MIS 301 RELATIONAL DATABASE MANAGEMENT SYSTEM

DATABASE MANAGEMENT SYSTEM

Concepts of tables, records, attributes, keys, integrity constraints, data independence

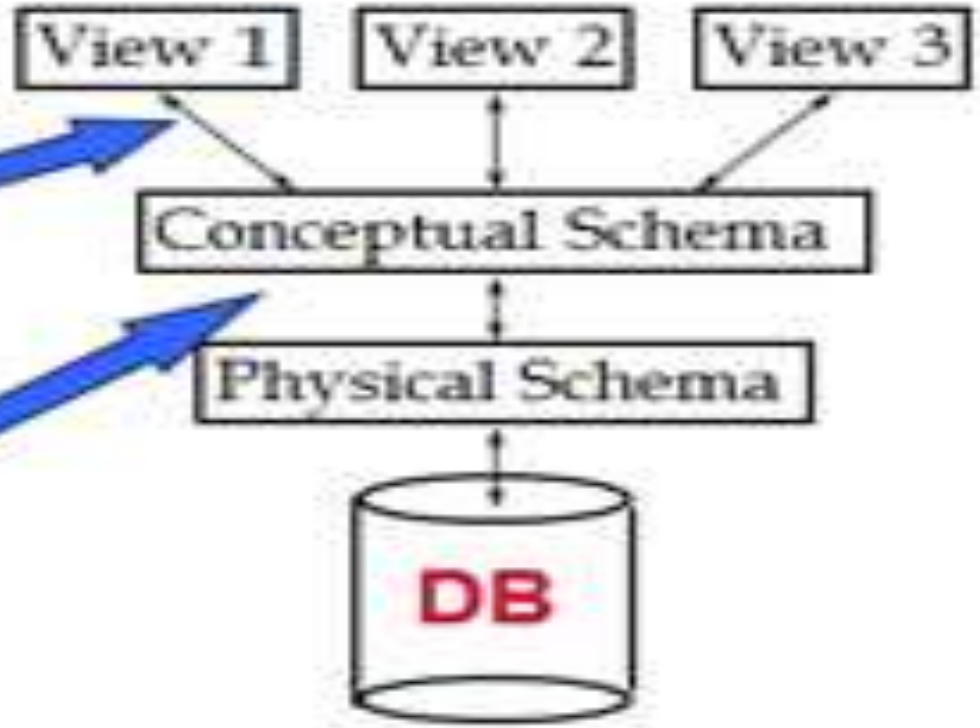
LECTURES 2&3

THREE LEVELS OF DATA ABSTRACTION

- **Physical**: This is the lowest level of data abstraction. It tells us how the data is actually stored in memory. The access methods like sequential or random access and file organisation methods like B+ trees, hashing are used for the same. Suppose we need to store the details of an employee. Blocks of storage and the amount of memory used for these purposes is kept hidden from the user.
- **Logical**: This level comprises of the information that is actually stored in the database in the form of tables. It also stores the relationship among the data entities in relatively simple structures. At this level, the information available to the user at the view level is unknown.
  We can store the various attributes of an employee and relationships, e.g. with the manager can also be stored.
- **View**: This is the highest level of abstraction. Only a part of the actual database is viewed by the users. This level exists to ease the accessibility of the database by an individual user. Users view data in the form of rows and columns. Tables and relations are used to store data. Multiple views of the same database may exist. Users can just view the data and interact with the database, storage and implementation details are hidden from them.

THREE LEVELS OF DATA ABSTRACTION

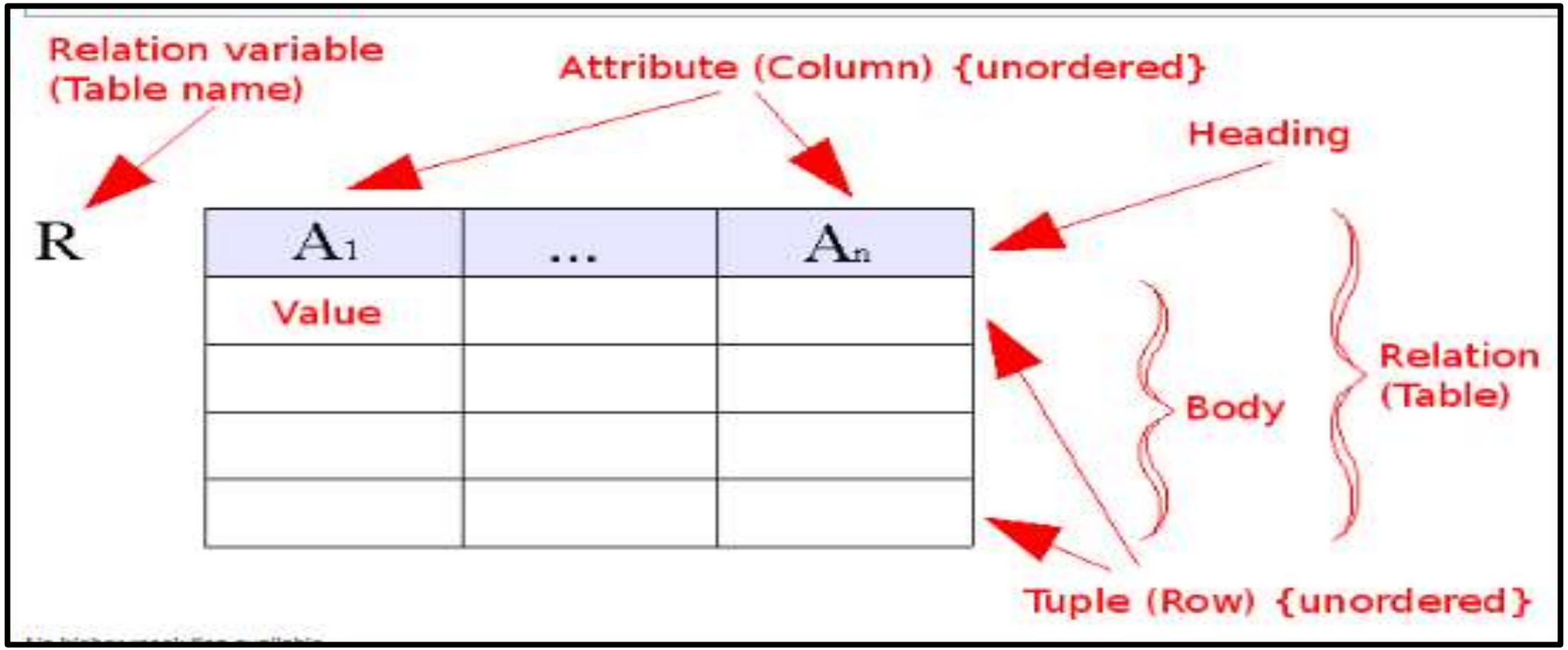
- A Simple Idea: Applications should be insulated from how data is structured and stored.
- Logical data independence: Protection from changes in *logical* structure of data.
- Physical data independence: Protection from changes in *physical* structure of data.



DATA INDEPENDENCE

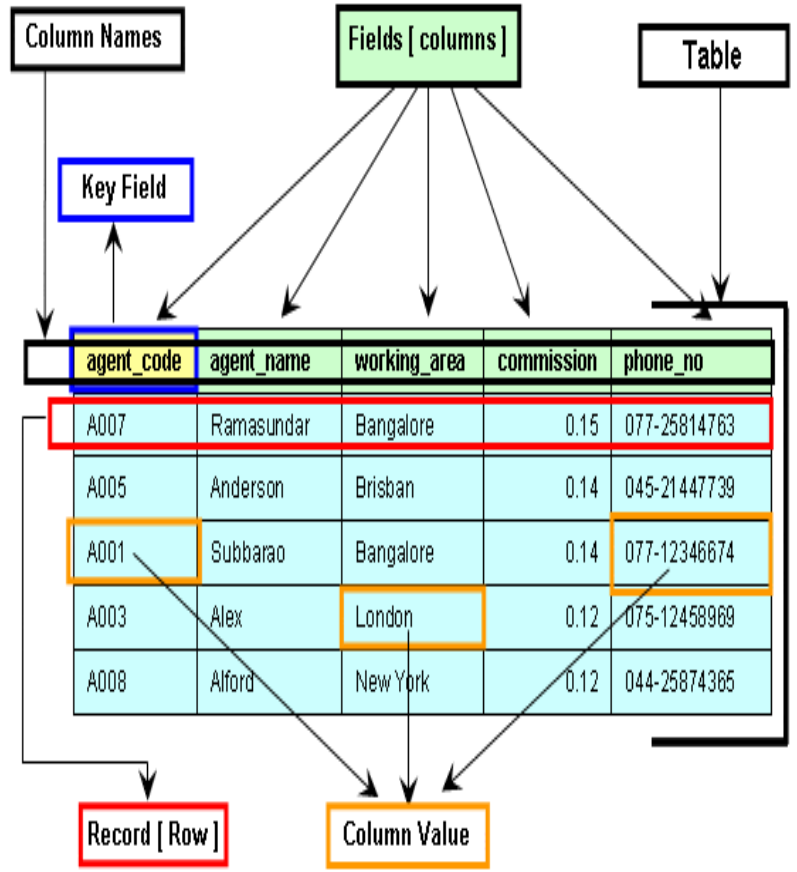
- **Physical Data Independence :**
  Physical Data Independence is defined as the ability to make changes in the structure of the lowest level of the Database Management System (DBMS) without affecting the higher-level schemas. Hence, modification in the Physical level should not result in any changes in the Logical or View levels.
- **Logical Data Independence :**
  Logical Data Independence is defined as the ability to make changes in the structure of the middle level of the Database Management System (DBMS) without affecting the highest-level schema or application programs. Hence, modification in the logical level should not result in any changes in the view levels or application programs.

CONCEPTS OF TABLES, RECORDS, ATTRIBUTES, KEYS

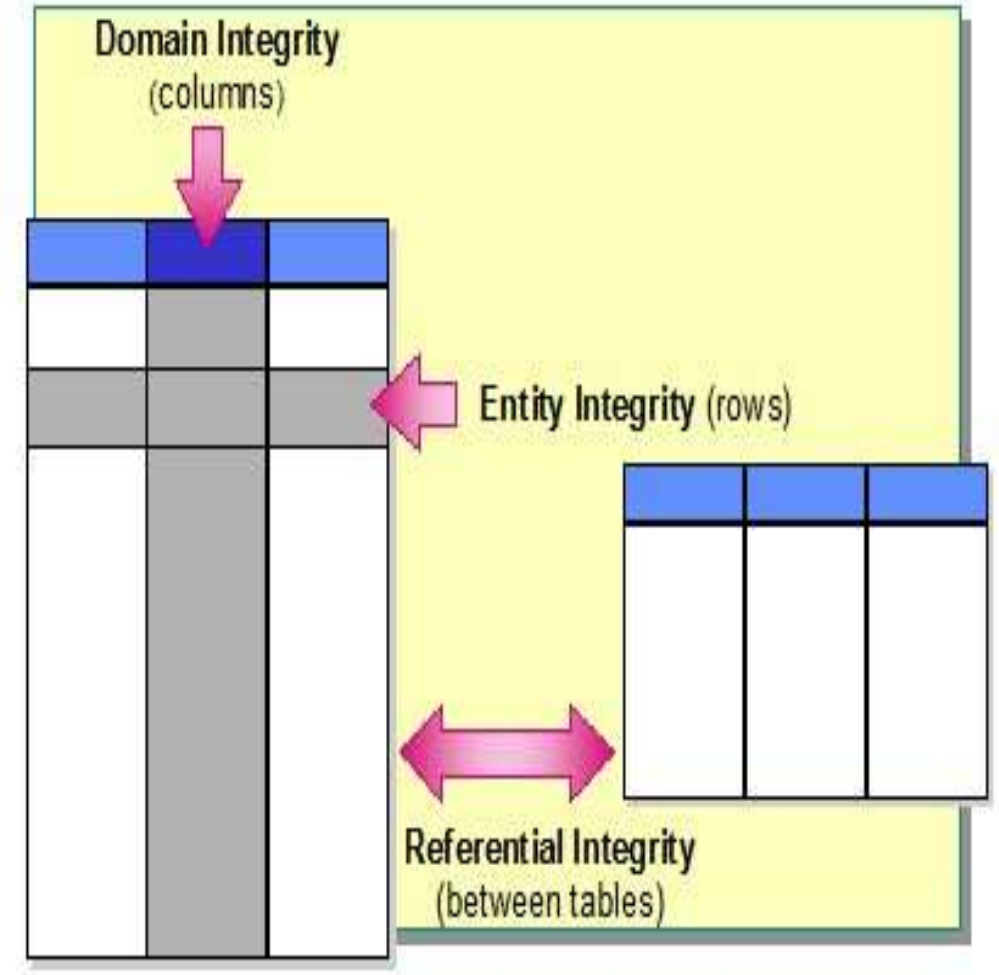


RELATIONS

- A **table or relation** is a collection of data elements for an entity set organized in terms of rows and columns.
- A single entry in a table is called a **Tuple** or **Record** or **Row**. It is represented by a row in a relation.
- An **attribute** is a specific characteristic taken up by each entity in a particular column of the relation
- The attribute description is also called a **fieldname.**
- The value taken up by an attribute in a tuple is called the **field value.**
- An attribute or combination of attributes that is never duplicated in a relation is called a **candidate key.**
- A candidate key chosen for unique identification of tuples in a relation is called a **primary key.**
- Candidate keys which are not primary keys are called **alternate keys.**
- An attribute which is not a primary key in the current relation but draws values from the domain shared by the primary key of some other relation in the same database is called a **foreign key.**



INTEGRITY CONSTRAINTS



attributes

column

SID	SName	SAge	SClass	SSection
1101	Alex	14	9	A
1102	Maria	15	9	A
1103	Maya	14	10	B
1104	Bob	14	9	A
1105	Newton	15	10	B

tuple

table (relation)

INTEGRITY CONSTRAINTS

Key Constraints or Entity Integrity

- There must be at least one minimal subset of attributes in the relation, which can identify a tuple uniquely. This minimal subset of attributes is called **key** for that relation. If there are more than one such minimal subsets, these are called ***candidate keys***.
- Key constraints force that –
- in a relation with a key attribute, no two tuples can have identical values for key attributes.
- a key attribute can not have NULL values.
- Key constraints are also referred to as Entity Constraints.

EMPLOYEE

EMP_ID	EMP_NAME	SALARY
123	Jack	30000
142	Harry	60000
164	John	20000
	Jackson	27000

Not allowed as primary key can't contain a NULL value

Table EMP

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7329	SMITH	CEO		17-DEC-85	9,000.00		20
7499	ALLEN	VP-SALES	7329	20-FEB-90	7,500.00	100.00	30
7521	WARD	MANAGER	7499	22-FEB-90	5,000.00	200.00	30
7566	JONES	SALESMAN	7521	02-APR-90	2,975.00	400.00	30

NOT NULL Constraint
(no row may contain a null value for this column)

Absence of NOT NULL Constraint
(any row can contain a null for this column)

UNIQUE Key Constraint
(no row may duplicate a value in the constraint's column)

Table DEPT

DEPNO	DNAME	LOC
20	RESEARCH	DALLAS
30	SALES	NEW
40	MARKETING	BOSTON

INSERT INTO

50	SALES	NEW YORK
60		BOSTON

This row violates the UNIQUE key constraint, because 'SALES' is already present in another row; therefore, it is not allowed in the table.

This row is allowed because a null value is entered for the DNAME column; however, if a NOT NULL constraint is also defined on the DNAME column, this row is not allowed.

INTEGRITY CONSTRAINTS

Domain Constraints

- Attributes have specific values in real-world scenario. For example, age can only be a positive integer. The same constraints have been tried to employ on the attributes of a relation. Every attribute is bound to have a specific range of values. For example, age cannot be less than zero and telephone numbers cannot contain a digit outside 0-9.

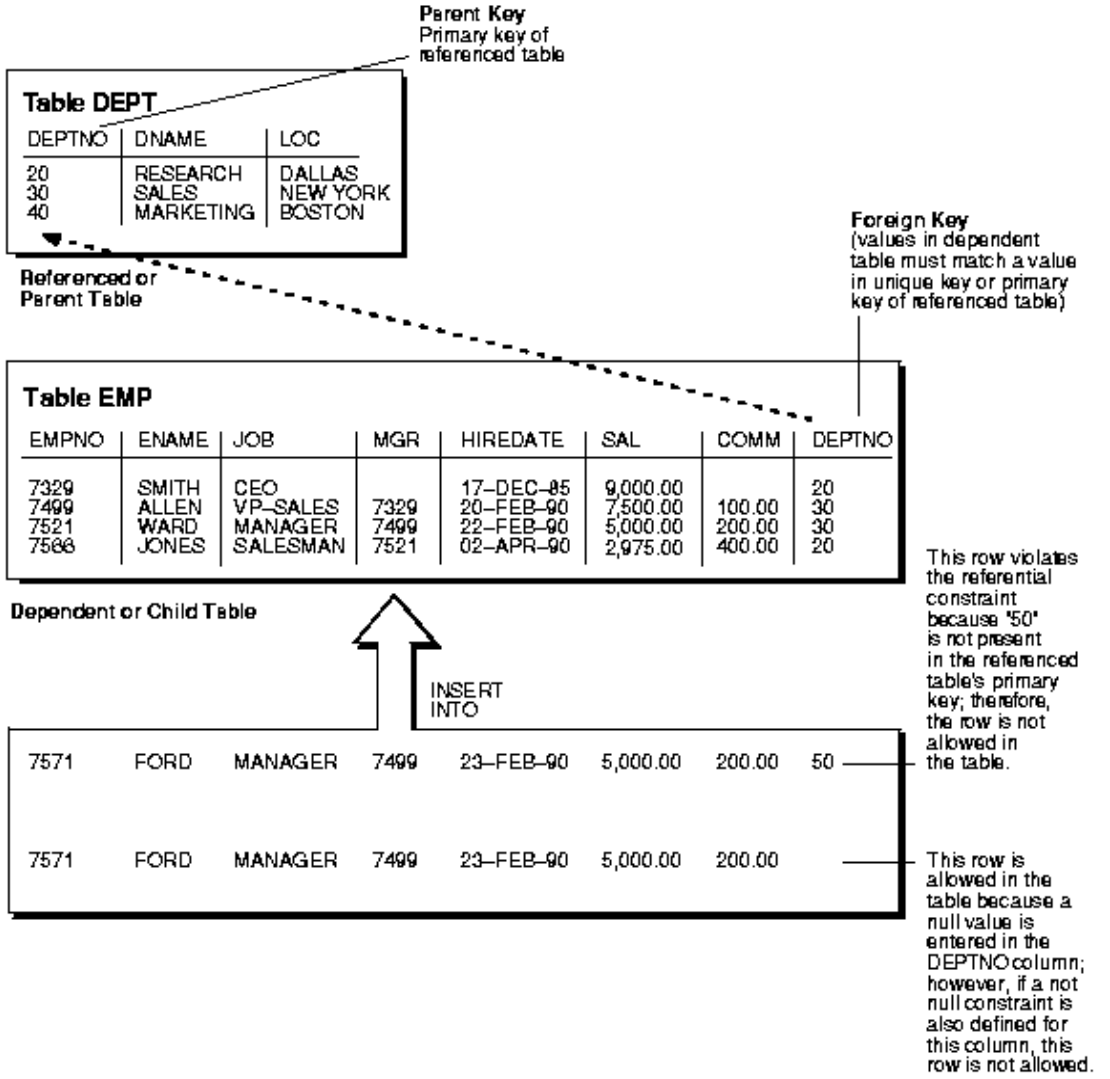
ID	NAME	SEMENSTER	AGE
1000	Tom	1st	17
1001	Johnson	2nd	24
1002	Leonardo	5th	21
1003	Kate	3rd	19
1004	Morgan	8th	A

Not allowed. Because AGE is an integer attribute

INTEGRITY CONSTRAINTS

Referential integrity Constraints

- Referential integrity constraints work on the concept of Foreign Keys. A foreign key is a key attribute of a relation that can be referred in other relation.
- Referential integrity constraint states that if a relation refers to a key attribute of a different or same relation, then that key element must exist.



- TILL WE MEET AGAIN IN THE NEXT CLASS.....