Database Assignment 3

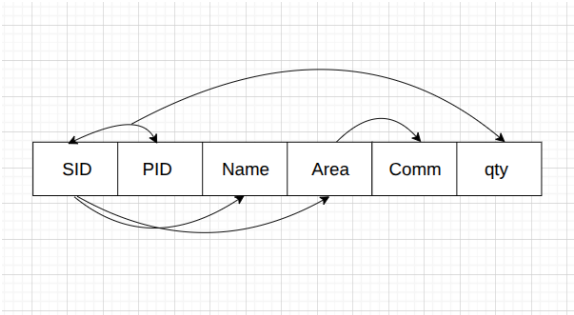
Ziad Mohamed Eliwa - 900246124

November 9, 2025

Problem 1

a. i. Primary key (SID,PID)

ii. Functional Dependencies.



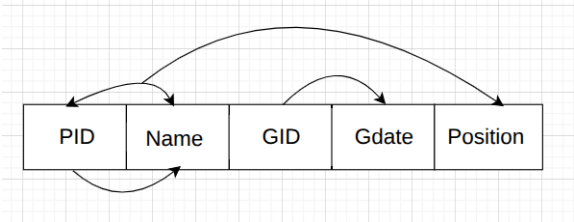
iii. This relation stands in the first normal form. Because it fails in the second normal form because we have a dependency between a non-prime attribute (name,comm,area) and a subset of the primary key.

iv. First, we split the dependencies SID → name, SID → area to reach the second normal form, then we split the transitive dependency Area → comm and SID → comm to reach the third normal form We get the following:
SAP (SID,PID,qty)
Salesman (SID,Name,Area)
Area(Area,Comm)
This is also in Boyce-Codd normal form as there is no dependency between two non-candidate keys.
It is also in fourth normal form as there is no more than one multivalued dependency.

b. i. Primary Key (PID,GID)

ii. Functional dependencies

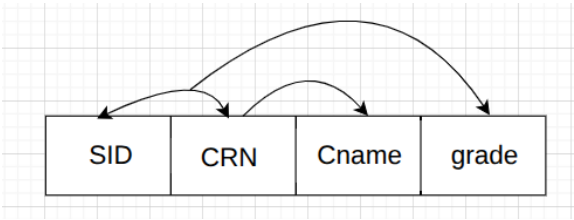
iii. This relation stands in the first normal form. Because it fails in the second normal form because we have a dependency between a non-prime attribute (name,Gdate,position) and a subset of the primary key.



iv. First, we get rid of the partial dependencies between PID → name and GID → GDate to reach second normal form. Then, we can deduce that it reached the forth normal form. We get the relation as follows:
ATLANTA(PID,GID,position)
Game(GID,Gdate)
Player(PID,Name)

c. i. Primary Key(SID,CRN)

ii. Functional dependencies

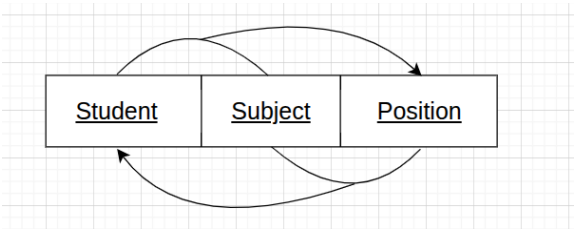


iii. This relation stands in the first normal form. Because it fails in the second normal form because we have a dependency between a non-prime attribute (Cname) and a subset of the primary key.

iv. We split the dependency CRN → Cname to get the following relation:
GRADE(SID,CRN,grade)
Course(CRN,Cname)

d. i. Primary Key (s,j,p)

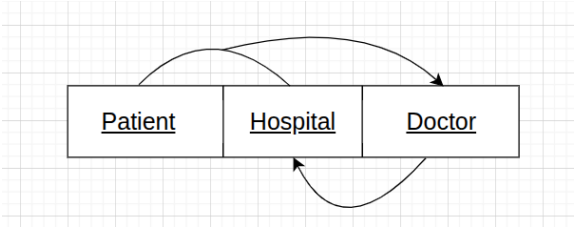
ii. Functional Dependencies.



iii. The table is in the forth normal form because there is no multivalued dependencies and there are no dependencies exist on non-candidate keys as dependencies are Subject,Position → Student and Student,Subject → Position.

e. i. Candidate Keys (Patient,Doctor,Hospital)

ii. Functional Dependencies.



iii. The table is in Boyce-Codd normal form as there are no non-prime attributes. But it is not in fourth normal form as there are two multivalued dependencies Patient,Hospital → Doctor and Doctor → Hospital.

iv. We normalize this to fourth normal form by splitting the multivalued dependencies to the following:
PD (Patient, Doctor)
DH (Doctor, Hospital)

Problem 2

a.
```
CREATE ROLE roleA;
GRANT SELECT, UPDATE, DELETE, INSERT ,REFERENCES ON EMPLOYEE TO roleA WITH GRANT OPTION;
GRANT SELECT, UPDATE, DELETE, INSERT ,REFERENCES ON DEPARTMENT TO roleA WITH GRANT OPTION;
GRANT SELECT, UPDATE, DELETE, INSERT ,REFERENCES ON DEPT_LOCATIONS TO roleA WITH GRANT OPTION;
GRANT SELECT, UPDATE, DELETE, INSERT ,REFERENCES ON PROJECT TO roleA WITH GRANT OPTION;
GRANT SELECT, UPDATE, DELETE, INSERT ,REFERENCES ON WORKS_ON TO roleA WITH GRANT OPTION;
CREATE USER 'A'@'localhost' IDENTIFIED BY RANDOM PASSWORD;
GRANT roleA TO 'A'@'localhost';
SET DEFAULT ROLE roleA TO 'A'@'localhost';
```

b.
```
CREATE ROLE roleB;
CREATE VIEW EmployeeBView AS SELECT FNAME, MINIT, LNAME, SSN, BDATE, ADDRESS, SEX, SUPERSSN, DN
CREATE VIEW DepartmentBView AS SELECT DNAME, DNUMBER FROM DEPARTMENT;
GRANT SELECT ON EmployeeBView TO roleB;
GRANT SELECT, REFERENCES ON DepartmentBView TO roleB;
CREATE USER 'B'@'localhost' IDENTIFIED BY RANDOM PASSWORD;
GRANT roleB TO 'B'@'localhost';
SET DEFAULT ROLE roleB TO 'B'@'localhost';
```

c.
```
CREATE ROLE roleC;
CREATE VIEW EmployeeCView AS SELECT FNAME, MINIT, LNAME, SSN FROM EMPLOYEE;
CREATE VIEW ProjectCView AS SELECT PNAME, PNUMBER FROM PROJECT;
GRANT SELECT, REFERENCES ON EmployeeCView TO roleC;
GRANT SELECT, REFERENCES ON ProjectCView TO roleC;
GRANT SELECT, UPDATE, DELETE, INSERT ON WORKS_ON TO roleC;
CREATE USER 'C'@'localhost' IDENTIFIED BY RANDOM PASSWORD;
GRANT roleC TO 'C'@'localhost';
SET DEFAULT ROLE roleC TO 'C'@'localhost';
```

d.
```
CREATE ROLE roleD;
GRANT SELECT, REFERENCES ON EMPLOYEE TO roleD;
GRANT SELECT, UPDATE, DELETE,INSERT ON DEPENDENT TO roleD;
CREATE USER 'D'@'localhost' IDENTIFIED BY RANDOM PASSWORD;
GRANT roleD TO 'D'@'localhost';
SET DEFAULT ROLE roleD TO 'D'@'localhost';
```

e.
```
CREATE ROLE roleE;
CREATE VIEW EmployeeEView AS SELECT * FROM EMPLOYEE WHERE DNO = 3;
GRANT SELECT ON EmployeeEView TO roleE;
CREATE USER 'E'@'localhost' IDENTIFIED BY RANDOM PASSWORD;
GRANT roleE TO 'E'@'localhost';
SET DEFAULT ROLE roleE TO 'E'@'localhost';
```