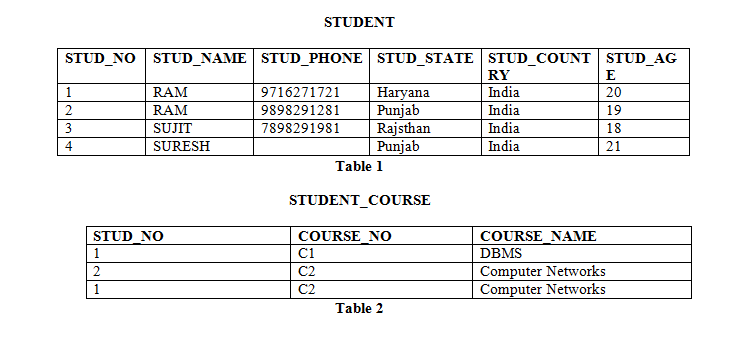
DBMS | Keys in Relational Model (Candidate, Super, Primary, Alternate and Foreign)

We strongly recommend to refer below post as a prerequisite of this.

[DBMS | Relational Model Introduction and Codd Rules](http://quiz.geeksforgeeks.org/dbms-relational-model-introduction-and-codd-rules/)

**Different Types of Keys in Relational Model**



**Candidate Key:** The minimal set of attribute which can uniquely identify a tuple is known as candidate key. For Example, STUD\_NO in STUDENT relation.

* The value of Candidate Key is unique and non-null for every tuple.
* There can be more than one candidate key in a relation. For Example, STUD\_NO as well as STUD\_PHONE both are candidate keys for relation STUDENT.
* The candidate key can be simple (having only one attribute) or composite as well. For Example, {STUD\_NO, COURSE\_NO} is a composite candidate key for relation STUDENT\_COURSE.

**Note –** In Sql Server a unique constraint that has a nullable column, **allows** the value ‘**null**‘ in that column **only once**. That’s why STUD\_PHONE attribute as candidate here, but can not be ‘null’ values in primary key attribute.

**Super Key:**The set of attributes which can uniquely identify a tuple is known as Super Key. For Example, STUD\_NO, (STUD\_NO, STUD\_NAME) etc.

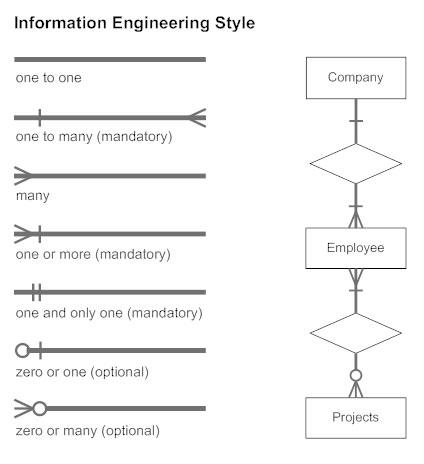
* Adding zero or more attributes to candidate key generates super key.
* A candidate key is a super key but vice versa is not true.

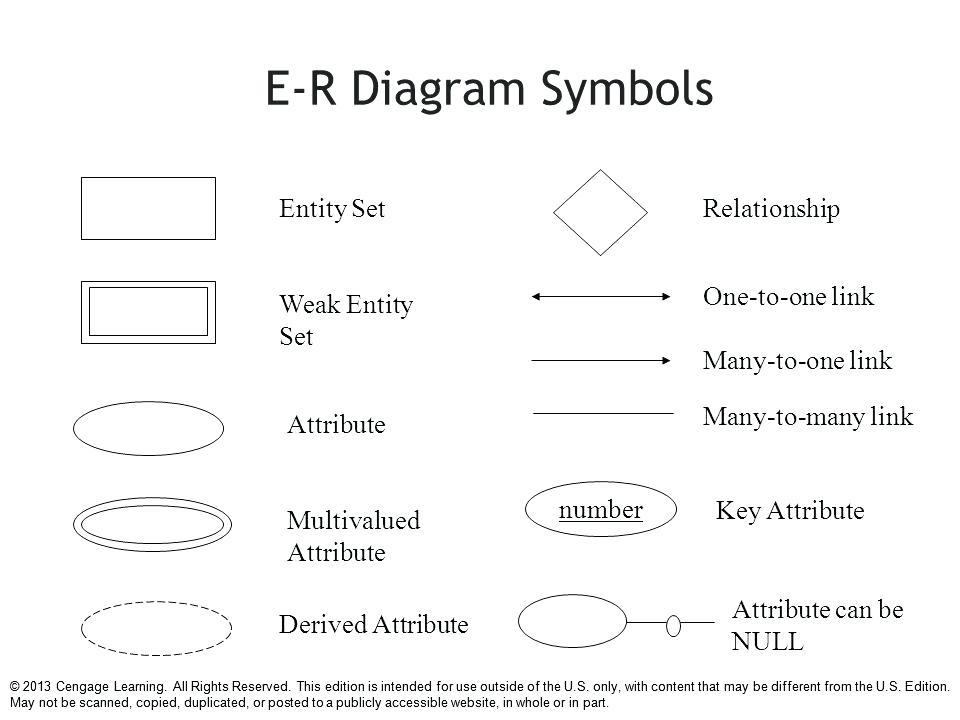
**Primary Key:** There can be more than one candidate key in a relation out of which one can be chosen as primary key. For Example, STUD\_NO as well as STUD\_PHONE both are candidate keys for relation STUDENT but STUD\_NO can be chosen as primary key (only one out of many candidate keys).

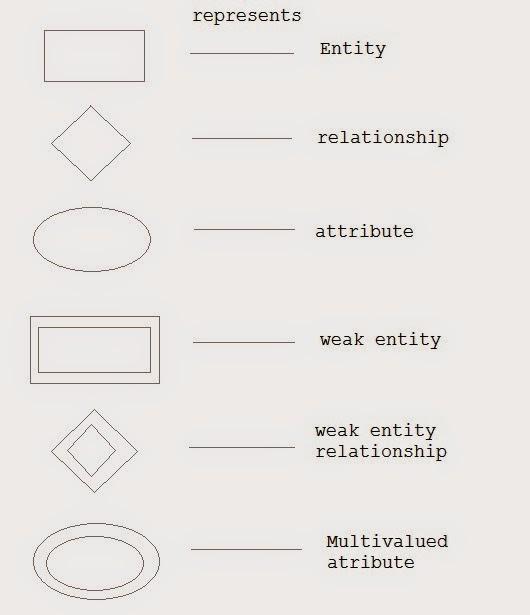
**Alternate Key:**The candidate key other than primary key is called as alternate key. For Example, STUD\_NO as well as STUD\_PHONE both are candidate keys for relation STUDENT but STUD\_PHONE will be alternate key (only one out of many candidate keys).

**Foreign Key:** If an attribute can only take the values which are present as values of some other attribute, it will be foreign key to the attribute to which it refers. The relation which is being referenced is called referenced relation and corresponding attribute is called referenced attribute and the relation which refers to referenced relation is called referencing relation and corresponding attribute is called referencing attribute. Referenced attribute of referencing attribute should be primary key. For Example, STUD\_NO in STUDENT\_COURSE is a foreign key to STUD\_NO in STUDENT relation.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above







| Star schema vs. Snowflake Schema | | |
| --- | --- | --- |
|  | **Star Schema** | **Snowflake Schema** |
| **Understandability** | Easier for business users and analysts to query data. | Maybe more difficult for business users and analysts due to a number of tables they have to deal with. |
| **Dimension table** | Only has one dimension table for each dimension that groups related attributes. Dimension tables are not in the third normal form. | May have more than 1 dimension table for each dimension due to the further normalization of each dimension table.  Dimension tables are in the third normal form (3NF). |
| **Query complexity** | The query is very simple and easy to understand | More complex query due to multiple foreign keys joins between dimension tables |
| **Query performance** | High performance. The database engine can optimize and boost the query performance based on a predictable framework. | More foreign key joins, therefore, longer execution time of query in compare with star schema |
| **When to use** | When dimension tables store a relatively small number of rows, space is not a big issue we can use star schema. | When dimension tables store a large number of rows with redundancy data and space is such an issue, we can choose snowflake schema to save space. |
| **Foreign Key Joins** | Fewer Joins | Higher number of joins |
| **Data warehouse system** | Work best in any data warehouse/data mart | Better for small data warehouse/ data mart |