Topics to be covered:

- Relational Model Concepts
- What is Key?
- Types of Keys.
- Determination & Functional Dependence

At the end of this chapter the students will be able to:

• Describe Relational Model Concepts
• Describe properties of a relation and relational keys

Relational Model

The relational model is today the primary data model for commercial data processing applications. It attained its primary position because of its simplicity, which eases the job of the programmer, compared to earlier data models such as the network model or the hierarchical model. In this, we first study the fundamentals of the relational model. A substantial theory exists for relational databases.

Relational Model Concepts

We shall represent a relation as a table with columns and rows. Each column of the table has a name, or attribute. Each row is called a tuple.

Relation:

•We shall represent a relation as a table with columns and rows.
•Each column of the table has a name, or attribute.
•Each row is called a tuple

• **Domain**: a set of atomic values that an attribute can take

• **Attribute**: name of a column in a particular table (all data is stored in tables).
Each attribute Ai must have a domain, dom(A i).

•**Relational Schema**: The design of one table, containing the name of the table (i.e. the name of the relation), and the names of all the columns, or attributes.
Example: STUDENT(Name, SID, Age, GPA)

dept_name	*building*	*budget*
Biology	Watson	90000
Comp. Sci.	Taylor	100000
Elec. Eng.	Taylor	85000
Finance	Painter	120000
History	Painter	50000
Music	Packard	80000
Physics	Watson	70000

Figure: The department relation

Consider the department relation of Figure. The schema for that relation is department (dept name, building, budget)

• **Degree of a Relation**: the number of attributes in the relation's schema.

• **Tuple**, t, of R(A1, A2, A3, ..., An): an ORDERED set of values, < v1, v2, v3,..., vn>, where each v i is a value from dom(A i).

• **Relation Instance**, r(R): a set of tuples; thus, r(R) = { t1, t2, t3, ..., tm}

NOTES:

1) The tuples in an instance of a relation are not considered to be ordered putting the rows in a different sequence does not change the table.
2) Once the schema, R(A1, A2, A3, ..., An) is defined, the values, vi, in each tuple, t, must be ordered as t = <v1, v2, v3, ..., vn>

Properties of database relations are:
• relation name is distinct from all other relations
• each cell of relation contains exactly one atomic (single) value
• each attribute has a distinct name
• values of an attribute are all from the same domain
• order of attributes has no significance
• each tuple is distinct; there are no duplicate tuples
• order of tuples has no significance, theoretically.

What is Key?

Key is a set of one or more columns whose combined values are unique among all the occurrences of the given table. A key is a relational means of specifying uniqueness.

Significance of Key Attributes

- Key attributes in a table are of strong significance in database design process.
- Realization of key attributes in a data environment let the designer come up with an efficient and error-free design.

Concept of Determination & Functional Dependence
Identifying key attributes essentially requires the understanding of the following relevant concepts:

Determination:

STUDENT

STU_NUM	STU_LNAME	STU_FNAME	STU_INIT	STU_PHONE	STU_HRS	STU_CLASS	STU_GPA
321452	Bowser	William	C	2134	66	So	2.11
324257	Smith	Anne	K	2256	81	Jr	3.27
324258	Bowser	John	H	2256	36	So	2.26
324269	Oblonski	Walter	D	2114	66	Jr	3.09
324273	Smith	John	P	2231	81	Sr	2.11

Figure 3.1: Table demonstrating the concept of Determination

- Attribute A determines attribute B (A → B) means if you know the value of A, you can determine the corresponding value of B.
- In other words, in a table having many records if A → B then for each particular value of A one should always find the same value for B.
- In Figure 3.1, a deep look over the column values reveals the following facts:

1.	STU_LNAME doesn't determine STU_NUM	STU_LNAME's value 'Bowser' is appearing with two different STU_NUM values (321452 & 324258).
2.	STU_FNAME doesn't determine STU_NUM	STU_FNAME's value 'John' is appearing with two different STU_NUM values (324258 & 321273).
3.	STU_NUM determines STU_LNAME (**STU_NUM → STU_LNAME**)	For a particular value of STU_NUM, always the same value appear in STU_LNAME. For example, whenever 321452 appears in the table, it always comes with Bowser in STU_LNAME. In fact, only a single occurrence for '321452' is given in the example. But it is obvious that even if the value comes again, it will always come with 'Bowser'.

Functional Dependence

INVOICE_DETAIL

INV_NUMBER	LINE_NUMBER	PROD_CODE	LINE_UNITS	LINE_PRICE
1001	1	123-21UUY	1	$189.99
1001	2	SRE-657UG	3	$2.99
1002	1	123-21UUY	2	$18.63
1003	1	ZZX/3245Q	1	$6.79
1003	2	SRE-657UG	1	$2.99
1003	3	001278-AB	1	$12.95

Figure 3.2: Table demonstrating the concept Full Functional Dependence

Following can be seen in the table's data shown in Figure3.2.

1. INV_NUMBER doesn't determine PROD_CODE	INV_NUMBER value '1001' appears with more than one value of PROD_CODE (123-21UUY & SRE-657UG)
2. LINE_NUMBER doesn't determine PROD-CODE	LINE_NUMBER value '1' appears with more than one values of PROD_CODE (123-21UUY & ZZX/3245Q)
3. A combination of INV_NUMBER & LINE_NUMBER determines PROD_CODE	Any particular value combination for INV_NUMBER & LINE_NUMBER (say, 1001 & 1) appears only with a single value of PROD_CODE (123-21UUY)
Thus, PROD_CODE is having a full functional dependence on (INV_NUMBER, LINE_NUMBER)	

Types of Keys in Relational Database Model

1. Superkey is an attribute or a composite attribute which functionally determines all of the entity's attributes. In other words, a super key uniquely identifies each entity in a table.
Considering the example given in Figure3.1, any of the following can be identified as the table's superkey:

1. STU_NUM	As it determines all other attributes in the table.
2. STU_NUM, STU_LNAME	As the combination also determines all remaining attributes.
3. STU_NUM, STU_LNAME, STU_FNAME	As this combination also determines all remaining attributes.
In fact, STU_NUM whenever consider either alone or in any possible combination with other attributes, realizes a super key. This is even true if the additional attributes are having redundant values in the table's record.	

2. Candidate Key is a super key whose values are not repeated in the table records. In other words, when the values in a super key are not repeated in the table's records, then such a key is called a candidate key. Considering the example in Figure 3.1, following facts can be revealed

1. The superkey attribute STU_NUM can also be termed as a candidate key because the values for STU_NUM are not redundant in the given example

2. The composite superkey (STU_NUM, STU_LNAME) cannot be considered as a candidate key because STU_NUM by itself is a candidate key

3. The combination (STU_LNAME, STU_FNAME, STU_INIT, STU_PHONE) can also be considered as a candidate key provided that the values under the combination are not be repeated in the table's records.

3. Primary Key is a candidate key which doesn't have repeated values nor does it comes with a NULL value in the table. A primary key can uniquely identifies each row in any table, thus a primary key is mainly utilized for record searching.

- A primary key in any table is both a superkey as well as a candidate key.

- It is possible to have more than one choice of candidate key in a particular table example. In that case, the selection of the primary key would be driven by the designer's choice or by end user requirements.

4. Secondary Key, like Primary Key doesn't fulfill the property of unique record searching. Nevertheless, a secondary key is used occasionally to narrow down the searching of particular records in a table.

- The favorable feature of the key is 'easier-to-remember' as compared with the primary key values.

- In Figure 3.1, searching with STU_LNAME or STU_PHONE will result in a group of student records. The searching of a particular student will then be made a simpler job by narrowing the search within this result.

- The choice of secondary key should be made with some care. Otherwise the search could not be narrowed properly. In Figure 3.1, STU_CLASS is not a good choice for a secondary key search because it will result in a large size record group to be returned, thus failing the idea of providing ease of search.

5. Unique key: In relational database design, a **unique key** or **primary key** is a candidate key to uniquely identify each row in a table. A unique key or primary key comprises a single column or set of columns. No two distinct rows in a table can have the same value (or combination of values) in those columns. Depending on its design, a table may have arbitrarily many unique keys but at most one primary key.
A primary key is a special case of unique keys. The major difference is that for unique keys the implicit NOT NULL constraint is not automatically enforced, while for primary keys it is. Thus, the values in a unique key column may or may not be NULL. Another difference is that primary keys must be defined using another syntax.

6. Foreign Key is a table's primary key attribute which is repeated in another related table (having related data) to maintain the required data relationship.
In Figure 3.3, the primary key attribute of table VENDOR is VEND_CODE. In order to make record linking between the tables, VEND_CODE is being used as a foreign key in the table PRODUCT.

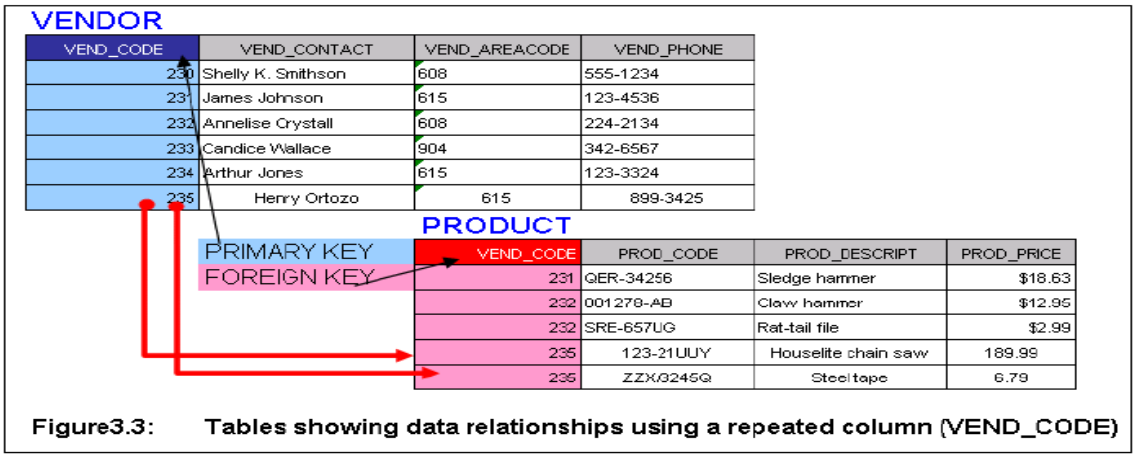

Figure3.3: Tables showing data relationships using a repeated column (VEND_CODE)

7. Alternate Key: The alternate keys of any table are simply those candidate keys, which are not currently selected as a primary key. Exactly one of those candidate key is selected as a primary key and the remainder, if any, are then alternate keys. An alternate key is a function of all the candidate keys minus the primary key.

8. Composite Keys: Sometimes it requires more than one attribute to uniquely identify the entity. A primary key that is made up of more than one attribute is known as composite key.

Summary

- Super key:- Set of an attribute which can uniquely identify a tuple
- Primary key :- The attribute or combination of attributes that uniquely identifies a row or record.
- Unique key:- ensures that all values in a column are different.
- Foreign Key:- an attribute or combination of attribute in a table whose value match a primary key in another table.
- Composite key:- A primary key that consists of two or more attributes is known as composite key.
- Candidate key:- is a column in a table which has the ability to become a primary key.
- Alternate Key:- Any of the candidate keys that is not part of the primary key is called an alternate key.
- Secondary key:- alternate of primary key.

Long Answer Type Questions:

1. Explain the terms: Relation, Tuple, Domain, Relational schema, Attribute, Degree and Cardinality of relation.
2. What is key? Explain various types of keys in relational data model.
3. Explain following terms with suitable example :-
   (i) Primary Key (ii) Candidate Key (iii) Foreign Key (iv) Super Key