Null is the special value given to unknown, undefined or non-applicable fields (columns) in a tuple (row)

The entity is an "object", or "thing" in the real world that is differentiable from all other objects

Functions of Database Management System (DBMS):

A) DBMS are responsible for Data definition and the Updating of such data

B) Front-End Design is not a function of DBMS. A DBMS is responsible for querying the data from a database, but it does not responsible for converting such data into graphics; this feature can be done with front end technologies such as HTML, CSS and JavaScript in the case of web development .

C) DBMS are responsible for providing information in a form directly usable or for further processing by other applications

D) DBMS are responsible for administration of databases

Advantages of DBMS:

Data abstraction, reliability and efficiency

Advantages of DBMS:

Cost up, and over-kill when the database has simple structure, for simple application and multiple user access is not required

Which one of the following attributes can be null?

1. Entity’s primary key
2. Entity’s foreign key
3. Weak entity’s partial key
4. Participant in a relationship

A - Entities can only be uniquely identified if they have a valid primary key.

B - A foreign key can be null if the parent table does not contain a relevant tuple.

C - Similar to A, since a weak entity's partial key is part of its unique identifier.

D - A relationship between entities is not well-defined if one or more of those entities is null.

In spite of the advantages of using a DBMS, there are a few situations in which a

DBMS may involve unnecessary overhead costs that would not be incurred in

traditional file processing. The overhead costs of using a DBMS are due to the

following:

■ High initial investment in hardware, software, and training

■ The generality that a DBMS provides for defining and processing data

■ Overhead for providing security, concurrency control, recovery, and integrity

functions

Therefore, it may be more desirable to develop customized database applications

under the following circumstances:

■ Simple, well-defined database applications that are not expected to change

at all

■ Stringent, real-time requirements for some application programs that may

not be met because of DBMS overhead

■ Embedded systems with limited storage capacity, where a general-purpose

DBMS would not fit

■ No multiple-user access to data