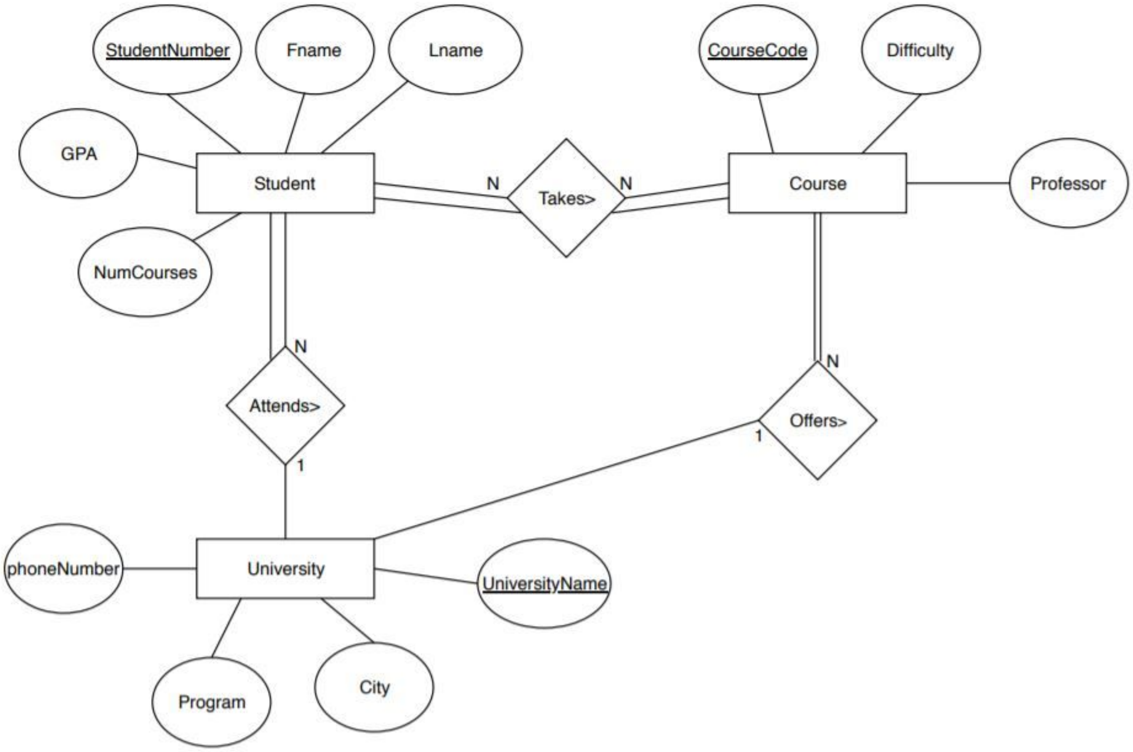
AB➝A,D,E,F (Argumentation)
AB➝C,D,E,F,G (Decomposition)

B)F2= A,B➝G
A➝C
B➝D,E,F
E➝F

AB➝C,G (transitive A➝C)
AB➝C,D,E,F,G (transitive B➝D,E,F)

F2: AB➝C,D,E,F,G = F1: AB➝C,D,E,F,G

3-2:



http://localhost:3000/api/student this url should show a list of all the students in the Database

http://localhost:3000/api/student/fname?=Abdul this will return a list of all students with the first name Abdul

http://localhost:3000/api/university this url should return a list of universities in the database.

http://localhost:3000/api/Course this url should return a list of all Courses offered.

http://localhost:3000/api/University/city?=Ottawa this url should return a list of all universities in Ottawa

http://localhost:3000/api/Course/Professor this url will return a list of all professors in the database