

Lecture 14

Normalization

Normalization

- The process of decomposing unsatisfactory "bad" relations by breaking up their attributes into smaller relations
- The process of analyzing the given relation schemas based on their FDs and primary keys to achieve the desirable properties:
 - Minimizing redundancy
 - Minimizing the insertion, deletion, and update anomalies

Normal Forms

- If a relation is in a certain *normal form* (BCNF, 3NF etc.), it is known that certain kinds of problems are avoided
- Normal Form: the highest normal form condition that it meets, and hence indicates the degree to which it has been normalized

1NF

- 1NF: disallows multivalued attributes, composite attributes and their combinations
- The only attribute values permitted by 1NF are single atomic values
- Considered to be part of the definition of relation

Example 1: Normalization into 1NF

DEPARTMENT

Dname	<u>Dnumber</u>	Dmgr_ssn	Dlocations
Research	5	333445555	{Bellaire, Sugarland, Houston}
Administration	4	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

DEPARTMENT

Dname	<u>Dnumber</u>	Dmgr_ssn	<u>Dlocation</u>
Research	5	333445555	Bellaire
Research	5	333445555	Sugarland
Research	5	333445555	Houston
Administration	4	987654321	Stafford
Headquarters	1	888665555	Houston

DEPARTMENT

Dname	<u>Dnumber</u>	Dmgr_ssn
Research	5	333445555
Administration	4	987654321
Headquarters	1	888665555

DEPT_LOCATIONS

<u>Dnumber</u>	<u>Dlocation</u>
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

Example 2: Normalization into 1NF

EMP_PROJ		Projs	
Ssn	Ename	Pnumber	Hours

EMP_PROJ			
Ssn	Ename	Pnumber	Hours
123456789	Smith, John B.	1	32.5
		2	7.5
666884444	Narayan, Ramesh K.	3	40.0
453453453	English, Joyce A.	1	20.0
		2	20.0
333445555	Wong, Franklin T.	2	10.0
		3	10.0
		10	10.0
		20	10.0
999887777	Zelaya, AliciaJ.	30	30.0
		10	10.0
987987987	Jabbar, Ahmad V.	10	35.0
		30	5.0
987654321	Wallace, Jennifer S.	30	20.0
		20	15.0
888665555	Borg, James E.	20	NULL

EMP_PROJ1	
<u>Ssn</u>	Ename

EMP_PROJ2		
<u>Ssn</u>	<u>Pnumber</u>	Hours

Prime or Nonprime Attributes

- **Prime attribute:** a member of *some* candidate key
- **Nonprime attribute:** not a prime attribute (not a member of any candidate key)

Full or Partial Functional Dependency

- $X \rightarrow Y$ is a full functional dependency if removal of any attribute A from X means that the dependency does not hold any more
- $X \rightarrow Y$ is a partial functional dependency if some attribute A can be removed from X and the dependency still holds

2NF

- 2NF: if it is in 1NF and every nonprime attribute A in R is fully functionally dependent on the primary key
- Normalize 1NF relations to 2NF: remove partial functional dependencies by placing the RHS attribute(s) in a new relation along with a copy of the determinant(s)



When a 1NF table has no composite PK, is the table already in 2NF?

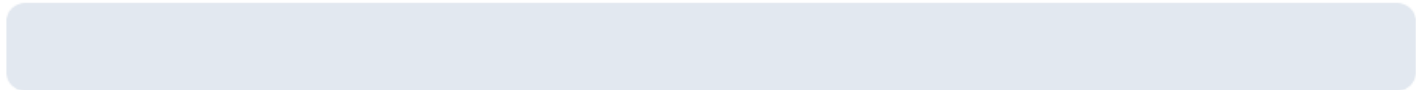
Yes

No



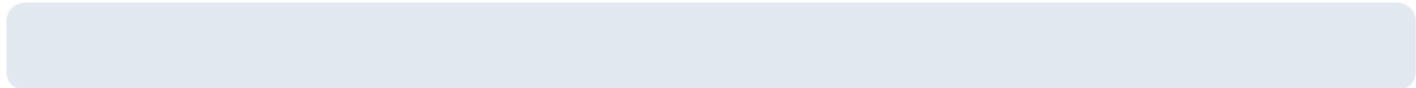
When a 1NF table has no composite PK, is the table already in 2NF?

Yes



0%

No

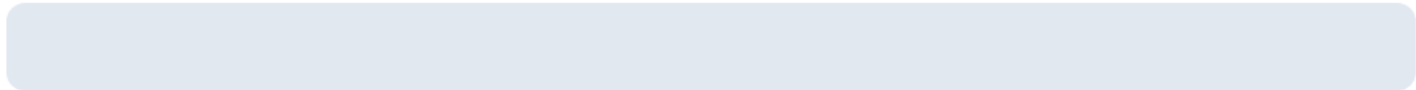


0%



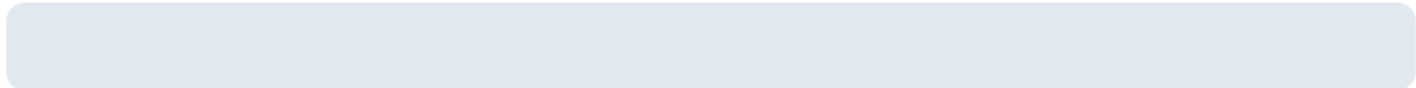
When a 1NF table has no composite PK, is the table already in 2NF?

Yes



0%

No

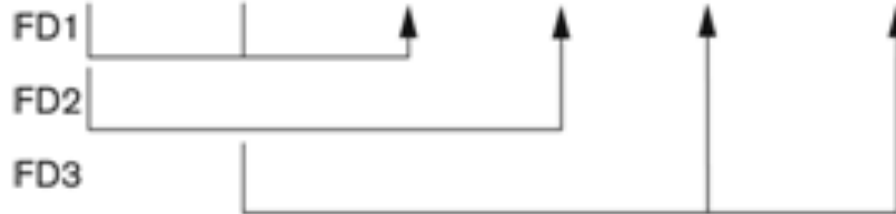


0%

Example: Normalization into 2NF

EMP_PROJ

<u>Ssn</u>	<u>Pnumber</u>	Hours	Ename	Pname	Plocation
------------	----------------	-------	-------	-------	-----------



2NF Normalization

EP1

<u>Ssn</u>	<u>Pnumber</u>	Hours
------------	----------------	-------



EP2

<u>Ssn</u>	Ename
------------	-------



EP3

<u>Pnumber</u>	Pname	Plocation
----------------	-------	-----------



3NF

- Transitive functional dependency: If $A \rightarrow B$ and $B \rightarrow C$, then C is transitively dependent on A via B
- 3NF: if it is in 2NF and no nonprime attribute A in R is transitively dependent on the primary key
- Normalize 2NF relations to 3NF: remove transitive dependencies by placing the RHS attribute(s) in a new relation along with a copy of the determinant(s)



If all attributes of a table are prime attributes, is the relation already in 3NF?

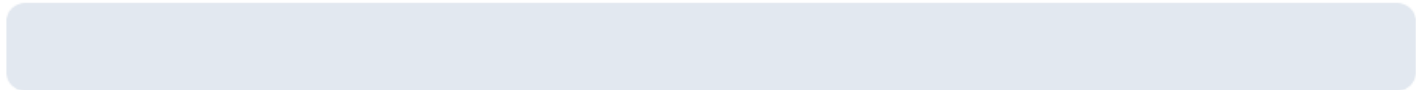
Yes

No



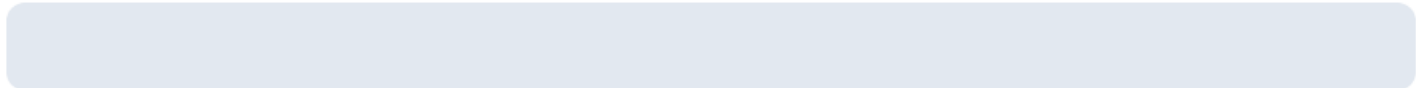
If all attributes of a table are prime attributes, is the relation already in 3NF?

Yes



0%

No

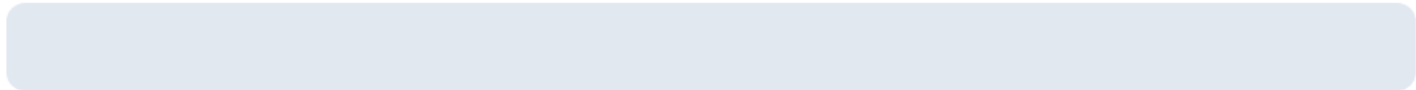


0%



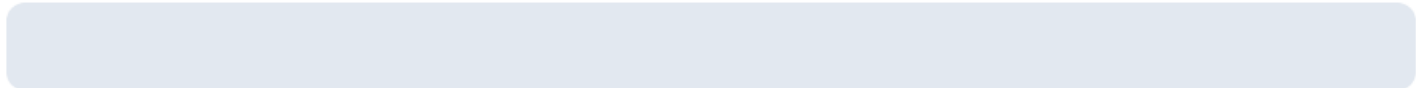
If all attributes of a table are prime attributes, is the relation already in 3NF?

Yes



0%

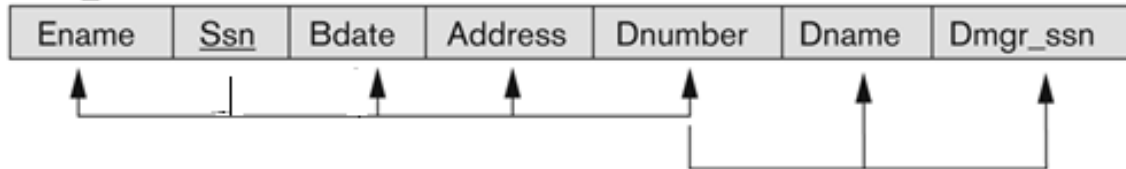
No



0%

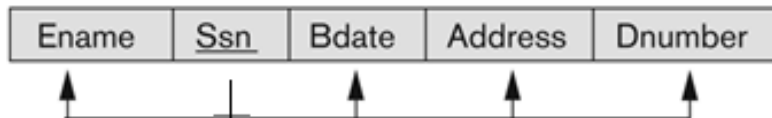
Example: Normalization into 3NF

EMP_DEPT

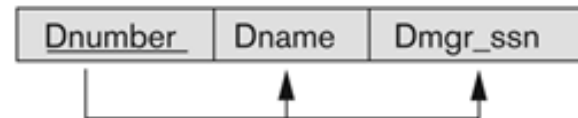


3NF Normalization

ED1



ED2



General Definitions of 2NF and 3NF

- Take all candidate keys of a relation into account
- 2NF: every nonprime attribute A in a relation schema R is not partially dependent on any key of R
- 3NF: it is in 2NF and every nonprime attribute A is not transitively dependent on any key of R

BCNF (Boyce-Codd Normal Form)

- BCNF: if every determinant is a superkey of R
- Each normal form is strictly stronger than the previous one
 - Every 2NF relation is in 1NF
 - Every 3NF relation is in 2NF
 - Every BCNF relation is in 3NF

BCNF (Cont.)

- Test BCNF: identify all the determinants and make sure that they are the superkeys
- For a functional dependency $A \rightarrow B$:
 - 3NF allows this dependency in a relation if B is a prime attribute and A is not a superkey
 - BCNF insists that A must be a superkey for this dependency to remain in a relation
- BCNF is a stronger form of 3NF
 - Every relation in BCNF is also in 3NF
 - A relation in 3NF not necessarily in BCNF



If a relation is in BCNF, is it free of redundancies that can be detected using FDs?

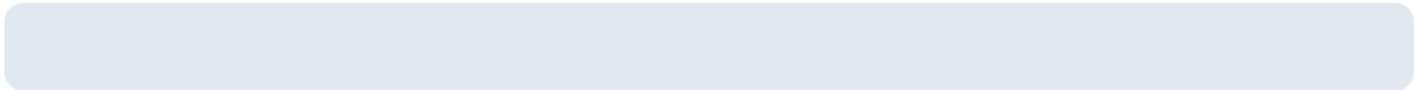
Yes

No



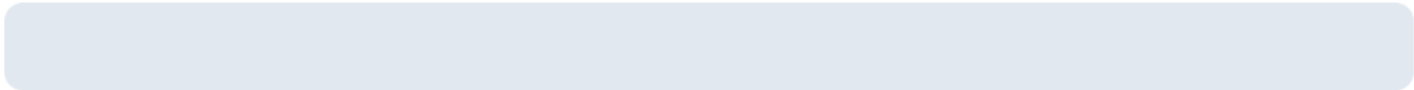
If a relation is in BCNF, is it free of redundancies that can be detected using FDs?

Yes



0%

No

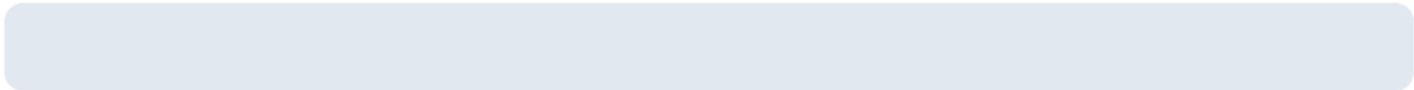


0%



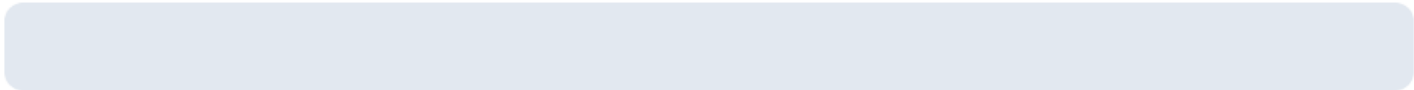
If a relation is in BCNF, is it free of redundancies that can be detected using FDs?

Yes



0%

No



0%

Summary

- 1NF: removes repeating groups
- 2NF: removes partial dependencies on any key
- 3NF: removes transitive dependencies on any key
- BCNF: removes remaining anomalies



Does a relational schema that satisfies the conditions of a higher normal form also satisfy the conditions of a lower one?

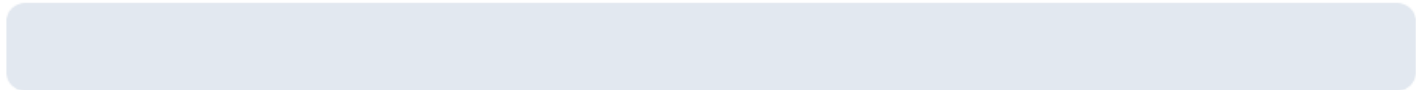
Yes

No



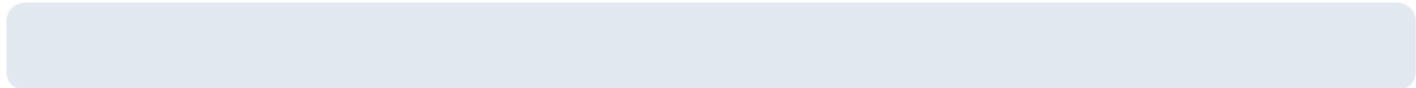
Does a relational schema that satisfies the conditions of a higher normal form also satisfy the conditions of a lower one?

Yes



0%

No

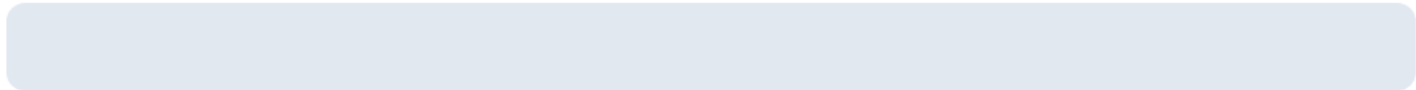


0%



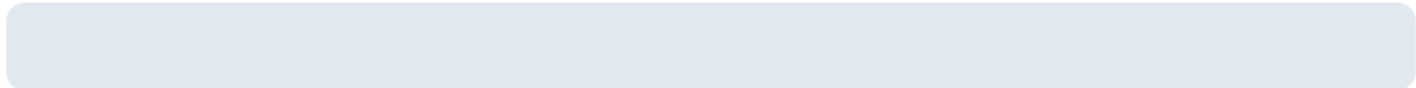
Does a relational schema that satisfies the conditions of a higher normal form also satisfy the conditions of a lower one?

Yes



0%

No



0%

Exercise 1

- Lots(Property_id#, County_name, Lot#, Area, Price, Tax_rate)
- FDs:
 - Primary key: Property_id#
 - Candidate key: {County_name, Lot#}
 - County_name \rightarrow Tax_rate
 - Area \rightarrow Price

Exercise 2

- (Property#, Idate, Itime, Paddress, Comments, Staff#, Sname, Car_reg)
- FDs:
 - Property# \rightarrow Paddress
 - Staff# \rightarrow Sname
 - {Idate, Staff#} \rightarrow Car_reg
 - Candidate key: {Idate, Itime, Car_reg}