

Week 6 Normalisation



Help

http://en.wikipedia.org/wiki/Ursus_Wehrli



Housekeeping

Assignment Project Exam Help

- The mark and feedback will be released on 13 Sep 2022.
- A tailored database will be designed to reveal common issues of incorrect queries and made ivaliable to you as part of the feedback.

 More drop-in sessions will be available after 14 Sep if you need any
- More prop-in sessions will be available after 14 Sep if you need any clarification on the marking of Assignment 1.



Housekeeping

Assignment Project Exam Help

- The mark and feedback will be released on 13 Sep 2022.
- A tailored database will be designed to reveal common issues of pircon extraveles and made tvallable to you aspart of the feedback.

 More drop-in sessions will be available after 14 Sep if you need any
- More drop-in sessions will be available after 14 Sep if you need any clarification on the marking of Assignment 1.
- Thanks for participating in the anonymous survey in Week 5 and more information will be available on Wattle during the teaching break.
- An optional exercise website for our course

https://cs.anu.edu.au/dab/bench/db-exercises/



Housekeeping

Assignment Project Exam Help

- The mark and feedback will be released on 13 Sep 2022.
- A tailored database will be designed to reveal common issues of incomes for the feedback.
- More drop-in sessions will be available after 14 Sep if you need any clarification on the marking of Assignment 1.
- Thanks for participating in the anonymous survey in Week 5 and more information will be available on Wettle during the teaching break.
- An optional exercise website for our course

https://cs.anu.edu.au/dab/bench/db-exercises/

Enjoy the semester break!





https://tutorcs.com



Assignment Project Exam Help more problems than it solves!

- This terpess constitution portans questions
 - Do we need to decompose a relation?



Assignment Project Exam Help more problems than it solves!

- This terpes constitution to tan our storm
 - Do we need to decompose a relation?

Several normal forms

Chell and decide whether the to Cess mpose a relation



Assignment Project Exam Help more problems than it solves!

- This terpess constitution portans questions
 - Do we need to decompose a relation?
 - Several normal forms

 Chell at the cide whether hard of Cess mpose a relation
 - What problem (if any) does a given decomposition cause?



Assignment Project Exam Help more problems than it solves!

- This terpes constitution to tan question m
 - Do we need to decompose a relation?

Several normal forms

Check the latest of th

- 2 What problem (if any) does a given decomposition cause?
 - Two properties
 - → help us to decide how to decompose a relation



Two Properties

Assignment Project Exam Help In addition to data redundancy, we need to consider the following

In addition to data redundancy, we need to consider the following properties when decomposing a relation:

https://tutorcs.com



Two Properties

Assignment Project Exam Help In addition to data redundancy, we need to consider the following

properties when decomposing a relation:

https://taluelesage.atom

To disallow the possibility of generating spurious tuples when a NATURAL JOIN operation is applied to the relations after decomposition.

WeChat: cstutorcs



Two Properties

Assignment Project Exam Help In addition to data redundancy, we need to consider the following

 In addition to data redundancy, we'need to consider the following properties when decomposing a relation:

https://tuteress.com

To disallow the possibility of generating spurious tuples when a NATURAL_JOIN operation is applied to the relations after decomposition.



To ensure that each functional dependency can be inferred from functional dependencies after decomposition.

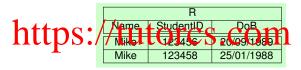


Assignmental possibility of generating spurious tupies when a NATURAL possibility of generating spurious tupies and spurious tupies

https://tutorcs.com



Assistance in the same data of the same

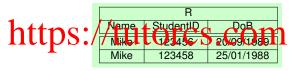


W	Cane	hat:	cst	utor (SDOB
	Mike	123456		123456	20/09/1989
	Mike	123458		123458	25/01/1988

• **Example 1:** Does the decomposition of *R* into *R*₁ and *R*₂ has the lossless join property?



Assistance in the same data of the same



W	Cane	hati	cst	utor (SDOB
	Mike	123456		123456	20/09/1989
	Mike	123458		123458	25/01/1988

• **Example 1:** Does the decomposition of *R* into *R*₁ and *R*₂ has the lossless join property?

Yes, because the natural join of R_1 and R_2 yields R.



A SST cisalled the cossibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and analysis of possibility of general his surfound tuples and t

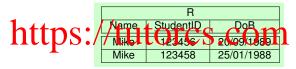




• **Example 2:** Does the decomposition of *R* into *R*₃ and *R*₄ has the lossless join property?



As ST gisallow that possibility of general his sturbus tuples and almatural purpose of the possibility of general his sturbus tuples and almatural purpose of the relations after decomposition.





• **Example 2:** Does the decomposition of *R* into *R*₃ and *R*₄ has the lossless join property?

No, because the natural join of R_3 and R_4 generates spurious tuples.



A SSEkanth Photodoling Decompositor Gram First Randa deshelp have the lossless join property. It generates spurious tuples.

1.				//						
m	ĺΊ	n	S R	7	Ù	\mathbf{I}	Ī		Ϊ(
Nam	ie	Stu	identil	<u> 2</u>		Do	ρB		ľ	
Mike	е	1:	23456		20	/09	/19	989		
Mike	е	1:	23458		25	/01	/19	988		
	_	_	\sim							

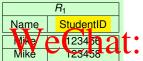
	SELECT * FROM R ₃ NATURAL JOIN R ₄							
S	Name	StudentID	DoB					
$\tilde{}$	Mike	123456	20/09/1989					
	Mike	123456	25/01/1988					
	Mike	123458	20/09/1989					
ļ	∠ Mike	123458	25/01/1988					

R ₃						
Name	StudentID					
Mike	123456					
Mike	123458					

R_4						
Name	DoB					
Mike	20/09/1989					
Mike	25/01/1988					



Assignine interpretient Exam Help



R ₃						
Name	StudentID					
Mike	123456					
Mike	123458					

R_2					
StudentID DoB					
(123 F56)	20/09/1989				
123458	25/01/1988				

Lossless join

H ₄					
Name	DoB				
Mike	20/09/1989				
Mike	25/01/1988				

Not lossless join



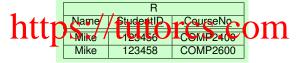
As Schregelinferred from unctional dependencies after descriposition. Telp

https://tutorcs.com



As Scarge interior in the citing that each functional dependency preservation. To ensure that each functional dependency preservation is the composition.

• Example 1: Given a FD $\{StudentID\} \rightarrow \{Name\}\ defined\ on\ R$





■ Does the above decomposition preserves {StudentID} → {Name}?



As Scarge inferred from Junctional dependencies after descriptional dependency p

• Example 1: Given a FD {StudentID} → {Name} defined on R

_		, R				
httr	Name/	StudentID	CeurseNo	om		
11111	Mike	123456	COMP2409	OIII		
1	Mike	123458	COMP2600			

_	•	\sim		
\/		R_1	าว	1
 W	lame	St	udent	
	Mike	1	2345	6
	Mike	1	2345	3

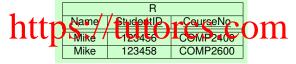
~~	4	4-0-	Poo C
CS	CO to c		Cou seNo
	123456		COMP2400
	123458		COMP2600

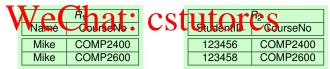
• Does the above decomposition preserves $\{StudentID\} \rightarrow \{Name\}$? Yes, because $\{StudentID\}$ and $\{Name\}$ are both in R_1 after decomposition and thus $\{StudentID\} \rightarrow \{Name\}$ is preserved in R_1 .



As Scarge interior translation in the control of the composition of th

• Example 2: Given a FD $\{StudentID\} \rightarrow \{Name\}\ defined\ on\ R$



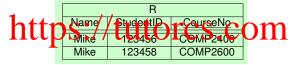


■ Does the above decomposition preserves {StudentID} → {Name}?



A S Scan beliefered from Lunctional dependencies after descriptional dependency preservation. The position of the composition o

• Example 2: Given a FD $\{StudentID\} \rightarrow \{Name\}\ defined\ on\ R$





Does the above decomposition preserves {StudentID} → {Name}?
 No, because {StudentID} and {Name} are not in a same relation after decomposition.



A S Schrebeling rich unctional dependency preservation. To ensure that each functional dependency preservation. The properties after descomposition.

• **Example 3:** Given a set of FDs $\{ \{ \text{StudentID} \} \rightarrow \{ \text{Email} \}, \{ \text{Email} \} \rightarrow \{ \text{Name} \}, \{ \text{StudentID} \} \rightarrow \{ \text{Name} \} \}$ defined on R

1_44_	~ . /	144-	R
nu	Name	St ic elit! D	
1	Mike	123456	123456@anu.edu.au
	Tom	123123	123123@anu.edu.au

	T					_
	V			n	21	•
Nam	e 🖊		Ema	til		
Mike		123456				
Tom		123123	3@ar	ıu.e	edu.a	เน

S		It	\bigcap	R		2						
Stu	identID		StudentID					En	nai			
12	23456		1:	234	560	@:	anı	ı.ec	du.a	au		
12	123123			231	230	@	anı	ı.ec	du.a	au		



A S Schrebeling rich unctional dependency preservation. To ensure that each functional dependency preservation. The properties after descomposition.

• **Example 3:** Given a set of FDs $\{ \{ \text{StudentID} \} \rightarrow \{ \text{Email} \}, \{ \text{Email} \} \rightarrow \{ \text{Name} \}, \{ \text{StudentID} \} \rightarrow \{ \text{Name} \} \}$ defined on R

1_44	~ ~ /	144-	R
nu	Name	Student D	CS Enail C
	Mike	123456	123456@anu.edu.au
	Tom	123123	123123@anu.edu.au

	V	A R		h	12	ī	- •
Name	V		Em	ail	Ю	7	•
Mike	1	23456	6@a	nu.	edu	.a	u
Tom	1	23123	3@a	nu.	edu	.a	u

S	uto		OFC Smail
Oil	Studentib		
12	123456		123456@anu.edu.au
12	123123		123123@anu.edu.au

■ Does the above decomposition preserves {StudentID} → {Name}?



A S Schredelight right troth functional dependency preservation. To ensure that each functional dependency preservation. To ensure that each functional dependency preservation.

• Example 3: Given a set of FDs $\{ \{ \text{StudentID} \} \rightarrow \{ \text{Email} \}, \{ \text{Email} \} \rightarrow \{ \text{Name} \}, \{ \text{StudentID} \} \rightarrow \{ \text{Name} \} \}$ defined on R

1_44	~ ~ /	144-	R
nu	Name	Student D	CS Enail C
	Mike	123456	123456@anu.edu.au
	Tom	123123	123123@anu.edu.au

T	T	<i>T</i>		1		4
	N			n	เล	T.
Name	\		Em	ail		
Mike	1	23456	6@a	nu.	edu.	au
Tom	1	23123	3@a	nu.	edu.	au

Stu	der			OFC S Email
12	234	23456		123456@anu.edu.au
12	23123			123123@anu.edu.au

• Does the above decomposition preserves {StudentID} \rightarrow {Name}? Yes, because {StudentID} \rightarrow {Name} can be inferred by {StudentID} \rightarrow {Email} (preserved in R_2) and {Email} \rightarrow {Name} (preserved in R_1).



Assignment Bergiect, Examithelp

https://tutorcs.com



Assignment Bernject, Examithelp

• Lossless join if and only if the common attributes of R_1 and R_2 are a superkey for R_1 or R_2 ;

https://tutorcs.com



Assignment Bernject, Examithelp

• Lossless join if and only if the common attributes of R_1 and R_2 are a superkey for R_1 or R_2 ;

 $\begin{array}{ll} \textbf{herendency overerying} \text{ if and only if } (\Sigma, \cup \Sigma_s)^* = \Sigma^* \text{ holds.} \\ \textbf{https://tutolcs.com} \end{array}$



- Lossless join if and only if the common attributes of R_1 and R_2 are a superkey for R_1 or R_2 ;
- hereing the period of the property of the period of the p
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \to B, B \to C, A \to C\}$. Does the decompostion of R into $R_1 = \{A, B\}$ and $R_2 = \{A, C\}$ fullfill lossless icid and dependency presenting? TCS



- Lossless join if and only if the common attributes of R_1 and R_2 are a superkey for R_1 or R_2 ;
- herendency breserving if and only if $(\Sigma, \cup \Sigma_s)^* = \Sigma^*$ holds. $(\Sigma, \cup \Sigma_s)^* = \Sigma^*$
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \rightarrow B, B \rightarrow C, A \rightarrow C\}$. Does the decomposition of R into $R_1 = \{A, B\}$ and $R_2 = \{A, C\}$ fullfill los was initial and dependency presenting TCS $\Sigma_1 = \{A \to B\} \text{ and } \Sigma_2 = \{A \to C\}$



- Lossless join if and only if the common attributes of R₁ and R₂ are a superkey for R₁ or R₂;
- $\begin{array}{ll} & \text{ h condency or eserving if and only if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if and only if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if and only if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if and only if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if and only if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if and only if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condency or eserving if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ h condenc$
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \to B, B \to C, A \to C\}$. Does the decomposition of R into $R_1 = \{A, B\}$ and $R_2 = \{A, C\}$ fullfill loss less initial and dependency presenting?
 - $\Sigma_1 = \{A \rightarrow B\}$ and $\Sigma_2 = \{A \rightarrow C\}$
 - Lossless join?



- Lossless join if and only if the common attributes of R₁ and R₂ are a superkey for R₁ or R₂;
- $\begin{array}{ll} \textbf{hereodency or eserving} \text{ if and only if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ \textbf{COM} \end{array}$
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \to B, B \to C, A \to C\}$. Does the decompostion of R into $R_1 = \{A, B\}$ and $R_2 = \{A, C\}$ fullfill lossless initial and dependency presenting?
 - $\Sigma_1 = \{A \rightarrow B\}$ and $\Sigma_2 = \{A \rightarrow C\}$
 - Lossless join? Yes because A is a superkey for R_1 .



- Lossless join if and only if the common attributes of R₁ and R₂ are a superkey for R₁ or R₂;
- $\begin{array}{ll} & & \text{hereodency breserving if and only if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{holds.} \end{array}$
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \to B, B \to C, A \to C\}$. Does the decomposition of R into $R_1 = \{A, B\}$ and $R_2 = \{A, C\}$ fullfill loss icin and dependency presenting?
 - $\Sigma_1 = \{A \rightarrow B\}$ and $\Sigma_2 = \{A \rightarrow C\}$
 - Lossless join? Yes because A is a superkey for R₁.
 - Dependency preserving?



- Lossless join if and only if the common attributes of R_1 and R_2 are a superkey for R_1 or R_2 ;
- $\begin{array}{ll} & \text{ The pendency of eserving if and only if } (\Sigma, \cup \Sigma_s)^* = \Sigma^* \text{ holds.} \\ & \text{ COM} \end{array}$
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \to B, B \to C, A \to C\}$. Does the decomposition of R into $R_1 = \{A, B\}$ and $R_2 = \{A, C\}$ fullfill loss in an dependency presenting?
 - $\Sigma_1 = \{A \rightarrow B\}$ and $\Sigma_2 = \{A \rightarrow C\}$
 - Lossless join? Yes because A is a superkey for R_1 .
 - Dependency preserving? No because $(\Sigma_1 \cup \Sigma_2)^* \neq \Sigma^*$ from the fact that $\{A \to B, A \to C\} \not\vDash B \to C$.



- Lossless join if and only if the common attributes of R_1 and R_2 are a superkey for R_1 or R_2 ;
- hereing the period of the property of the period of the p
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \to B, B \to C, A \to C\}$. Does the decomposition of R into $R_1 = \{A, B\}$ and $R_3 = \{B, C\}$ fullfill loss less icin and dependency presenting? TCS



- Lossless join if and only if the common attributes of R_1 and R_2 are a superkey for R_1 or R_2 ;
- hereing the period of the per
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \rightarrow B, B \rightarrow C, A \rightarrow C\}$. Does the decomposition of R into $R_1 = \{A, B\}$ and $R_3 = \{B, C\}$ fullfill los was initial and dependency presenting TCS $\Sigma_1 = \{A \to B\} \text{ and } \Sigma_3 = \{B \to C\}$



- Lossless join if and only if the common attributes of R₁ and R₂ are a superkey for R₁ or R₂;
- $\begin{array}{ll} & \text{ } &$
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \to B, B \to C, A \to C\}$. Does the decomposition of R into $R_1 = \{A, B\}$ and $R_3 = \{B, C\}$ fullfill lossless join and dependency presenting? TCS
 - $\Sigma_1 = \{A \rightarrow B\}$ and $\Sigma_3 = \{B \rightarrow C\}$
 - Lossless join?



- Lossless join if and only if the common attributes of R₁ and R₂ are a superkey for R₁ or R₂;
- $\begin{array}{ll} & \text{ holds.} \\ & \text{ holds.}$
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \to B, B \to C, A \to C\}$. Does the decomposition of R into $R_1 = \{A, B\}$ and $R_3 = \{B, C\}$ fullfill loss icin and dependency present in $\{A, B\}$?
 - $\Sigma_1 = \{A \rightarrow B\}$ and $\Sigma_3 = \{B \rightarrow C\}$
 - Lossless join? Yes because B is a superkey for R_3 .

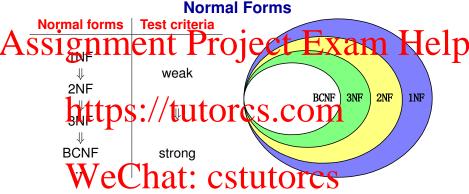


- Lossless join if and only if the common attributes of R₁ and R₂ are a superkey for R₁ or R₂;
- $\begin{array}{ll} & \text{ here products} \text{ if and only if } (\Sigma, \cup \Sigma)^* = \Sigma^* \text{ holds.} \\ & \text{ holds.} \end{array}$
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \to B, B \to C, A \to C\}$. Does the decompostion of R into $R_1 = \{A, B\}$ and $R_3 = \{B, C\}$ fullfill loss so in an dependency presenting?
 - $\Sigma_1 = \{A \rightarrow B\}$ and $\Sigma_3 = \{B \rightarrow C\}$
 - Lossless join? Yes because B is a superkey for R_3 .
 - Dependency preserving?



- Lossless join if and only if the common attributes of R_1 and R_2 are a superkey for R_1 or R_2 ;
- $\begin{array}{ll} & \text{ The pendency of serving if and only if } (\Sigma, \cup \Sigma_s)^* = \Sigma^* \text{ holds.} \\ & \text{ COM} \end{array}$
- Consider R={A, B, C} with the set of FDs $\Sigma = \{A \to B, B \to C, A \to C\}$. Does the decomposition of R into $R_1 = \{A, B\}$ and $R_3 = \{B, C\}$ fullfill loss in an dependency presenting?
 - $\Sigma_1 = \{A \rightarrow B\}$ and $\Sigma_3 = \{B \rightarrow C\}$
 - Lossless join? Yes because B is a superkey for R_3 .
 - Dependency preserving? Yes because $(\Sigma_1 \cup \Sigma_3)^* = \Sigma^*$ from the fact that $\{A \to B, B \to C\} \models A \to C$.





Note that:

- 1NF is independent of keys and functional dependencies.
- 2NF, 3NF and BCNF are based on keys and functional dependencies.
- 4NF and 5NF are based on other dependencies (will not be covered in this course).



BCNF

Assignment Project Exam Help

Dd nottrepresent the same fact twice (within a relation)!



BCNF - Definition

Assarato mana nstin Prepierta neutica medica per la neutica ne

https://tutorcs.com



BCNF - Definition

A S-SA reation man Ristin Post of the rest and the Assa SA reation in the Risting Risting Rest of the Risting Risting Rest of the Risting Rest of

 When a relation schema is in BCNF, all data redundancy based on fun treatment for the schema is in BCNF, all data redundancy based on



BCNF - Definition

Assarato mana natio Proposeta nativation Interp

 When a relation schema is in BCNF, all data redundancy based on functional rependency alert novides.

• Here data redundancy is considered in terms of FDs, i.e., for a non-trivial FD $X \rightarrow Y$, there exists a relation R that contains two distinct tuples t, and t, with t [XXI]

We chat: Cstutores t_1 and t_2 with $t_1[XY] = t_2[XY]$.

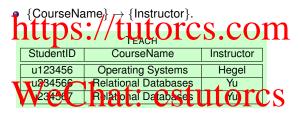
 $\{CourseName\} \rightarrow \{Instructor\}$

· - · · · · · · · · · · · · · · · · · ·				
StudentID	Instructor			
u123456	Hegel			
u234566	Relational Databases	Yu		
u234567	Relational Databases	Yu		



Assated Kida Superkey Project Exam Help

Consider the relation schema TEACH with the following FD:

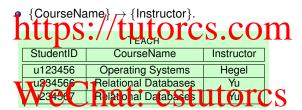


Is TEACH in BCNF?



Assa delation schema Bisin the it whenever a nurtrivial FD X Help Help Assa delation schema Bisin the property of the schema Bisin the schema

Consider the relation schema TEACH with the following FD:



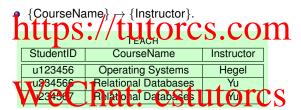
Is TEACH in BCNF?

Not in BCNF because {CourseName} is not a superkey.



Assatelation schema Riskin Project Exam Help

Consider the relation schema TEACH with the following FD:



Is TEACH in BCNF?

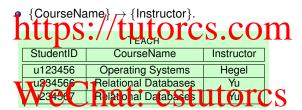
Not in BCNF because {CourseName} is not a superkey.

• Did we represent the same fact twice (or more times)?



Assa relation schema Bisin Project Exam Help

Consider the relation schema TEACH with the following FD:



Is TEACH in BCNF?

Not in BCNF because {CourseName} is not a superkey.

Did we represent the same fact twice (or more times)?
 Yes, the Instructor of Relational Databases is Yu.



Assign ithm for a BGNF (della prosition ect Exam Help

 $\textbf{Output:} \ a \ \mathsf{set} \ \mathcal{S} \ \mathsf{of} \ \mathsf{relation} \ \mathsf{schemas} \ \mathsf{in} \ \mathsf{BCNF}, \ \mathsf{each} \ \mathsf{having} \ \mathsf{a} \ \mathsf{set} \ \mathsf{of} \ \mathsf{FDs}$

https://tutorcs.com



Assign ithm for a BGNFI detemposition ect Exam Help

Output: a set $\mathcal S$ of relation schemas in BCNF, each having a set of FDs

https://tutorcs.com



Assign ithm for a BGNFf de Phoposition et Exam Help

Output: a set S of relation schemas in BCNF, each having a set of FDs

Start with $S = /\{R'\}$; the start with $S = /\{R'\}$; the start with R changes on S:

- Find a (non-trivial) FD $X \rightarrow Y$ on R that violates BCNF, if any;
- Replace R in S by two relation schemas XY and (R-Y) and

We chat: CStutores



Assign ithm for a BGNFf de Phoposition et Exam Help

Output: a set \mathcal{S} of relation schemas in BCNF, each having a set of FDs

Start with $S = \{R'_i\}$; De the following for each O is clear tively until R changes on S:

- Find a (non-trivial) FD $X \rightarrow Y$ on R that violates BCNF, if any;
- Replace R in S by two relation schemas XY and (R-Y) and Y project the FDs to these two relation schemas.
- Does the above Algorithm always produce a lessless decomposition?



Assign ithm for a BGNFf de Phoposition et Exam Help

Output: a set S of relation schemas in BCNF, each having a set of FDs

Start with $S = \{P_i'\}$; the start with $S = \{P_i'\}$ with $S = \{P_i'\}$; the start with $S = \{P_i'\}$ with

- Find a (non-trivial) FD $X \rightarrow Y$ on R that violates BCNF, if any;
- Replace R in S by two relation schemas XY and (R Y) and project the FDs to these two relation schemas.
- Does the above Algorithm always produce a lessless decomposition?

If R with a set Σ of FDs is decomposed into R_1 with Σ_1 and R_2 with Σ_2 , this decomposition is **lossless join** if and only if the common attributes of R_1 and R_2 are a superkey for R_1 or R_2 .



Assign ithm for a BGNFf de Phoposition et Exam Help

Output: a set S of relation schemas in BCNF, each having a set of FDs

- Find a (non-trivial) FD $X \rightarrow Y$ on R that violates BCNF, if any;
- Replace R in S by two relation schemas XY and (R Y) and project the FDs to these two relation schemas.
- Does the above algorithm always produce a lessless decomposition?

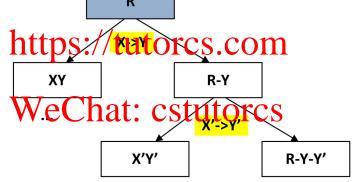
If R with a set Σ of FDs is decomposed into R_1 with Σ_1 and R_2 with Σ_2 , this decomposition is **lossless join** if and only if the common attributes of R_1 and R_2 are a superkey for R_1 or R_2 .

• Yes because X is a superkey for XY.





Assignment Project Exam Help





Assign Riskin Project Exam Help

• Consider the relation schema TEACH with the following FD:

	TEACH		
{CourseNam}ttnstructor}/t	StudentID	CourseName	Instructor
{CourseNaine} Instructor}	u 28458	Operating Systems	Hegel
•	u234566	Relational Databases	Yu
	u234567	Relational Databases	Yu



Assa ber kind superkey Project Exam Help

Consider the relation schema TEACH with the following FD:

	TEACH		
{CourseNam}ttpstrotor/t	StudentID	CourseName	Instructor
{CourseName} lpstruotor}	u 28458	Operating Systems	Hegel
•	u234566	Relational Databases	Yu
	u234567	Relational Databases	Yu

• Cal We hormalised FEACH into BCNE? Utorcs



Assa delation schema Risin PNE if whenever a non-trivial FDX Tholds in PNE if whenever a non-trivial FDX Holds in PNE if whenever a non-trivial FDX Holds

• Consider the relation schema TEACH with the following FD:

	TEACH		
{CourseName}ttps://t	StudentID	CourseName	Instructor
{CourseName} lpstruotor}	u 29455	Operating Systems	Hegel
_	u234566	Relational Databases	Yu
	u234567	Relational Databases	Yu

• Calve hormalise Teach into BCNE? Yet OTGS

<i>i</i> i1			
CourseName	Instructor		
Operating Systems	Hegel		
Relational Databases	Yu		

_				
StudentID	CourseName			
1004EC	Onerating Customs			
u123456	Operating Systems			
u234566	Relational Databases			
u234300	Tielational Databases			
u234567	Relational Databases			
u204307	Ticiational Databases			



Assign Riskin Project Exam Help

Consider the relation schema TEACH with the following FD:

	TEACH		
{CourseNahattustructor}/t	StudentID	CourseName	Instructor
{CourseName} lastruotor}	U128456	Operating Systems	Hegel
_	u234566	Relational Databases	Yu
	u234567	Relational Databases	Yu

• Calve hormalise Teach into BCNE? YETOTGS

**1		
CourseName	Instructor	
Operating Systems	Hegel	
Relational Databases	Yu	

712			
StudentID CourseName			
u123456	Operating Systems		
u234566	Relational Databases		
u234567	Relational Databases		

Do not represent the same fact twice (within a relation)!



Assign premotor point, of xing many the p

- {OfficerID, Date} → {Room}
 {CustomerID, Date} → {OfficerID, Time}
 {Date, Time, Room} → {CustomerID}
- Is Interview in BCNF? If not, normalize Interview into BCNF.

 WeChat: cstutorcs



Assign present of Project, De xam my Welp

```
    {OfficerID, Date} → {Room}
    {CustomerID, Date} → {OfficerID, Time}
    {Dtite!S pate, Fine CustomerID}
    {Date, Time, Room} → {CustomerID}
```

- Is INTERVIEW in BCNF? If not, normalize INTERVIEW into BCNF.
 - {Customerlb, Date}, {Officerlb, Date, Time}, and {Date, Time, Room} are the keys.



Assignment of Project, Paramy Help

- {OfficerID, Date} → {Room}
- {Date, Time, Room} → {CustomerID}
- Is INTERVIEW in BCNF? If not, normalize INTERVIEW into BCNF.
 - {CustomerIb, Date}, {OfficerIb, Date, Time}, and {Date, Time, Room} are the keys.
 - Any superkey must contain one of these keys as a subset.



Assignment of Project, Paramy Help

- $\{OfficerID, Date\} \rightarrow \{Room\}$
- {Date, Time, Room} → {CustomerID}
- Is INTERVIEW in BCNF? If not, normalize INTERVIEW into BCNF.
 - {CustomerIb, Date}, {OfficerIb, Date, Time}, and {Date, Time, Room} are the keys.
 - Any superkey must contain one of of these keys as a subset.
- INTERVIEW is not in BCNF because $\{OfficerID, Date\} \rightarrow \{Room\}$ and $\{OfficerID, Date\}$ is not a superkey.



Assignment Project Lxam Help

INTERVIEW				
OfficerID	CustomerID	Date	Time	Room
\$101,1 P100 / /		12/11/2013	10:00	R15
\$1111	P 105 /	12/(1)2013	12:00	P 5
\$1024	P108′	14/11/2013	14:00	R10
S1024	P107	14/11/2013	14:00	R10

TT 7 (1)					
	N/TERVIEW 1	กลา			
OfficerID	Date	Room			
S1011	12/11/2013	R15			
S1024	14/11/2013	R10			

INTERVIEW2 Officer D				
S1011	P105	12/11/2013	12:00	
S1024	P108	14/11/2013	14:00	
S1024	P107	14/11/2013	14:00	



Assignment Project Examellelp

INTERVIEW					
OfficerID CustomerID Date Time Room					
91011	P100 / /	12/11/2013	10:00	R15	
51)11	P 105	12/(1)2013	2:(0)	715	
\$1024	P108'	14/11/2013	14:00	R10	
S1024	P107	14/11/2013	14:00	R10	

		T		1_		Ε.
	V/V	RVÆW	1	n	121	Ι.
OfficerID	Dat	:e		R	oom	֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓֓
S1011	12/	11/20	13	R	15	1
S1024	14/	11/20	13	R	10	

	INTERVIEW2				
(Officer D (u)t in er D Date Time				
	S1011	P100	12/11/2013	10:00	
	S1011	P105	12/11/2013	12:00	
	S1024	P108	14/11/2013	14:00	
	S1024	P107	14/11/2013	14:00	

Do not represent the same fact twice (within a relation)!



As suppose the consider Interview of the project $\stackrel{\text{Consider Interview}}{=} \{ \text{OfficerID, Date} \} \rightarrow \{ \text{Room} \}$

- {CustomerID, Date} → {OfficerID, Time}
- {OfficerID, Date, Time} → {CustomerID}

https://tutorcs.com

INTERVIEW1			
OfficerID	OfficerID Date Room		
S1011	72/11/2013	R157	
S1024	14/11/2013	IRIGIT.	

_					
	INTERVIEW2				
	OfficerID	CustomerID	Date	Time	
	S1011	P100	12/11/2013	10:00	
	6t0ff11	TPMSTCC	12/11/2013	12:00	
	S4024	HANSE CD	14/11/2013	14:00	
	S1024	P107	14/11/2013	14:00	

Project FDs on two new relation schemas.



As suppose the consider Interview of the project $\stackrel{\text{Consider Interview}}{=} \{ \text{OfficerID, Date} \} \rightarrow \{ \text{Room} \}$

- {CustomerID, Date} → {OfficerID, Time}
- $\bullet \ \{ \text{OfficerID, Date, Time} \} \rightarrow \{ \text{CustomerID} \}$

hftps://tut6rcs.con

INTERVIEW1			
OfficerID	Date	Room	
S1011	72/11/2013	R157 €	
S1024	14/11/2013	Intell.	

~					
	INTERVIEW2				
	OfficerID	CustomerID	Date	Time	
	S1011	P100	12/11/2013	10:00	
	601	PMSC C	12/11/2013	12:00	
	S4024	HANS CD	14/11/2013	14:00	
	S1024	P107	14/11/2013	14:00	

Project FDs on two new relation schemas. INTERVIEW1: {OfficerID, Date} → {Room}



Assignment Project Exam Help

- {CustomerID, Date} → {OfficerID, Time}
- {OfficerID, Date, Time} → {CustomerID}

https://tutorcs.com

INTERVIEW1			
OfficerID	Date	Room	
S1011	72/11/2013	R157 €	
S1024	14/11/2013	Intell.	

~						
	INTERVIEW2					
	OfficerID	CustomerID	Date	Time		
	S1011	P100	12/11/2013	10:00		
	601	TPMST CC	12/11/2013	12:00		
	S4024	HANSE CO	14/11/2013	14:00		
	S1024	P107	14/11/2013	14:00		

• Project FDs on two new relation schemas. INTERVIEW1: $\{OfficerID, Date\} \rightarrow \{Room\}$ INTERVIEW2: $\{CustomerID, Date\} \rightarrow \{OfficerID, Time\}$, $\{OfficerID, Date, Time\} \rightarrow \{CustomerID\}$.



BCNF - Exercise

Assauging Interview = Office ID, Customer ID, Date, Time, Room with the p

- {OfficerID, Date} \rightarrow {Room}
- {CustomerID, Date} → {OfficerID, Time}
- OfficerID, Date, Time

 CustomerID

https://tutorosecom

INTERVIEW1					
OfficerID Nate Boom					
S1011	12/11/2013	IRIGH.			
S1024	14/11/2013	R10			

INTERVIEW2					
OfficerID	CustomerID	Date	Time		
610111	Ptop C	12/11/2013	10:00		
84014	HUS CO	12/11/2013	12:00		
S1024	P108	14/11/2013	14:00		
S1024	P107	14/11/2013	14:00		

Is this decomposition dependency-preservation?



BCNF - Exercise

Assalging Interview = {OfficerID, CustomerID, Date, Time, Room} with the p

- {OfficerID, Date} \rightarrow {Room}
- {CustomerID, Date} → {OfficerID, Time}
- {OfficerID, Date, Time}, → {CustomerID}

https://tutoros.com

INTERVIEW1					
OfficerID Date Boom					
S1011	12/11/2013	HELL.			
S1024	14/11/2013	R10			

INTERVIEW2					
OfficerID	CustomerID	Date	Time		
6401411 1 P1090 C		12/11/2013	10:00		
S4014 U	HUS CO	12/11/2013	12:00		
S1024	P108	14/11/2013	14:00		
S1024	P107	14/11/2013	14:00		

Is this decomposition dependency-preservation?
 No because {Date, Time, Room} → {CustomerID} is lost (and cannot be recovered)!



Assignment Project Exam Help

 When applying BCNF decomposition, the order in which the FDs are applied may lead to different results.

https://tutorcs.com



Assignment Project Exam Help

- When applying BCNF decomposition, the order in which the FDs are applied may lead to different results.
- Example: P(A) = P(A) + P(A



Assignment Project Exam Help

- When applying BCNF decomposition, the order in which the FDs are applied may lead to different results.
- Example: P(A) = P(A) + P(A
 - Case 1: (Using $C \rightarrow B$ first)



- When applying BCNF decomposition, the order in which the FDs are applied may lead to different results.
- Example: P(A) = P(A) + P(A
 - Case 1: (Using $C \rightarrow B$ first)

```
 \begin{array}{l} R_1 = \{B,C\}, \Sigma_1 = \{B \rightarrow C.C \rightarrow B\}; R_2 = \{A,C\}, \Sigma_2 = \{A \rightarrow C\} \\ \hline We Chat: cstutores \end{array}
```



- When applying BCNF decomposition, the order in which the FDs are applied may lead to different results.
- Example: P(A) = P(A) + P(A
 - Case 1: (Using $C \rightarrow B$ first)

```
R_1 = \{B,C\}, \Sigma_1 = \{B \rightarrow C.C \rightarrow B\}; R_2 = \{A,C\}, \Sigma_2 = \{A \rightarrow C\}
```

- When applying BCNF decomposition, the order in which the FDs are applied may lead to different results.
- Example: P(A) = P(A) + P(A
 - Case 1: (Using $C \rightarrow B$ first)

$$R_1 = \{B, C\}, \Sigma_1 = \{B \rightarrow C.C \rightarrow B\}; R_2 = \{A, C\}, \Sigma_2 = \{A \rightarrow C\}$$

$$Stutorcs$$

$$R_1' = \{B, C\}, \Sigma_1' = \{B \to C, C \to B\}; R_2' = \{A, B\}, \Sigma_2' = \{A \to B\};$$



Lossless Join & Dependency Preservation

- So far, we know how to find a lossless BCNF-decomposition, but it may not be lapeled proper serving LOTCS. COM
- Is there a less restrictive normal form such that a lossless and dependency-preserving decomposition can always be found?
 CSTUTOTCS



Lossless Join & Dependency Preservation

- So far, we know how to find a lossless BCNF-decomposition, but it may not be dependency preserving LOTCS.COM
- Is there a less restrictive normal form such that a lossless and dependency-preserving decomposition can always be found?
 Yes rejective national control of the control of the



3NF - Definition

Assignment Project Exam Help

• A relation schema R is in 3NF if whenever a non-trivial FD $X \to A$ holds in R, then X is a superkey or A is a prime attribute . • A relation schema R is in 3NF if whenever a non-trivial FD $X \to A$ holds in R, then X is a superkey or A is a prime attribute .

Question: If R is in BCNF, then R is in 3NF?



3NF - Definition

Assignment Project Exam Help

• A relation schema R is in 3NF if whenever a non-trivial FD $X \to A$ holds in R, then X is a superkey or A is a prime attribute . • A relation schema R is in 3NF if whenever a non-trivial FD $X \to A$ holds in R, then X is a superkey or A is a prime attribute .

Question: # R is in BCNF, then R is in 3NF?

Yes



3NF - Definition

Assignment Project Exam Help

• A relation schema R is in 3NF if whenever a non-trivial FD $X \to A$ holds in R, then X is a superkey or A is a prime attribute . • A relation schema R is in 3NF if whenever a non-trivial FD $X \to A$ holds in R, then X is a superkey or A is a prime attribute .

Question: # R is in BCNF, then R is in 3NF?

Yes

3NF preserves all ineffunctional dependencies at the cost of allowing some data redundancy.



 $Assignment \begin{picture}(20,0) \put(0,0){\line(1,0){19}} \put(0,0){\line(1,0){$

4	ENROL					
ł	1 Studen IC	• Course No	Semester	ConfirmedBy_ID	StaffName	
1	123 56	COMP2400	2010 S2	u12	Jane	
	123458	COMP2400	2008 S2	u13	Linda	
	123458	COMP2600	2008 S2	u13	Linda	



 $Assignment \begin{picture}(20,0) \put(0,0){\line(1,0){19}} \put(0,0){\line(1,0){$

4	ENROL					
ł	Studen IC	• CourseNp1	Semester	ConfirmedBy_ID	StaffName	
1	123 56	COMP2400	2010 S2	u12	Jane	
	123458	COMP2400	2008 S2	u13	Linda	
	123458	COMP2600	2008 S2	u13	Linda	

• Is Weethat: cstutores



 $Assignment \begin{picture}(20,0) \put(0,0){\line(1,0){19}} \put(0,0){\line(1,0){$

4	ENROL					
ŀ	Studentie	• Course No	Semester	Confirmed By_ID	StaffName	
1	123 56	COMP2400	2010 52	u12	Jane	
	123458	COMP2400	2008 S2	u13	Linda	
	123458	COMP2600	2008 S2	u13	Linda	

• Is EvyoLing NF hat CStutorcs
• {Studentito, CourseNo, Semester} is the only key.



 $Assignments, fense \cite{Confirmed By LID} \to \{StaffName\}.$

4	ENROL					
ł	Studentie	• CourseNp1	Semester	Confirmed By_ID	StaffName	
1	123 56	COMP2400	2010 52	u12	Jane	
	123458	COMP2400	2008 S2	u13	Linda	
	123458	COMP2600	2008 S2	u13	Linda	

• Is Evyling INF hat CStutores • {Studentito, Course No, Semester} is the only key.

A relation schema R is in **3NF** if whenever a non-trivial FD $X \to A$ holds in R, then X is a **superkey** or A is a **prime attribute**.



 $Assignment \begin{picture}(200,0) \put(0,0){\line(1,0){19}} \put(0,0){\line(1,0)$

4	ENROL					
ł	T Stude of IC	 Course No 	Semester	Confirmed By_ID	StaffName	
1	123 56	COMP2400	2010 52	u12	Jane	
	123458	COMP2400	2008 S2	u13	Linda	
	123458	COMP2600	2008 S2	u13	Linda	

• Is Evyling NF nat. CStutorcs {Studentib, CourseNo, Semester; is the only key.

A relation schema R is in 3NF if whenever a non-trivial FD $X \to A$ holds in R, then X is a superkey or A is a prime attribute.

 Not in 3NF, because of {ConfirmedBy_ID} → {StaffName}: {ConfirmedBy_ID} is NOT a superkey and {StaffName} is NOT a prime attribute.



Assignment Project Exam Help

Input: a relation schema R and a set Σ of FDs on R.

Output: a set ${\cal S}$ of relation schemas in 3NF, each having a set of FDs $1110\,{\rm FS}$. / $1110\,{\rm FCS}$



Assignment Project Exam Help

Input: a relation schema R and a set Σ of FDs on R.

Output: a set ${\cal S}$ of relation schemas in 3NF, each having a set of FDs Compute a minimal cover ${\cal E}'$ for ${\cal E}$ and start with ${\cal S}=\phi$



Assignment Project Exam Help

Input: a relation schema R and a set Σ of FDs on R.

Output: a set S of relation schemas in 3NF, each having a set of FDs

• Group FDs in Σ' by their left-hand-side attribue sets

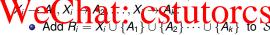


Assignment Project Exam Help

Input: a relation schema R and a set Σ of FDs on R.

Output: a set S of relation schemas in 3NF, each having a set of FDs Compute a minimal cover Σ' for Σ and start with $S = \phi$

- Group FDs in Σ' by their left-hand-side attribue sets
- For each distinct left-hand-side X_i of FDs in Σ' that includes





Assignment Project Exam Help

Input: a relation schema R and a set Σ of FDs on R.

Output: a set S of relation schemas in 3NF, each having a set of FDs Compute a minimal cover Σ' for Σ and start with $S=\phi$

- Group FDs in Σ' by their left-hand-side attribue sets
- For each distinct left-hand-side X_i of FDs in Σ' that includes

$$W$$
-e $(X_i$ -hat: X_i -stutores for X_i -stutores

• Remove all redundant ones from S (i.e., remove R_i if $R_i \subseteq R_i$)

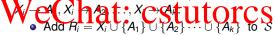


Assignment Project Exam Help

Input: a relation schema R and a set Σ of FDs on R.

Output: a set ${\cal S}$ of relation schemas in 3NF, each having a set of FDs Compute a minimal cover ${\bf Z}'$ for ${\bf Z}$ and start with ${\cal S}=\phi$

- Group FDs in Σ' by their left-hand-side attribue sets
- For each distinct left-hand-side X_i of FDs in Σ' that includes



- Remove all redundant ones from S (i.e., remove R_i if $R_i \subseteq R_i$)
- if S does not contain a superkey of R, add a key of R as R_0 into S.



Assignment Project Exam Help

Input: a relation schema R and a set Σ of FDs on R.

Output: a set ${\cal S}$ of relation schemas in 3NF, each having a set of FDs Compute a minimal cover ${\cal E}'$ for ${\cal E}$ and start with ${\cal S}=\phi$

- Group FDs in Σ' by their left-hand-side attribue sets
- For each distinct left-hand-side X_i of FDs in Σ' that includes

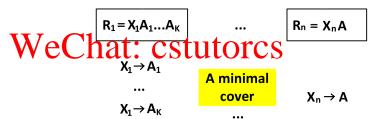
W-e(x) hat: x estutores Add $A_i = x_i \cup \{A_1\} \cup \{A_2\} \cdots \cup \{A_k\}$ to s

- Remove all redundant ones from S (i.e., remove R_i if $R_i \subseteq R_j$)
- if S does not contain a superkey of R, add a key of R as R_0 into S.
- Project the FDs in Σ' onto each relation schema in S

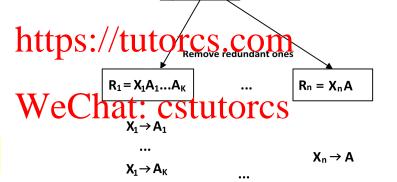


Assignment Project Exam Help

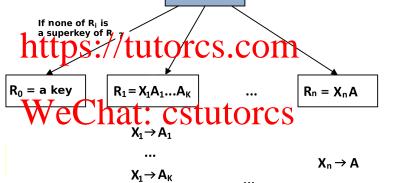
https://tutorcs.com













Assignment Project Exam Help

https://tutores.com 5.*



Assignment Project Exam Help





Assignment Project Exam Help



• Example 1: Σ_1 **Chat** Σ_2 **Cot*** Σ_2 If $\Sigma_1^* = \Sigma_2^*$, then Σ_1 is not minimal





- Example 1: Chat: $\Sigma_1 = \{X \in \mathbb{C} \mid A \in \mathbb{C} \}$ If $\Sigma_1^* = \Sigma_2^*$, then Σ_1 is not minimal
- Example 2:

$$\Sigma_1=\{X\to Y,XY\to Z\} \text{ and } \Sigma_2=\{X\to Y,X\to Z\}$$
 If $\Sigma_1^*=\Sigma_2^*,$





- Example 1: Chat: $\Sigma_1 = \{X \in \mathbb{C} \mid A \in \mathbb{C} \}$ If $\Sigma_1^* = \Sigma_2^*$, then Σ_1 is not minimal
- Example 2:

$$\begin{split} \Sigma_1 &= \{X \to Y, XY \to Z\} \text{ and } \Sigma_2 = \{X \to Y, X \to Z\} \\ \text{If } \Sigma_1^* &= \Sigma_2^*, \text{ then } \Sigma_1 \text{ is not } \underset{\textbf{minimal}}{\textbf{minimal}} \end{split}$$



Assignment Project Exam Help



- Example 1: Σ_1 ** Chat: Σ_2 ** Case ** Lite ** Label** Σ_1 ** If $\Sigma_1^* = \Sigma_2^*$, then Σ_1 is not minimal
- Example 2:

$$\Sigma_1 = \{X \to Y, XY \to Z\}$$
 and $\Sigma_2 = \{X \to Y, X \to Z\}$
If $\Sigma_1^* = \Sigma_2^*$, then Σ_1 is not **minimal**

Questions: Can we find the minimal one among equivalent sets of FDs?



Minimal Cover - The Hard Part!

Assignment Project Exam Help

https://tutorcs.com



Minimal Cover – The Hard Part!

Assignment Project Exam Help

 $\begin{array}{l} \Sigma_{\it m} \text{ is equivalent, to } \Sigma, \text{ i.e., start with } \Sigma_{\it m} = \Sigma; \\ \textbf{https://tutorcs.com} \end{array}$



Minimal Cover – The Hard Part!

Assignment Project Exam Help

- Σ_m is equivalent to Σ , i.e., start with $\Sigma_m = \Sigma$;
 - **Dependent:** each FD in Σ_m has only a single attribute on its right hand side, i.e., replace each FD $X \to \{A_1, \ldots, A_k\}$ in Σ_m with $X \to A_1, \ldots, X \to A_k$;



Minimal Cover – The Hard Part!

Assignment Project Exam Help Let Doe a set of FDs. A minimal ouver Σ_m of Σ is a set of FDs such that

- Σ_m is equivalent to Σ , i.e., start with $\Sigma_m = \Sigma$;

 Dependent each FD in Σ_m has only a single attribute on its right hand side, i.e., replace each FD $X \to \{A_1, \ldots, A_k\}$ in Σ_m with $X \to A_1, \ldots, X \to A_k$;
 - possible, i.e., for each FD $X \to A$ in Σ_m , check each attribute B of X to see if we can replace $X \to A$ with $(X B) \to A$ in Σ_m ;



Minimal Cover - The Hard Part!

Assignment Project Exam Help Let Doe a set of FDs. A minimal ouver Σ_m of Σ is a set of FDs such that

- \bigcirc Σ_m is equivalent, to Σ , i.e., start with $\Sigma_m = \Sigma$;
- hand side, i.e., replace each FD $X \to \{A_1, \ldots, A_k\}$ in Σ_m with $X \to A_1, \ldots, X \to A_k$;
- **Solution** name that FD has safely attribute on the left hand side as possible, i.e., for each FD $X \to A$ in Σ_m , check each attribute B of X to see if we can replace $X \to A$ with $(X B) \to A$ in Σ_m ;
- **a** Remove a FD from Σ_m if it is redundant.



Given the set of FDs Σ

 $SS_{\text{ConfirmedBy}, ID} \to \{\text{StaffName}\} \text{ entropic methy partition} + ID$

• we can compute the minimal cover of Σ as follows:

https://tutorcs.com



Given the set of FDs Σ

 $SS\{\text{prompressor} \text{Semestry}\} \text{ Confirmed By JD} \rightarrow \{\text{Staff Name}\}$

• we can compute the minimal cover of Σ as follows:

https://tutorcs.com



Given the set of FDs Σ

 $s_{\text{ConfirmedBy_ID}} \to \{s_{\text{taffName}}\} \text{ confirmedBy_D} \to \{s_{\text{taffName}}\} \text{ confirmedBy_D}$

• we can compute the minimal cover of Σ as follows:

start from Σ ; check yiether all the FDs in Z have only one athibute on the right hand side.



Given the set of FDs Σ

• we can compute the minimal cover of Σ as follows:

start from Σ ; chack yiether all the FDs in Z there only one athibute on the right hand side, $\{ StudentID, CourseNo, Semester \} \rightarrow \{ \textcolor{red}{\textbf{ConfirmedBy_ID, StaffName}} \}$



SSIPPEND SET OF FDS E SET OF CONTINUE BYXD STIFFE FOR THE PROPERTY OF THE PROP

 $\{ConfirmedBy_ID\} \rightarrow \{StaffName\}$

ullet we can compute the minimal cover of Σ as follows:

start from Σ;

check vicether all tine FDs in Σ have only one attribute on the right hand side;

{StudentID, CourseNo, Semester} → { ConfirmedBy_ID, StaffName } can be replaced by

StudentID, CourseNo, Semester} → { ConfirmedBy_ID } { StudentID, CourseNo, Semester} → { StaffName }



Given the set of FDs Σ

SS{Figure 11, pp. per 17 of Semester} + (StudentID, CourseNo, Semester) + (StudentI

 $\{ConfirmedBy_ID\} \rightarrow \{StaffName\}$

• we can compute the minimal cover of Σ as follows:

sating Σ; //tutorcs.com
check whether all the FDs in Σ have only one attribute on the right hand side;



Given the set of FDs Σ Signally, present Semesters by Confirmed ByxDam Help (StudentID, CourseNo, Semester) \rightarrow {StaffName}

{ConfirmedBy_ID} → {StaffName}

- we can compute the minimal cover of Σ as follows:
 - $\begin{array}{c} \text{ The proof of the pro$
 - hand side; \bigcirc check whether all the FDs in Σ have redundant attribute on the left



Given the set of FDs Σ

SS{Figrenting Semester} → {StaffName}

Help

StudentID, CourseNo, Semester} → {StaffName}

• we can compute the minimal cover of Σ as follows:

 $\{ConfirmedBy_ID\} \rightarrow \{StaffName\}$

- The state of the
- check whether all the FDs in Σ have redundant attribute on the left and side; Studentib, Courselve, Serbester CS (ConfirmedBy_ID) is minimal with respect to the left hand side check if { StudentID, CourseNo, Semester } \rightarrow {StaffName} is minimal with respect to the left hand side



Given the set of FDs Σ

SSTRUCTURE TO SE MESTER) + (StaffName)

StudentID, CourseNo, Semester) + (StaffName)

 $\{ConfirmedBy_ID\} \rightarrow \{StaffName\}$

- we can compute the minimal cover of Σ as follows:
 - sating Σ; //tutorcs.com
 check whether all the FDs in Σ have only one attribute on the right hand side;
 - check whether all the FDs in Σ have redundant attribute on the left and size;

 Land size



• Given the set of FDs Σ

 $\begin{array}{c} \textbf{ASS[Figer III] Performance Student ID, Course No, Semester]} \rightarrow \{\textbf{StaffName}\} \\ \{\textbf{Confirmed By_ID}\} \rightarrow \{\textbf{StaffName}\} \end{array}$

- we can compute the minimal cover of Σ as follows:
 - The state of the
 - $\ensuremath{ f 0}$ check whether all the FDs in Σ have redundant attribute on the left



- Given the set of FDs Σ S{Figure 110, CourseNo, Semester} \rightarrow {StaffName} {ConfirmedBy_ID} \rightarrow {StaffName}
 - we can compute the minimal cover of Σ as follows:
 - check whether all the FDs in Σ have only one attribute on the right hand side;
 - $oldsymbol{0}$ check whether all the FDs in Σ have redundant attribute on the left



Given the set of FDs Σ S{Figure 110, CourseNo, Semester} \rightarrow {StaffName} {ConfirmedBy_ID} \rightarrow {StaffName}

- we can compute the minimal cover of Σ as follows:
 - sating Σ; //tutorcs.com
 check whether all the FDs in Σ have only one attribute on the right hand side;
 - $\ensuremath{ f 0}$ check whether all the FDs in Σ have redundant attribute on the left
 - $\label{eq:confirmed} \begin{tabular}{ll} \begin{tabular}{ll} and side; \\ \begin{tabular}{ll} All & \begin{tabular}{ll}$
 - {StudentID, CourseNo, Semester} → {StaffName} is redundant and thus is removed



 Given the set of FDs Σ Specific Sements of StaffName | StaffName $\{ConfirmedBy_ID\} \rightarrow \{StaffName\}$

- we can compute the minimal cover of Σ as follows:
 - Utorcs.com ne FDs in Σ have only one attribute on the right hand side;
 - check whether all the FDs in Σ have redundant attribute on the left
 - hand side; hand FD C Student D to use No, Semester} \rightarrow {ConfirmedBy_ID}, {StudentID, CourseNo, Semester} → $\{StaffName\}, \{ConfirmedBy_ID\} \rightarrow \{StaffName\}\}$
 - {StudentID, CourseNo, Semester} → {StaffName} is redundant and thus is removed
 - Therefore, the minial cover of Σ is { StudentID, CourseNo, $\overline{\text{Semester}} \rightarrow \{\text{ConfirmedBy_ID}\}, \{\text{ConfirmedBy_ID}\} \rightarrow \{\text{StaffName}\}\}_{38/50}$



Assignmented by the project of the state of

 $\bullet \ \{\mathsf{ConfirmedBy_ID}\} \to \{\mathsf{StaffName}\}$

Stud	dentID	909.00.10		2011100101					
htt	DS	i//tut	orcs	.com					

 Can we normalise ENROL into 3NF by a lossless and dependency preserving decomposition?



$Assignment & Consider & Enrol & again: & Project & Farmer & Help & Confirmed & By_D, Staff Name & Confirmed & Staff Name & Staff Name$

	StudentID	<u>CourseNo</u>	<u>Semester</u>	ConfirmedBy_ID	StaffName
r	ittps:	://tut	Ofcs	.com	

- A minimal cover is {{StudentID, CourseNo, Semester} → {ConfirmedBy_ID}, {ConfirmedBy_ID} → {StaffName}}.
- WeChat: cstutorcs



Assignments, double No, semesters C (Confirmed By ID) again: Project Example (Confirmed By ID) -> {StaffName}

					rse	eΝ	0		S	en	ne	ester ConfirmedBy_ID						Sta	ffNar	ne							
ľ	F	t	t	D	S		/	/ '	t	u	Ĭ	Ţ)	1	(S	•	$\overline{\mathbb{C}}$	\overline{C})]	$\mathbf{\eta}$	1				

- A minimal cover is {{StudentID, CourseNo, Semester} → {ConfirmedBy_ID}, {ConfirmedBy_ID} → {StaffName}}.
- Hence, we have: • WY = Studentil , Coursello, Semester, Confirmed By ID} with {StudentID, CourseNo, Semester} → {Confirmed By ID}



Assignments, double No, semesters C (Confirmed By ID) again: Project Example (Confirmed By ID) -> {StaffName}

	StudentID			C	oų	rse	No	2	<u>S</u>	e)	me	ste	<u>er</u>	C	on	firr	ne	dΒ	By_	ID	Ι	StaffN	ame		
ľ	F	t		D	S	/	/	t	u	t	•)	r	\overline{C}	S		$\overline{\mathbb{C}}$	$\overline{\mathbf{C}}$)]	n	1				

- A minimar cover is {{StudentID, CourseNo, Semester} → {ConfirmedBy_ID}, {ConfirmedBy_ID} → {StaffName}}.
- Hence, we have: VV = Studentil , Coursello, Semester, Confirmed By ID} with {StudentID, CourseNo, Semester} → {Confirmed By ID}
 - R₂={ConfirmedBy_ID, StaffName} with {ConfirmedBy_ID} → {StaffName}



Assignments, dourselve, semesers C (Confirmed By ID) again: Project Fxam, taffinante) policy (Confirmed By ID) -> {StaffName}

	<u>StudentID</u>	<u>CourseNo</u>	<u>Semester</u>	ConfirmedBy ₋ ID	StaffName
r	ittps:	://tut	OFCS	.com	

- A minimar cover is {{StudentID, CourseNo, Semester} → {ConfirmedBy_ID}, {ConfirmedBy_ID} → {StaffName}}.
- Hence, we have: VV = Studentil Acurse C Semester. Confirmed By ID} with {StudentID, Course No, Semester} → {Confirmed By ID}
 - R₂={ConfirmedBy_ID, StaffName} with {ConfirmedBy_ID} → {StaffName}
 - Omit R_0 because R_1 is a superkey of ENROL.



Assignments, course No, semeser < to first each of the confirmed by LD; et a final and seminated by LD; et a final and semin

	StudentID	CourseNo	<u>Semester</u>	ConfirmedBy_ID	StaffName		
h	ittps:	://tut	Orcs	.com			

- A minimar cover is {{StudentID, CourseNo, Semester} → {ConfirmedBy_ID}, {ConfirmedBy_ID} → {StaffName}}.
- Hence, we have: VV = Studentil , I curse to Semester, Confirmed By LID } with {StudentID, Course No, Semester} → {Confirmed By LID}
 - R₂={ConfirmedBy_ID, StaffName} with {ConfirmedBy_ID} → {StaffName}
 - Omit R_0 because R_1 is a superkey of ENROL.
- Is {StudentID, CourseNo, Semester} → {ConfirmedBy_ID, StaffName} preserved?



$Assignment & Consider Enrol again: Project Example 1 project Confirmed By LD; et aff Name \} \\ & \{Confirmed By LD\} \rightarrow \{Staff Name\}$

St	<u>udentID</u>	CourseNo	Semester	Semester ConfirmedBy_ID						
ht	tps:	://tu	orcs	.com						

- A minimal cover is {{StudentID, CourseNo, Semester} → {ConfirmedBy_ID}, {ConfirmedBy_ID} → {StaffName}}.
- Hence, we have: • VA = Studentil, Acursello, Semester, Confirmed By_ID} with {StudentID, CourseNo, Semester} → {Confirmed By_ID}
 - R₂={ConfirmedBy_ID, StaffName} with {ConfirmedBy_ID} → {StaffName}
 - Omit R_0 because R_1 is a superkey of ENROL.
- Is {StudentID, CourseNo, Semester} → {ConfirmedBy_ID, StaffName} preserved? Yes.





...

httips bate tutores.com

- ② {CustomerID, Date} → {OfficerID, Time}
- - $\{Date, Time Room\} \rightarrow \{CustomerlD\}$
- Is IN IN THE in THE Inot, TO SHE IN THE WINTO I OSSIESS and dependency preserving 3NF.



Assignment Project Exam Help OfficerID CustomerID Date Time Room

12/11/2013

R15

...

10:00

https://tutorcs.com

{CustomerID, Date} → {OfficerID, Time}

P100

S1011

- - $\{ A \setminus Date, Time \mid Room \} \rightarrow \{CustomerlD\} \}$
- Is In the in 30-21 not, to sail to the first to lossless and dependency preserving 3NF.
 - A relation schema R is in **3NF** if whenever a non-trivial FD $X \to A$ holds in R, then X is a **superkey** or A is a **prime attribute**.



Assignment Project Exam Help

\$1011 P100 12/11/2013 10:00 R15

https://tutorcs.com

- ② {CustomerID, Date} → {OfficerID, Time}
- - $\{Date, Time Room\} \rightarrow \{CustomerlD\}$
- Is IN THE in The 21 fnot, to this to the lossless and dependency preserving 3NF.
 - A relation schema R is in **3NF** if whenever a non-trivial FD $X \to A$ holds in R, then X is a **superkey** or A is a **prime attribute**.
 - We know that {CustomerID, Date}, {OfficerID, Date, Time}, and {Date, Time, Room} are the keys.



Assignment Project Exam Help

12/11/2013

R15

...

10:00

ttipis intertutores.com

2 {CustomerID, Date} → {OfficerID, Time}

P100

- § OfficerID, Date, Time} → {CustomerID}

S1011

- Is IN IN THE in 30 F21 (not, 10 Shifts) INTER THE Who lossless and dependency preserving 3NF.
 - A relation schema R is in **3NF** if whenever a non-trivial FD $X \to A$ holds in R, then X is a **superkey** or A is a **prime attribute**.
 - We know that {CustomerID, Date}, {OfficerID, Date, Time}, and {Date, Time, Room} are the keys.

INTERVIEW is in 3NF because all the attributes are prime attributes.



Assignment Project Fxam, Help the following FDS:

• FD1: PropertyID → Lot, County, Area

FD2: Lot, County Area, PropertyID

FD8: Area, County LOTCS. COM



Assignment Project Fxam, Help the following FDS:

- FD1: PropertyID → Lot, County, Area
 FD2: Lot, County Area, PropertyID
 FD8: Area, County LOTCS.COM
- Let us abbriviate attributes of LOTS with first letter of each attribute and replequent ou set of dependencies as F; $\{P \cap LCA, LC \to AP, A \to C\}$
- The minimal cover of a set of functional dependencies always exists but is not necessarily unique.



Assignment Project Exam Help

https://tutorcs.com



Assignment Project Acx an Help initialise: $\{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$

https://tutorcs.com



```
Assignment Projecta Exam Help Initialise: {P - LCA, LC - AP, A - C}
```

Dependent: $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}$.

https://tutorcs.com



```
A SSC Infinite character team of the point of the point
```



```
A SSEQUENTIAL PROPERTY AND SECURE AND A SECONDARY AND SECURE AND A SECONDARY AND A SECURE AND A
```



Assignment Project Exam Help

- Initialise: $\{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$
- **Dependent:** $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}.$
- **Determinant:** $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}.$
- **Figure 9.** Thus, a minimal cover is $\{P \rightarrow E, P \rightarrow C, EC \rightarrow A, EC \rightarrow P, A \rightarrow C\}$ or $\{P \rightarrow LC, LC \rightarrow AP, A \rightarrow C\}.$



ssignment Project Exam Help

- **Initialise:** $\{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$
- **Dependent:** $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}.$
- **Determinant:** $(P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C)$.
- Fig. 1. Fig. $\{P \rightarrow LC, LC \rightarrow AP, A \rightarrow C\}.$
- (Case Y) Find a minimal cover of $F = \{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$ WeChat: cstutorcs



- **Initialise:** $\{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$
- **Dependent:** $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}.$
- **Determinant:** $(P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C)$.
- **Fig. 1.** Thus a minimal cover is $\{P \rightarrow L, P \rightarrow C, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}$ or $\{P \rightarrow LC, LC \rightarrow AP, A \rightarrow C\}.$
- (Case Y) Find a minimal cover of $F = \{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$
 - Min Chata cstutores



- Initialise: $\{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$
- **Dependent:** $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}.$
- **Determinant:** $(P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C)$.
- **Fig. 1.** Thus a minimal cover is $\{P \rightarrow L, P \rightarrow C, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}$ or $\{P \rightarrow LC, LC \rightarrow AP, A \rightarrow C\}.$
- (Case Y) Find a minimal cover of $F = \{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$
 - Tighte: $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}$.



- Initialise: $\{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$
- **Dependent:** $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}.$
- **Determinant:** $(P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C)$.
- **Figure 9.1.** Thus a minimal cover is $\{P \rightarrow L, P \rightarrow C, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}$ or $\{P \rightarrow LC, LC \rightarrow AP, A \rightarrow C\}.$
- (Case Y) Find a minimal cover of $F = \{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$

 - $\begin{array}{ll} \text{Mtiffle: } & \text{PATA, LCSA, NTCS} \\ \text{Dependent: } & \text{P} \rightarrow \text{L}, \text{P} \rightarrow \text{C}, \text{P} \rightarrow \text{A}, \text{LC} \rightarrow \text{A}, \text{LC} \rightarrow \text{P}, \text{A} \rightarrow \text{C}}. \end{array}$
 - **Determinant:** $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}.$



- **Initialise:** $\{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$
- **Dependent:** $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}.$
- **Determinant:** $_{I}\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}.$
- **Figure 9.** Thus a minimal cover is $\{P \rightarrow L, P \rightarrow C, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}$ or $\{P \rightarrow LC, LC \rightarrow AP, A \rightarrow C\}.$
- (Case Y) Find a minimal cover of $F = \{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$

 - white $P \rightarrow L$, $P \rightarrow C$, $P \rightarrow A$, $LC \rightarrow A$, $LC \rightarrow P$, $A \rightarrow C$.
 - **3** Determinant: $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}$.
 - **Remove redundant FD:** LC \rightarrow A and P \rightarrow C are redundant.



- Initialise: $\{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$
- **Dependent:** $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}.$
- **Determinant:** $(P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C)$.
- **February Section (a)** Thus a minimal cover is $\{P \rightarrow L, P \rightarrow C, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}$ or $\{P \rightarrow LC, LC \rightarrow AP, A \rightarrow C\}.$
- (Case Y) Find a minimal cover of $F = \{P \rightarrow LCA, LC \rightarrow AP, A \rightarrow C\}$
 - white $P \rightarrow L$, $P \rightarrow C$, $P \rightarrow A$, $LC \rightarrow A$, $LC \rightarrow P$, $A \rightarrow C$.

 - **3 Determinant:** $\{P \rightarrow L, P \rightarrow C, P \rightarrow A, LC \rightarrow A, LC \rightarrow P, A \rightarrow C\}.$
 - **Remove redundant FD:** LC \rightarrow A and P \rightarrow C are redundant.
 - **5** Thus a minimal cover is $\{P \rightarrow L, P \rightarrow A, LC \rightarrow P, A \rightarrow C\}$ or $\{P \rightarrow LA, LC \rightarrow P, A \rightarrow C\}$.



Assignment Project Exam Help **BONF: Whenever a non-trivial FD X **A holds in R, then X is a superkey.**

https://tutorcs.com



Assignment Project Exam Help **BONF: Whenever a non-trivial FD ** A holds in R, then X is a superkey.**

Ponet represent the same fast more than once within a relation, seven it some Ebs have to be abandoned!



Assignment Project Exam Help

BCNF: Whenever a non-trivial FD $X \rightarrow A$ holds in R, then X is a **superkey**.

Ponet represent the same fast more than once within a relation, seven it some Ebs have to be abandoned!

• 3NF: Whenever a non-trivial FD $X \to A$ holds in R, then or Aisa prime attribute. CSTUTOTCS



Assignment Project Exam Help BONF: Whenever a non-trivial FD X A holds in R, then

BCNF: Whenever a non-trivial FD $X \rightarrow A$ holds in R, then X is a **superkey**.

Ponet represent the same fast more than once within a relation, Seven it some Ebs have to be abandoned!

• 3NF: Whenever a non-trivial FD $X \to A$ holds in R, then or A is a superkey or A is a prime attribute. CSTUTOTCS

Do not abandon any FDs, even if some facts have to be represented more than once within a relation!



Assignment Project Exam Help

- Repeat until no changes
 - Find a problematic FD
 - Spit Flinto Co small light OTCS.COM



gament Projecto wan Help

- Repeat until no changes
 - Find a problematic FD
 - of Rinto No smale on the CS Remove rentondant ones
- Find a minimal cover
- Group FDs in the minimal cover

 - Add a key (if necessary)
 - Project FDs



ssignment Projectomean Help

- Repeat until no changes
 - Find a problematic FD
 - pt Fling 80 smale Interores Remove perfondant ones
- Find a minimal cover
- Group FDs in the minimal cover

 - Add a key (if necessary)
 - Project FDs

Wortportes 6 stell off shave?



gament Projecto Exam Help

- Repeat until no changes
 - Find a problematic FD
 - ot Flirto No smale Unit OTCS Remove rentondant ones
- Find a minimal cover
- Group FDs in the minimal cover
- - Add a key (if necessary)
 - Project FDs

World parties of stelling from Shave?



Lossless join

Lossless join + dependency preservation



Revise composition Properties Help

- Repeat until no changes
 - Find a problematic FD
 - Split Rinto two smaller eneso TCS Remove redundant ones and project FDs. Remove redundant ones
- Group FDs in the minimal cover

 - Project FDs

What dayoutneed to dompute using FDs?



schigeophoeinit Projucture in Help

- Repeat until no changes
 - Find a problematic FD
 - Split Reinte two smaller enes TCS Remove redundant ones and project FDs. Remove redundant ones
- Group FDs in the minimal cover
- - Project FDs

What dayoutneed to dompute using FDs?

SOME superkeys (check)

SOME superkeys (check) ALL candidate keys ONE minimal cover



Denormalisation

pormatizablation schemas in a cases when disiliping p

- Denormalisation is a design process that
 - happens after the normalisation process,
 - is often performed during the physical design sale, and
 - reduces the number of relations that need to be joined for certain queries.
- Unnormalised there is no systematic design.

 - Normalised redundancy is reduced after a systematic design (to minimise data inconsistencies).
 - Denormalised redundancy is introduced after analysing the normalised design (to improve efficiency of queries)



Data redundancies are eliminated in the following relations.





A SSNormalisation to prite Redundant to the following relations.



• However, the query for "list the names of students who enrolled in a course with vnis equiles 2 it in operations 110105

```
SELECT Name, CourseNo

FROM ENROL e, COURSE c, STUDENT s

WHERE e.StudentID=s.StudentID AND e.CourseNo=c.CourseNo
AND c.Unit=6;
```



Assignment Project Exam Help

 If a student enrolled 15 courses, then the name and DoB of this student need to be stored repeatedly 15 times in ENROLMENT.

ł	https://tutorcs.com								
	Name	StudentID	DoB	<u>CourseNo</u>	Semester	Unit			
	Tom	123456	25/01/1988	COMP2400	2010 S2	6			
4	Tom	123456	25/01/1988	COMP8740	2011 S2	12			
١	Michae	284581	21/04(1985	# PANT2400 C	2009 S2	6			



Assignment Project Exame Help

 If a student enrolled 15 courses, then the name and DoB of this student need to be stored repeatedly 15 times in ENROLMENT.

ľ	https://tutorcs.com							
	Name	StudentID	DoB	<u>CourseNo</u>	Semester	Unit		
	Tom	123456	25/01/1988	COMP2400	2010 S2	6		
¥	Tom	123456	25/01/1988	COMP8740	2011 S2	12		
1	Wichae	284581	21/04/1935	10MP2400	2009 S2	6		

 The query for "list the names of students who enrolled a course with 6 units" can be processed efficiently (no join needed).

SELECT Name, CourseNo FROM ENROLMENT WHERE Unit=6;



(credit cookie) Raymond F. Boyce (1947-1974)

Assignment Project Exam Help

Donald D. Chamberlin Raymond F. Boyce

https://tutespare.laboratory

etructured English query language (SEQUEL) which can be used for accessing data in an integrated relational data base. Without resorting to the concepts of bound variables and quantifiers SEQUEL identifies a set of simple operations on tabular structures, which can be shown to be of equivalent power to the first order predicate calculus. A SEQUEL user is presented with a consistent set of keyword English templates which reflect how people use tables to

tent set or keyword inglish templates which reflect now people use tables to other infringation, foreover, the SOUZI user is alle to compose these basic contains it a structured manor it order to from nor it makes exercise.

SPOUL is introduced a data large publication for tooth the tarofessional programmer and the more infrequent data base user.

"SEQUEL: A Structured English Query Language", D.D. Chamberlin and R.F. Boyce,

Proc. ACM SIGMOD Workshop on Data Description, Access and Control,

Ann Arbor, Michigan (May 1974)