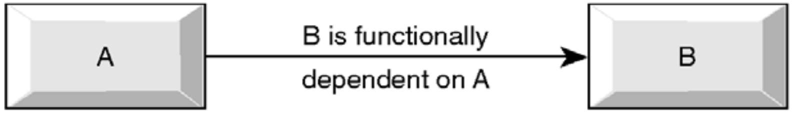
Functional Dependency

Objectives

- Definition of FD +
- Inference Rules for Functional Dependency (FD)
- Types of FDs +

- Definition of FD ...

- Describes the relationship between attributes in a relation.
- For example, if *A* and *B* are attributes of relation *R*, *B* is functionally dependent on *A* (denoted *A* → *B*), if each value of *A* in *R* is associated with exactly one value of *B* in *R*.
- Functional dependency is a property of the meaning or semantics of the attributes in a relation.
- The *determinant* of a functional dependency refers to the attribute or group of attributes on the left-hand side of the arrow.
- Diagrammatic representation.



... - Definition of FD ...

EMP_PROJ	SSN	Pnumber	Hours	Ename	Pname	Plocation

- From the semantics of the attributes, we know the following functional dependency should hold
  - SSN → Ename
  - Pnumber → {Pname, Location}
  - {SSN, Pnumber} → Hours

... - Definition of FD ...

- A functional dependency is a property of the relation schema (intension) R, not of a particular legal relation state (extension) of R.
- The figure below show a particular state of the TEACH relation.

TEACH

Teacher	Course	Text
Adel	Databases	Al-Masri
Adel	Data Structures	Al-Nour
Hani	Operating Systems	Khan
Baker	Java	Ahmed

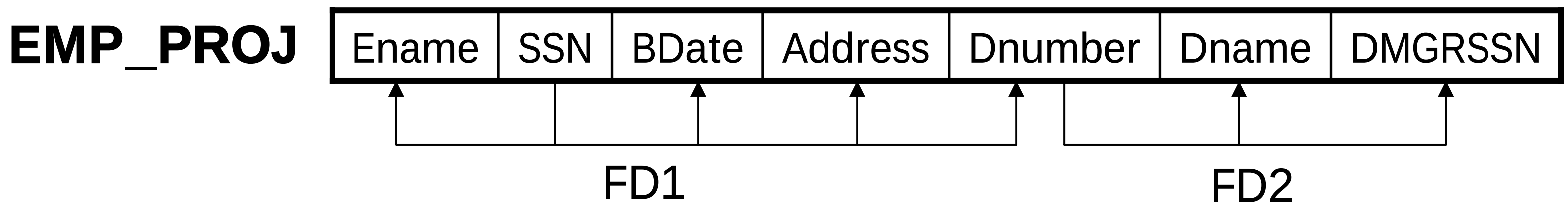
From the above state of TEACH we may conclude that Text → Course, we can not confirm this unless we know that it is true for all possible legal states of TEACH.

... - Definition of FD ...

- The set of FDs for a relation schema R are denoted by F.
- There could be other FDs that can be inferred from R.
- The set of all FDs (specified or deduced) is called the closure of F and is denoted by F^+.

- Inference Rules for FDs ...

Consider the relation schema EMP_PROJ in the figure below:



- We can infer the following addition FDs from F:
  - FD1 = {SSN → {Ename, BDate, Address, Dnumber}}
  - FD2 = {Dnumber → {Dname, DMGRSSN}}
  - FD3 = {SSN → {Dname, DMGRSSN}}
  - FD4 = {SSN → SSN}
  - FD5 = {Dnumber → Dname}

... - Inference Rules for FDs ...

- An FD $X \rightarrow Y$ is inferred from a set of dependencies F specified on R if $X \rightarrow Y$ holds in every relation state r that is a legal extension of R; that is whenever r satisfies all the dependencies in F, $X \rightarrow Y$ also holds in r.
- The closure of F is the set of all functional dependencies that can be inferred from F.
- A set of inference rules that can be used to infer new dependencies from a given set of dependencies.

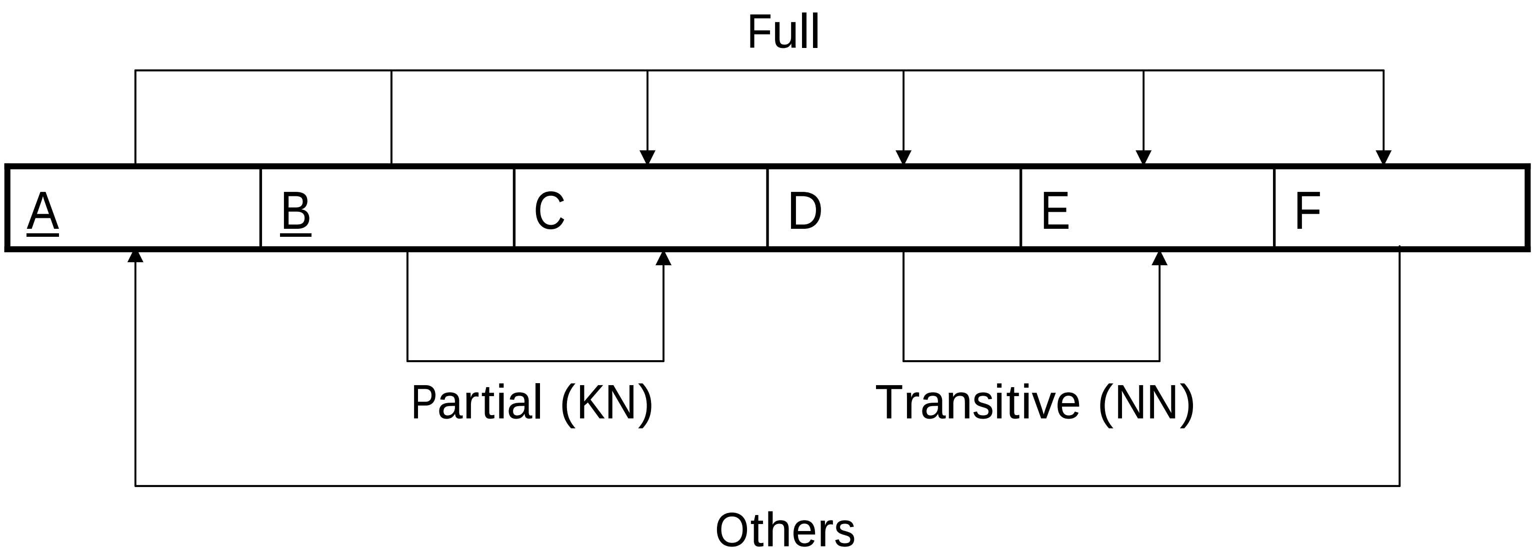
... - Inference Rules for FDs

- The following six rules (IR1 through IR6) are well-known inference rules for FDs.
  1. IR1 – Reflective rule: $Y \subseteq X$ then $X \rightarrow Y$.
  2. IR2 – Augmentation rule: $X \rightarrow Y$, then $XZ \rightarrow YZ$.
  3. IR3 – Transitive rule: $\{X \rightarrow Y, Y \rightarrow Z\}$ then $X \rightarrow Z$
  4. IR4 – Decomposition or projective, rule: $\{X \rightarrow YZ\}$ then $X \rightarrow Z$
  5. IR5 – Union or additive, rule: $\{X \rightarrow Y, X \rightarrow Z\}$ then $X \rightarrow YZ$
  6. IR6 – Pseudo transitive rule: $\{X \rightarrow Y, WY \rightarrow Z\}$ then $WX \rightarrow Z$

- Types of FDs ...

- **Definition**: X is a **key attribute** if it is a member of the primary key attributes otherwise it is a non-key attribute.
- Types of FD:
  - FD X → Y is a **full FD** if you remove any attribute A from X the dependency doesn't hold any more. {X-A} → Y is no longer true.
  - FD X → Y is a **partial FD** if X is a key attributes and Y is not.
  - FD X → Y is a **transitive FD** if both X and Y are non-key attributes.
  - The last type of FD, X → Y, is when Y is a key attribute and X is not.

... - Types of FDs



The concept of FDs will form the basis for our next topic **Normalization**.