Noémi Lemonnier
#40001085

ASSIGNMENT 3

QUESTION 1

b)

FD's would be {A -> B, B -> C, C -> D, D -> A}

c)

1) Closure:

AB+= {A, B}
A+={A, B} it means A→B
B+={B, A} it means B→A

2) Answer

FD's could be {A→B, B→A, C→ABD, D→ABC}

QUESTION 2

R(A,B,C,D,E) project on FD S(A,B,C)

c) AB→D, AC→E, BC→D, D→A, E→B

1) Closure

A+={A}
B+={B}
C+={C}
AB+={A,B,D}
AC+={A,C,E,B,D} candidate key
CB+={C,B,D,A,E} candidate key

2) FD's

~~AC→E~~	~~CB→D~~
AC→B	CD→ A
~~AC→D~~	~~CD→E~~

3) Answer

{AC→B, CD→ A }

d) A→B, B→C, C→D, D→E, E→A

1) Closure

A+={A,B,C,D,E} candidate key
B+={B,C,D,E,A} candidate key
C+={C,D,E,A}
AB+={A,B,C,D,E} candidate key
AC+={A,C,B,D,E} candidate key
BC+={B,C,D,E,A} candidate key

2) FD's

A→B	AC→B
~~A→C~~	AB→ C
B→C	BC→A

3) Answer

{A→B, B→C, AC→B, AB→ C, BC→A}

Noémi Lemonnier
#40001085

QUESTION 3

g)

R1:= ρ_{PC1}(PC)
R2:= ρ_{PC2}(PC)
R3:= R1 $\bowtie_{(PC1.speed=PC2.speed \text{ AND } PC1.ram=PC2.ram) \text{ AND } PC1.model< PC2.model}$ R2
R4:= $\pi_{PC1.model,PC2.model}$(R3)

i)

R1:= $\pi_{model,\ speed}$(PC)
R2:= $\pi_{model,\ speed}$(Laptop)
R3:=R1 $\cup$ R2
R4:= $\rho_{R4(model2,speed2)}$(R3)
R5:= $\pi_{model,\ speed}$(R3 $\bowtie_{(speed<speed2)}$ R4)
R6:= R3 − R5
R7:= π_{maker} (R6 $\bowtie$ Product)

j)

R1:= $\pi_{maker,\ speed}$ (Product $\bowtie$ PC)
R2:= $\rho_{R2(maker2,\ speed2)}$(R1)
R3:= $\rho_{R3(maker3,\ speed3)}$(R1)
R4:=R1 $\bowtie_{(maker=maker2 \text{ AND } speed<>speed2)}$ R2
R5:=R4 $\bowtie_{(maker=maker3 \text{ AND } speed<>speed3 \text{ AND } speed3<>speed2)}$ R3
R6:= π_{maker} (R 5)

k)

R1:= $\pi_{maker\ ,model}$ (Product $\bowtie$ PC)
R2:= $\rho_{R2(maker2,\ model2)(}$R1)
R3:= $\rho_{R3(maker3,\ model3)(}$R1)
R4:= $\rho_{R4(maker4,\ model4)(}$R1)
R5:= R1 $\bowtie_{(maker=maker2 \textbf{ AND } model<>model2)}$ R2
R6:= R3$\bowtie_{(maker=maker3 \textbf{ AND } model3<>model2 \textbf{ AND } model3<>model)}$ R5
R7:= R4$\bowtie_{(maker=maker4 \textbf{ AND } (model4=model \textbf{ OR } model4=model2 \text{ OR } model4=model3))}$R6
R8:=π_{maker} (R 7)

QUESTION 4

a)

Please see attached document "FindingKeyCandidate.xls"

The candidate keys are: ICEG, GCEI, IEGH, IBEG

b)

No because of CD→ A and GHB→ AB. CD and GHB are not part of any superkey and same for A and AB.

Noémi Lemonnier
#40001085

c)

1) put FD's in simple forms

F= {CD→A, EC→H, GHB→AB, C→D, EG→A, H→B, BE → CD, EC → B}

Decomposition to break not singleton FD's RHS:

F= {CD→A, EC→H, GHB→A, GHB→B, C→D, EG→A, H→B, BE → C, BE → D, EC → B}

2) Remove redundancy in the FDs

CD→ A where C^+={C,D} and D^+={D} so eliminate D and get C→A

EC→H where C^+={C,D} and E^+={E} <u>no redundancy</u>;

GHB→A where G^+={G} and H^+={HB} and B^+={B} so eliminate B and get GH→A

GHB→B where G^+={G} and H^+={HB} and B^+={B} is redundant so we can remove it because of H→B

C→D <u>no redundancy</u>;

EG→A where E^+={E} and G^+={G} <u>no redundancy</u>;

H→B <u>no redundancy</u>;

BE→ C where B^+={B} and E^+={E} <u>no redundancy</u>;

BE → D where B^+={B} and E^+={E} <u>no redundancy</u>;

EC → B where E^+={E} and C^+={C,D} <u>no redundancy</u>;

F= {C→A, EC→H, GH→A, C→D, EG→A, H→B, BE → C, BE → D, EC → B}

3) Remove redundancy of FDs

C→ A where C^+={C,D} so <u>no redundancy</u>;

EC→H where EC^+={C,D,A,E,B} so <u>no redundancy</u>;

GH→A where GH^+={G,H,B} so <u>no redundancy</u>;

C→D <u>no redundancy</u>;

EG→A where EG^+={E,G} so <u>no redundancy</u>;

H→B <u>no redundancy</u>;

BE→ C where BE^+={B,E,D} so <u>no redundancy</u>;

BE→ D where BE^+={B,E,C} so <u>no redundancy</u>;

EC → B where EC^+={E,C,D,A,H,B} so this is redundant and we can remove it.

F= {C→A, EC→H, GH→A, C→D, EG→A, H→B, BE → C, BE → D}

d)

R1(A,B,C) A^+={A} B^+={B} C^+={C,D,A}

R2(C,D,E) C^+={C,D,A} D^+={D} E^+={E}

R3(C, G,H,I) C^+={C,D,A} G^+={G} H^+={H,B} I^+={I}

R1∪R2∪R3 provides all FDs and it is lossless as R1∩R2 = {C,D,A} R2∩R3 = {C,D,A} R1∩R3 = {C,D,A,B} which is FD's CD→A

Noémi Lemonnier
#40001085

e)

1) Closure of attributes

A^+= A
B^+= B
C^+= C, D, A
D^+= D
AB^+= A, B
AC^+= A, C, D
AD^+=A, D
BC^+=B, C, D, A
BD^+=B, D
DC^+=D, C, A

2) Projection

A→B	B→D
A→C	~~C→D~~
A→D	AB→C
B→C	AB→D
AC→B	~~AC→D~~
AD→B	AD→C
~~BC→A~~	BD→A
~~BC→D~~	BD→C
~~CD→A~~	CD→B

3) Answer

{A→B, A→C, A→D, B→C, B→D, AB→C, AB→D, AC→B, B→C, AC→B, AD→B, AD→C, BD→A, BD→C, CD→B }

QUESTION 5

a)

A	B	C
X	candidate key	X

So any key should have B
<u>*BNCF*</u>: No because no superkey
<u>*3NF*</u>: No because C→A 1)C is not a candidate key or superkey. or 2) A should be a subset of B (in their closure)
<u>*2NF*</u>: AB→C partial dependency
<u>*1NF*</u>: YES.

b)

1) find candidate keys

A	B	C
X	candidate key	X

It is not a BCNF because it is not a 3NF and because of C→A and A→C which makes the LHS not a superkey.

1) To make it 3NF:
S1(A,B) FD's A→B, B→A
S2(B,C) FD's B→C, C→A

2) To make it BCNF
S1(A, B) FD's A→B, B→A
S3(B, C) FD's B→C , B→A

c) I think it is better to decompose it in BNCF as it is easier to use them and see that it is lossless and preserved functional dependency.

Noémi Lemonnier
#40001085

QUESTION 6

a)

1) put FD's in simple forms

F= {Ship→Capacity, ShipDate→Cargo, CargoCapacity→Value}

Nothing to decompose as all FD's RHS are singleton

2) Remove redundancy in the FDs

Ship→ Capacity no redundancy;

ShipDate→Cargo where $Ship^{+}$={Ship, Capacity} $Date^{+}$={Date} no redundancy;

CargoCapacity→Value where $Cargo^{+}$={Cargo} $Capacity^{+}$={Capacity} no redundancy;

3) Remove redundancy of FDs

Ship→ Capacity no redundancy;

ShipDate→Cargo where $ShipDate^{+}$={Ship, Capacity, Date} no redundancy

CargoCapacity→Value where $CargoCapacity^{+}$={Cargo, Capacity} no redundancy

F= {Ship→Capacity, ShipDate→Cargo, CargoCapacity→Value}

b)

R1(Ship, Capacity) $Ship^{+}$={Ship, Capacity} $Capacity^{+}$={Capacity}

R2(Ship, Date, Cargo, Value) $Ship^{+}$={Ship, Capacity} $Date^{+}$={Date} $Cargo^{+}$={Cargo} $Value^{+}$={Value}

R1∪R2 provides all FDs and it is lossless as R1∩R2 = {Ship, Capacity} which is FD's Ship→ Capacity.

c)

R1(Cargo, Capacity, Value) $Cargo^{+}$={Cargo} $Capacity^{+}$={Capacity} $Value^{+}$={Value}

R2(Ship, Capacity) $Ship^{+}$={Ship, Capacity} $Capacity^{+}$={Capacity}

R2(Ship, Date, Cargo) $Ship^{+}$={Ship, Capacity} $Date^{+}$={Date} $Cargo^{+}$={Cargo}

R1∪R2∪R3 provides all FDs and it is not lossless as R1∩R2 = {Capacity} R2∩R3={Ship} R1∩R3 = {Cargo} as Ship→ Capacity but none gives the rest