

A decorative graphic on the left side of the slide, consisting of a network of white lines and circles on a blue gradient background, resembling a circuit board or a neural network.

WEEK 9

NORMALIZATION AND FIRST NORMAL FORM

CS3319

STUDENT OBJECTIVES

- Upon completion of this video, you should be able to:
 - Identify the goal of normalization
 - Define the terms *“Candidate Key”*, *“Primary Key”*, *“Secondary Key”*, *“Prime Attribute”* and *“NonPrime Attribute”*
 - Given a table that is not normalized, put it into first normal form using one of 3 methods

normalization is the process of removing redundant data.

NORMALIZATION

- Normalization is the process of removing redundant data to entities

<u>SSN</u>	<u>PNUMBER</u>	<u>HOURS</u>
1	A1	5

- Normalizing a table to remove redundancy

<u>VehicleID</u>	<u>LicencePlate</u>	Model	Colour	Price	Manufacture
8790	AWLK 310	Civic	Navy	25,000	Honda
6545	BBEM 221	Corolla	Black	29,000	Toyota

- Some terminology: each is called a **Candidate key** for the relation. The minimal key, is chosen as the **Primary key** and the others are called **Secondary keys**.

3	A1	12
3	B1	33
4	B1	23
4	C1	45
4	D1	23

All candidate keys

- An attribute of R is a **prime attribute** of R if it is a member of any key of R, otherwise it is a **nonprime attribute**.

Candidate key: unique key.

Secondary key: candidate keys that are not primary key.

11/15/23

- In the ProjEmp (Works_On) table, SSN and Pnumber are **prime attributes** and all the rest (Hours) are nonprime.
- Normalization works through a series of stages, you move from first normal form to second normal form to third normal form
- Normalization is not always desirable because generally the higher the normal form, the more pointer movements (or joins) are required to get output, thus sometimes after normalizing the database you may then **denormalize** some portions for performance reasons.

FIRST NORMAL FORM

- All the relations we have been building so far have already been in first normal form.
- It disallows multi-valued attributes, composite attributes and their combinations
- To be in first normal form: the domains and attributes must include only atomic values and the values of any attribute in a tuple must be a single value from the domain of that attribute

EXAMPLE:

- Consider the following table that is **NOT** in first normal form (DNumber is the key):

Dname	<u>Dnumber</u>	DMgrSSN	DLocation
Accounting	S7G	1 234	{London, Toronto, Hong Kong}
Payroll	P87	1 1 1 1	{London}
Warehouse	D65	1 233	{New York}

← this is
a multi-
value
attribute.

To put it into First Normal Form we have 3 options:

- **Option 1:** Split out the repeating values and using the repeating value as part of the key:

lose 1 keeping

Dname	<u>Dnumber</u>	DMgrSSN	<u>DLocation</u>
Accounting	S7G	1 234	London
Accounting	S7G	1 234	Toronto
Accounting	S7G	1 234	Hong Kong
Payroll	P87	1 1 1 1	London
Warehouse	D65	1 233	New York

- **Option 2:** Break it into 2 tables as follows:

1. Department Table

Dname	<u>Dnumber</u>	DMgrSSN
Accounting	S7G	1234
Payroll	P87	1111
Warehouse	D65	1233

2. DeptLocation Table

<u>Dnumber</u>	<u>Dlocation</u>
S7G	London
S7G	Toronto
S7G	Hong Kong
P87	London
D65	New York

- **Option 3:** If you have a small LIMITED number of values that are repeated, you could make new attributes for those values:

Dname	<u>Dnumber</u>	DMgrSSN	Dloc1	Dloc2	Dloc3
Accounting	S7G	1234	London	Toronto	Hong Kong
Payroll	P87	1111	London	Null	Null
Warehouse	D65	1233	New York	Null	Null

NOTE: Option 2 is normally the best choice!