(DBMS)

Syllabus:-

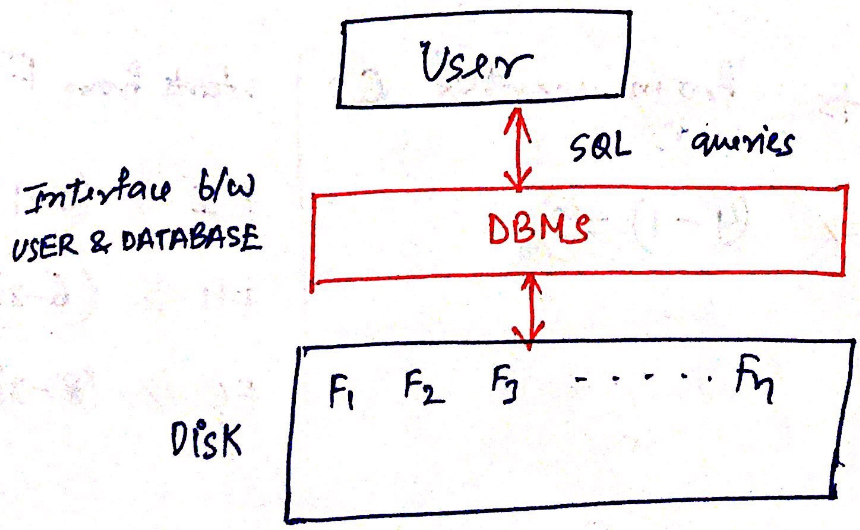
1. Integrity Constraints & ER Model
2. Normalization
3. Queries [RA, SQL, RC]
4. File organization && Indexing
5. Transaction && Concurrency control

Database:-

It is collection of related data. eg: set of Employee's info in a file

DBMS:-

Software used to manage & access database files in more efficient manner.

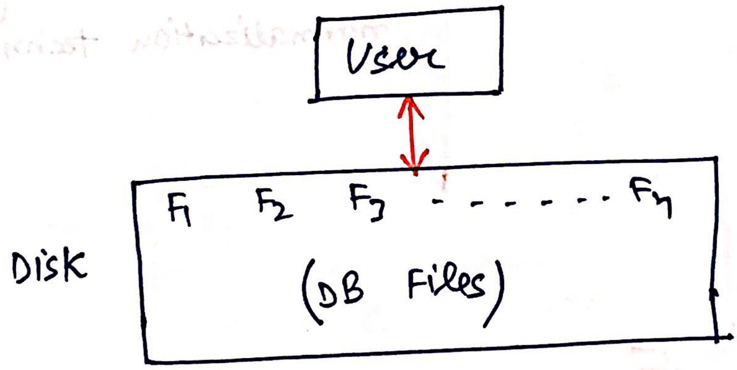


- DBMS provide SQL interface to user, so that user can fire its queries

- Storage (File structure) and other info .. related to its way of storing & accessing are governed by DBMS. These storage information are hidden to User. This is known as "data independence."

Flat File System :- (OS File system)

Database Files manages by user without DBMS.

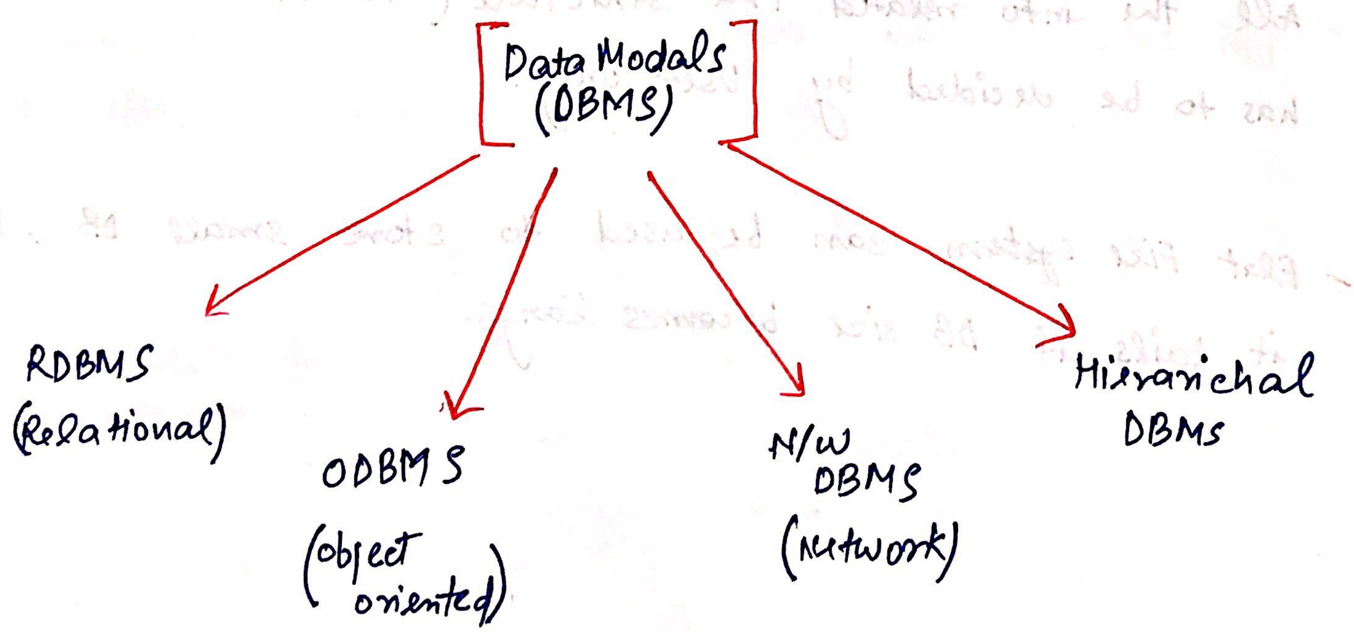


- All the info related File structure (Format of File & Data) has to be decided by User only.

- Flat File system can be used to store small DB, but it fails if DB size becomes large.

Limitation of Flat File System	Advantage of DBMS
- Development and management of application program too complex	- Because of data indipendency & SQL interface, development of application program is easier.
- More I/O Cost (Access cost) to access data	- Because of Indexing Less I/O Cost to access .data.
- Less degree of concurrency	- More degree of concurrency
- Complex to maintain non-redundant data	- Easy to maintain non-redundant data by using normalization techniques

Integrity Constraints :-



(i) Relational DBMS:-

- It is proposed by RJ-Codd (Codd's Data Modal)

- Codd proposed 12 rules to develop RDBMS Software. It is also called RDMS Guidelines.

⇒ **RDMS Guidelines:-**

1. Data in DB Files must be stored in Tabular format (Set of Row's & column's)

2. No two row's of RDBMS table should be same.

Student — Attribute or field

Sid	Sname	DOB
S_1	A	1995
S_2	C	1995
S_3	B	1994
S_4	A	1999

Record or tuple

Relational Instance

- Attributes or fields = Column of table
- Record or tuple = Entire row
- Relational Instance (Snapshot) = Set of Records

- **Relational schema** = structure / defination of RDMS table

eg: student(sid, sname, DOB)

- **Arity** = no. of attributes (columns) of RDBMS table

- **Cardinality** = no. of records (rows) of RDMS table.

- **Candidate Key** = Minimal set of attributes which can differentiate the records of relation uniquely.

eg:- 'Sid' is assigned as candidate Key.

'Sid | Sname' can't assigned as candidate key as sid can itself differentiate so it will not be a minimal set.

But sometime we need more than one attributes for candidate Key based on constraints.

eg:- Enroll (Sid Cid fee)

student id — course id — course fee

candidate key : sid / cid

Because one student can take more than one course and one course can be select by more than one student eg:- Some time in an employee table, more than one employee can have same A/c no. because may be they belong to different Branches of Banks.

EMP

eid	ename	DOB	PanID	Adhaar no	Ifsc	A/c no
e1	A		101134		SB101	101
e2	A		NULL		SB101	102
e3	B		102344		ICIC101	101
e4	C		NULL		ICIC101	102

NULL : Unknown / UnExisted value

NOT NULL : Constraint (NULL value not allowed)

Candidate Keys of EMP relation :-

$$\{ \underset{\text{primary key}}{\text{eid}}, \underbrace{\text{Pan id, Adhaar, IFSC A/c no}}_{\text{Altenate Key}} \}$$

more than 1 candidate keys

Primary key = Anyone candidate key of relational schema whose field value must not be NULL.

Primary Key Integrity constraints :

(i) Must be one of the candidate key

(ii) Field values can't be NULL

(iii) Atmost one Primary Key is allowed for relational schema

- **Alternate key** :- All candidate key's other than primary key.

= **Alternate Key Integrity constraints**:-

(i) Must be candidate Key
(ii) Field value can be NULL
(iii) More than one alternate key's are possible.

eg:

```
CREATE TABLE EMP
( eid      varchar(10)    PRIMARY KEY,
  ename    varchar(30)    NOT NULL,
  dob      date,
  PanID    varchar(9)     UNIQUE,
  Adhaar   integer(12)    UNIQUE, NOT NULL
  Ifsc     varchar(6),
  A/c no   integer(10),
  UNIQUE( IFSC, A/c no)
);
```

- **Super Key** :- Set of attributes of Relational schema which can differentiate records uniquely but may not be the minimal set of attributes.

Candidate Key's = { sid }

Super Key = { sid, sid sname, sid dob, sid sname dob }

So, candidate key is minimal superkey

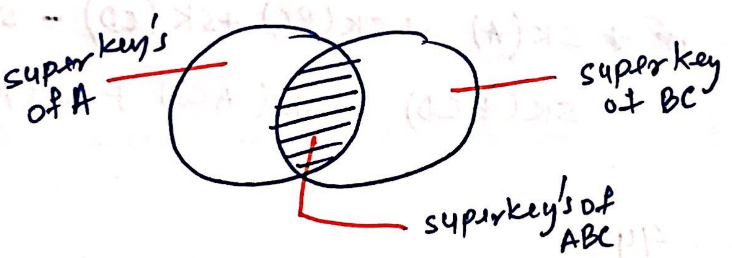
Q. Let R(A, B, C, D) be a relational schema and A is candidate key. Then how many superkey are possible.

$$\begin{array}{cccc} A & B & C & D \\ | & | & | & | \\ 1 & 0/1 & 0/1 & 0/1 \end{array}$$

$1 \times 2 \times 2 \times 2 = 2^3 = 8$ possible superkey's

Q. R(A, B, C, D, E)

(i) if {A, BC} candidate keys



So, Total (A ∪ BC) $= SK(A) + SK(BC) - SK(ABC)$

$= 2^4 + 2^3 - 2^2$

$= 20$

(ii) if (AD, BC) candidate keys

$$SK(AB \cup BC) = SK(AB) + SK(BC) - S(ABC)$$

$$= 2^3 + 2^3 - 2^2$$

$$= 12$$

(iii) if (AB, CD) cand keys

$$SK(AB \cup CD) = SK(AB) + SK(CD) - SK(ABCD)$$

$$= 2^3 + 2^3 - 2$$

$$= 14$$

Q. R(A, B, C, D, E, F) How many sk's are in R if

(i) A, B and C are candidate keys.

$$SK(A \cup B \cup C) = SK(A) + SK(B) + SK(C) - SK(AB) - SK(BC) - SK(AC) + SK(ABC)$$

$$= 2^5 + 2^5 + 2^5 - 2^4 - 2^4 - 2^4 + 2^2$$

$$= 56$$

(ii) {A, BC, CD} are candidate keys

$$SK(A \cup BC \cup CD) = SK(A) + SK(BC) + SK(CD) - SK(ABC) - SK(BCD) - SK(ACD) + SK(ABCD)$$

$$= 44$$

(iii) {AB, CD, E} candidate keys

$$SK(AB, CD, E) = SK(AB) + SK(CD) + SK(E) - SK(ABCD) - SK(CDE) - SK(ABE) + SK(ABCDE)$$

$$= 46$$

Q. $R(A,B,C,D,E)$, How many super key's in R if $\{AB, BC, CD, DE\}$ are candidate key (CK's).

$\underbrace{[AB], C, D, E}$	$\underbrace{A, [BC], D, E}$	$\underbrace{A, B [CD], E}$	$\underbrace{A, B, C, [DE]}$
AB	BC	CD	DE
ABC	BCD	ACD	ADE
ABD	BCE	CDE	BDE
ABE	BCDE	ACDE
ABCD		
ABCE		
ABDE		
ABCDE		

Q. $R(A_1, A_2, A_3, \ldots A_n)$ How many maximum SK's possible?

(a) 2^n (b) $n!$ (c) 2^{n+1} (d) $2^n - 1$ ✓

For maximum SK's, each attribute itself a CK.

- **Prime Attribute (Key attribute)** :- Attribute which belongs to some candidate key of relational schema.

Emp(eid, ename, dob, pan, adhaar, ifsc, A/c)

eg:- {eid, pan, adhaar, ifsc, A/c}

Non Prime Attribute :- Attributes which are not part of candidate key.

eg:- {ename, dob}

RDBMS Guidelines:-

3. Atleast one CK of relational schema whose field values must be NOT NULL.

eg:-

```
CREATE TABLE R
(
    A integer UNIQUE,
    B integer UNIQUE,
    C integer,
);
```

NOT RDBMS

{A and B are CK's but none of them are NOT NULL, so there can't be any PK}

A integer PRIMARY KEY | A integer UNIQUE, NOT NULL

Logically same but structurally different

PK has some more benefits than declaring UNIQUE NOT NULL

as:

PK → physical ordering & record file
→ default Index
→ Meta data is different

- **Foreign Key (Refrential Key)** :- It is set of attributes refrences to primary key or alternative key of same table (relation) or some other table.

- Used to relate data b/w the tables

- It is defined over two relations:

1) Refrenced Relation

2) Refrencing Relation

Refrenced Attr — Y — PK / AK

Refrencing Attr — X — foreign key

Refrenced Relation

Stud (Sid, Sname, dob)

Sid	Sname
S_1	A
S_2	A
S_3	B
S_4	B
S_5	B

Refreneind relation

Enroll(Sid, Cid, fee)

FK

Sid	Cid
S_1	C_1
S_1	C_2
S_2	C_2
S_4	C_1
~~S_{10}~~	~~C_1~~

Not Allowed as S_{10} not exist

eg:

RK ← FK

Emp (eid, ename, supID)

eid	ename	supID
e_1	A	NULL
e_2	B	e_1
e_3	C	e_1
e_4	C	e_2
e_5	D	NULL

eg:

RK	AK	
R(A	B	C)
a_1	2	
a_2	NULL	
a_3	4	
a_4	6	

FK	
S(D	E)
2	e_1
2	e_2
4	e_3
NULL	e_4

one to many relation

- Foreign Key allowed NULL values
- Record whose FK value is NULL is out of refrential integrity constraints.
- Each record of refrenced relation can relati by many (0 or more) refrencing relation records.
- Each record of refrencing relation can relati atmost 1 refrenced rel records.

Foreign Key Integrity constraints :-

op^n on Refrenced Relation (stud)

(i) **Insertion** : No FK voilation

(ii) **Deletion**: May cause FK violation

(DEFAULT) Sol 1 ⇒ On Delete no Action
(Refrenced Rel DELETE not Allowed)

Sol 2 ⇒ On Delete Cascade

(Allowed to DELETE refrenced relation + DBMS delete all refrencing records)

Sol 3 ⇒ On Delete set NULL

(if setting NULL in FK is allowed then set it NULL, and allowed to delete refrenced record)

(iii) **Updation** : May cause FK violation

(DEFAULT) SOL 1 ⇒ On update no Action

Sol 2 ⇒ On update cascade

Sol 3 ⇒ On update set NULL

op^n on Refrencing Relation (Enroll)

(i) **Insertion** : May cause FK voilation

if FK voilation then not allowed to insert

(ii) **Deletion** : No voilation

(iii) **Updation** : May cause FK voilation, if yes then not allowed eg:-

PK: Stud (sid, sname, DOB) | FK: Enroll(Sid, Cid, fee)

```
CREATE TABLE stud
(
  sid varchar(10) PRIMARY KEY,
  Sname varchar(50),
  DOB date
);
```

```
CREATE TABLE Enroll
(
  Sid varchar(10),
  cid varchar(10),
  fee integer(5)
  Primary key (sid, cid),
  FOREIGN KEY (Sid) REFERENCES
  Stud(Sid)
  ON DELETE CASCADE
  ON UPDATE CASCADE
);
```

Q. R(A B) , S(C D) where

- B FK refrences to S with ON DELETE CASCADE
- D FK refrences to R with ON DELETE SET NULL

R

A	B
2	3
3	3
4	4
5	4
6	7

S

C	D
3	2
4	2
5	4
6	4
7	5

(i) How many records forced to delete if (2,3) record in R deleted. (0)

R

A	B
2	3
3	3
4	4
5	4
6	7

ON DELETE CASCADE

S

C	D
3	~~2~~ NULL
4	~~2~~ NULL
5	4
6	4
7	5

ON DELETE SET NULL

So if we DELETE R(2,3) then FK pointing to it is set to NULL.

(ii) How many records forced to delete if (3,2) record in S is deleted. (2)

R

A	B
~~2~~	~~3~~
~~3~~	~~3~~
4	4
5	4
6	7

ON DELETE CASCADE

S

C	D
3	2
4	~~2~~ NULL
5	4
6	4
7	5

ON DELETE SET NULL

FK = 3

DELETE S(3,2) → DELETE R(2,3) → SET NULL in S at FK = 2

↓

FK = 3

DELETE R(3,3)

↓

SET NULL in S at FK = 3

NOTE:- while Performing DELETION opⁿ on a table, first check for any FK which is refrencing to this table.

If some FK is refrencing then perform DELETION at refrencing table also according to DELETE action mentioned in its defination (relational schema).

Normalization :-

- It is used to reduce or eliminate redundancy in DB relations or tables.

- Redundancy can occur in relation if two or more independent relations stored in one single relation.

eg:-

(i) Sid → Sname, DOB (Stud Info)

(ii) Cid → Cname, Instructor (Course Info)

(iii) sid cid → fee (Mapping)
PK

Now make a single relation for mapping:-

Sid	Sname	DOB	c id	c-name	inst	fee
S_1	A	2000	C_1	DB	karth	2000
S_2	A	2000	C_1	DB	karth	2000
S_3	B	1995	C_1	DB	karth	2000
S_3	B	1995	C_2	OS	Sagar	3000
S_3	B	1995	C_3	CN	Ram	2500

redundancy

Problem due to redundancy (DB Anomalies) :-

(i) Insert Anamolies:- Not possible to insert some data without other independent data.

eg:- Not possible to insert course without atleast one student

(ii) Deletion Anamolies :- Because of deletion of data also deletes other independent data.

eg: Because of deletion of student info also lost course info.

(iii) Updation Anamolies :- If some redundant copy updated, other redundant copy fails to update which causes inconsistency.

Normalization of DB :-

Decompose relation 'R' into two or more subrelation to reduce or eliminate redundancy.

R_1(Sid, Sname, DOB)
PK (Sid)

Sid	Sname	DOB
s_1	–	–
s_2	–	–
s_3	–	–

R_2(Sid, cid, fee)
FK (Sid), FK (cid)

Sid	cid	fee
s_1	c_1	–
s_2	c_1	–
s_3	c_1	–
s_3	c_2	–
s_3	c_3	–

R_3(cid, name, Inst)
PK (cid)

cid	name	Inst
c_1	–	–
c_2	–	–
c_3	–	–

Functional Dependency ($X \to Y$) -

X and Y some attributes set over R

$X \to Y$ exist in R

if $t_1.x = t_2.x$ then $t_1.y = t_2.y$

for any tuple t_1, t_2

R_1

	X	Y
t_1	x_1	y_5
t_2	x_1	y_5
	x_2	y_4
	x_3	y_4
t_3	x_1	y_5

must be same

may or may not same

$X \to Y$ FD exist

R_2

X	Y
x_1	y_2
x_1	y_1
x_2	y_3

$X \to Y$ FD not implied / exist

R

A	B	C
a_1		
a_2		
a_3		

A : key

$A \to B$

$A \to C$

Trivial FD (Reflexive FD) :-

X , Y some attributes over R

$X \rightarrow Y$ trivial FD only if $X \supseteq Y$ (superset)

eg:

R

Sid	Sname	Cid
S_1	A	C_1
S_1	A	C_2
S_2	B	C_2
S_3	B	C_3

$$\left.\begin{array}{l} Sid \rightarrow Sid \\ Sname \rightarrow Sname \\ Cid \rightarrow Cid \\ Sid\,Sname \rightarrow Sname \\ Sid\,Sname \rightarrow Sid\,Sname \end{array}\right\} \text{Trivial FD's}$$

Every possible trivial FD over attributes of Relational schema R is always members of relation R.

Non-trivial FDs :-

$X \rightarrow Y$ non trivial FD if no common Attr in X & Y attr sets.

eg:- Sid → Sname ✓ } Non-trivial FD's

Sid Cid → Sname ✓

Sid Sname → Sname ✗ (common attr)

Sname → Sid ✗ (X→Y not exist in relation)

Semi-Non Trivial FD:-

Combination of trivial and non-trivial FD

X → XY

Sid → Sid Sname — splitting:
- Sid → Sid Trivial
- Sid → Sname non trivial

X Y → YZ

Sid Cid → Cid Sname — splitting:
- Sid Cid → Cid Trivial
- Sid Cid → Sname Non-trivial

Armstrong Rules over FD's:-

X, Y, Z some attribute sets over R

(i) Reflexivity :- X→X Always true

(ii) Transitivity :- if X→Y, Y→Z then X→Z

(iii) Augmentation :- if X→Y then XZ→YZ

(iv) Split Rule :- if X→YZ then X→Y & X→Z

(v) Union/merge rule :- if $X \to Y$, $X \to Z$ then $X \to YZ$ always true

Attribute closure (X^+):-

$X^+ = \{$ set of all attributes which can be determined by attribute set X $\}$

eg:- FD set $\{A \to B, D \to E, BC \to D, AB \to C, EF \to G\}$

$(A^+) = \{A, B, C, D, E\}$

$(B)^+ = \{B\}$

$(AF)^+ = \{A, B, C, D, E, F\}$

Superkey :-

X is some attr set over R

- X is super key of relation R iff X^+ must determines all attributes of Relation R

eg:- $R(A, B, C, D, E)$

$\{AB \to C, B \to D, C \to E\}$

$(AB)^+ = \{A, B, C, D, E\}$ super key

$(ABC)^+ = \{A, B, C, D, E\}$ super key

$(BC)^+ = \{B, C, D, E\}$ Not super key

Candidate key :-

- It is minimal superkey.

X is candidate key of Rel R iff

(i) X must be super key of R

$X^+ = \{\text{all attributes of } R\}$

(ii) no proper subset of X is super key of R.

$\forall y \subset X$ such that $y^+ = \{$ should not contain all attr of R $\}$

Q. Find CK's of given Relational scheme :-

R(A, B, C, D, E, F) (CK = BDE)

(i) FD's $= \{C \to F, B \to A, D \to C\}$

as E is not mentioned in FD's

so $E^+ = \{E\}$ so E must be present in CK.

$X \longrightarrow Y$

$$\begin{bmatrix} C \\ B \\ D \end{bmatrix} \quad \begin{bmatrix} F \\ A \\ C \end{bmatrix}$$

(ii) $R(ABCDEF)$ $\{F \to E, B \to F, C \to A\}$

X	$\to Y$
F	E
B	F
C	A

$(CK = BCD)$

NOTE :-

[$X \to Y$ is non trivial FD with Y is prime attribute in R] $\Leftrightarrow$ [R has atleast two candidate keys]

Q: $R(ABCDEF) = \{F \to E, B \to F, C \to A, A \to C\}$

C is prime Attr

$(CK = BCD)$ already calculated

$(CBD)^+ = \{B D F E A C\}$

$(ABD)^+ = \{B D F E A C\}$

Steps:-

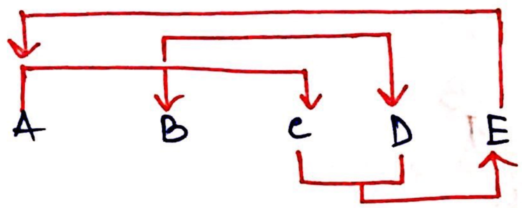
1. Find one candidate key [XY]

2. Now if $Z \to Y$ given or exist in R

   then $XY \Rightarrow \underbrace{XZ}_{\text{Test minimal set}}$

3. Now if $W \to Z$, repeat same

Q. $R(ABCDE) = \{A \to BC,\ CD \to E,\ B \to D,\ E \to A\}$



$(A)^+ = \{A, B, C, D, E\}$ CK

- since $E \to A$ prime

so, check $(E)^+$

$(E)^+ = \{E\ A\ B\ C\ D\}$ CK

- $CD \to E$ prime

Now checks for $(CD)^+$

so, $(CD)^+ = \{C, D, E, A, B\}$

proper subset

$(C)^+ = C$

$(D)^+ = D$

so, $(CD)^+$ is CK

$B \to D$ prime as CD is CK

so, CD , CB

replace

$(CB)^+ = \{C, B, D, E, A\}$

$C^+ = \{C\}$ $B^+ = \{B, D\}$

so, CB is CK

So there are total 4 CK's

$CK = \{A, E, CD, BC\}$

Q: $R(ABCDEF) = \{AB \to C, C \to D, CD \to BE, DE \to F, EF \to A\}$

$(AB)^+ = \{A, B, C, D, E, F\}$ — CK

$EF \to A$ (prime)

$(BEF)^+ = \{A, B, C, D, E, F\}$ — CK

proper subset

$B^+ = \{B\}$
$E^+ = \{E\}$
$F^+ = \{F\}$

$(BE)^+ = \{B, E\}$
$(EF)^+ = \{E, F, A\}$

$DF \rightarrow F$ (prime)

$BE\underline{F} \longrightarrow (BDE)^+ = \{A, B, C, D, E, F\}$

checks for its proper subset.

similarly, we get (C+)

so, $\left[CK = \{ AB,\ C,\ BEF,\ BDE,\ .. \} \right]$

NOTE:- To finding out candidate key is an example of NP- complete problem. $\underline{T(n) = O(a^n)}$