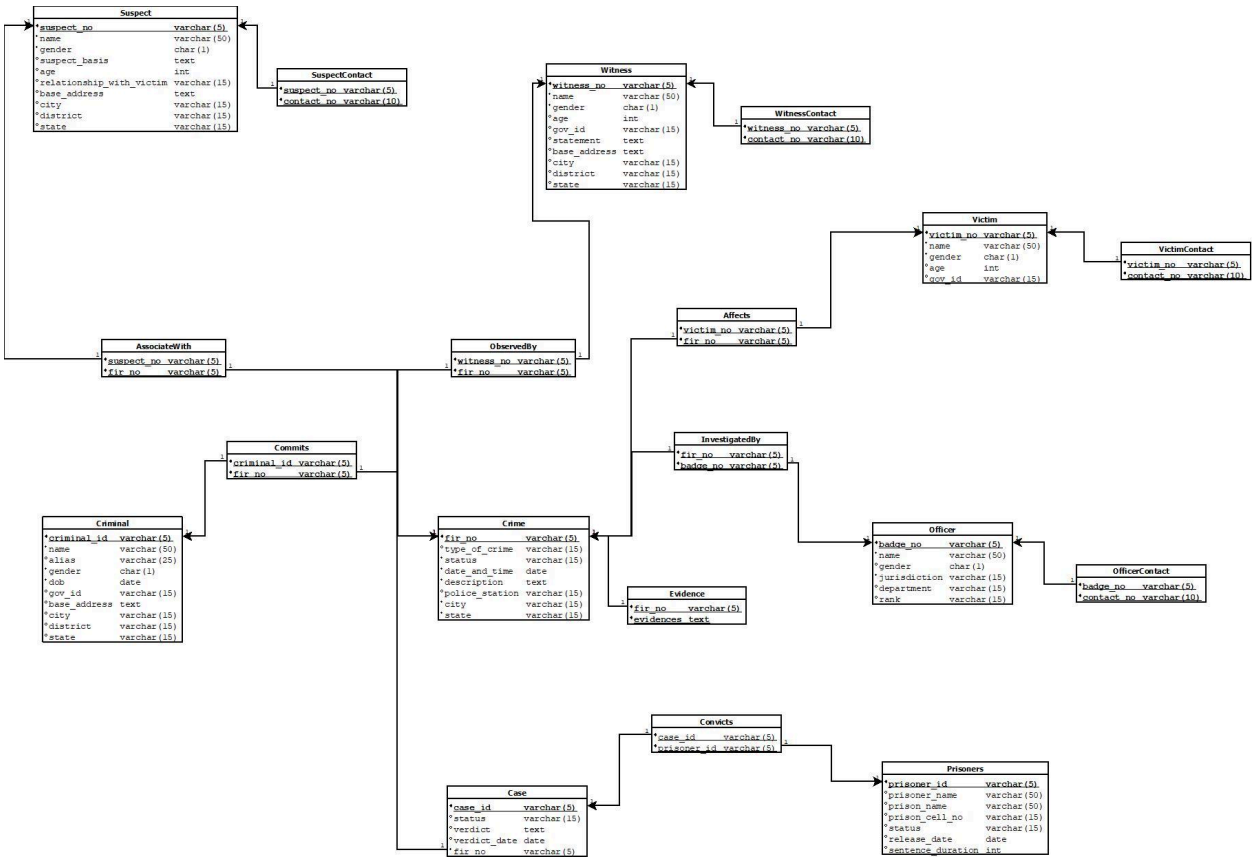
Criminal Record Management System

Course: IT214 - Database Management System
Instructor: Prof. P M Jat

Table of Contents

(1) Relational Schema



(2) Minimal FD Set

- Criminal
  - criminal_id → name
  - criminal_id → alias
  - criminal_id → gender
  - criminal_id → dob

    - criminal_id → gov_id
    - criminal_id → base_address
    - criminal_id → city
    - criminal_id → district
    - criminal_id → state
        - Key : criminal_id

- Crime
    - fir_no → type_of_crime
    - fir_no → status
    - fir_no → date_and_time
    - fir_no → description
    - fir_no → police_station
    - fir_no → city
    - fir_no → state
        - Key : fir_no

- Officer
    - badge_no → name
    - badge_no → gender
    - badge_no → jurisdiction
    - badge_no → department
    - badge_no → rank
        - Key : adge_no

- Case
    - case_id → status
    - case_id → verdict
    - case_id → verdict_date
    - case_id → fir_no
        - Key : case_id

- Prisoners
    - prisoner_id → prisoner_name
    - prisoner_id → prison_name
    - prisoner_id → prison_cell_no

    - prisoner_id → status
    - prisoner_id → release_date
    - prisoner_id → sentence_duration
        - Key : prisoner_id

- Victim
    - victim_no → name
    - victim_no → gender
    - victim_no → age
    - victim_no → gov_id
        - Key : victim_no

- Witness
    - witness_no → name
    - witness_no → gender
    - witness_no → age
    - witness_no → gov_id
    - witness_no → statement
    - witness_no → base_address
    - witness_no → city
    - witness_no → district
    - witness_no → state
        - Key : witness_no

- Suspect
    - suspect_no → name
    - suspect_no → gender
    - suspect_no → suspect_basis
    - suspect_no → age
    - suspect_no → relationship_with_victim
    - suspect_no → base_address
    - suspect_no → city
    - suspect_no → district
    - suspect_no → state
        - Key : suspect_no

- There is no functional dependency present in the remaining relationships.

(3) Proof that relations are in BCNF.

Definition of Boyce-Codd Normal Form (BCNF)

A relation R is in Boyce-Codd Normal Form, when determinant of every FD that holds on R, is the super-key of R. In other words, For every FD A → B that holds in relation R, A is its super-key.

We can observe this : all functional dependencies (FDs) present in the Fmin set have their left side as a key. There is not a functional dependency X → Y in the Fmin set, where X is not a key of the relation.

Therefore, all relations are in the Boyce-Codd Normal Form (BCNF).

(4) DDL Scripts

```
CREATE SCHEMA CRMS;

SET search_path TO CRMS;

--1. Criminal Table
CREATE TABLE criminal (
    criminal_id VARCHAR(5) PRIMARY KEY,
    name VARCHAR(50),
    alias VARCHAR(25),
    gender CHAR(1) NOT NULL CHECK (gender IN ('M', 'F', 'O')),
    dob DATE NOT NULL,
    gov_id VARCHAR(15) UNIQUE,
    base_address TEXT,
    city VARCHAR(15) NOT NULL,
    district VARCHAR(15) NOT NULL,
```

```
    state VARCHAR(15) NOT NULL
);


--2. Crime Table
CREATE TABLE crime (
    fir_no VARCHAR(5) PRIMARY KEY,
    type_of_crime VARCHAR(15) NOT NULL,
    status VARCHAR(15) DEFAULT 'Open',
    date_and_time TIMESTAMP NOT NULL,
    description TEXT NOT NULL,
    police_station VARCHAR(15) NOT NULL,
    city VARCHAR(15) NOT NULL,
    state VARCHAR(15) NOT NULL
);


--3. Evidence Table(Multi Value attribute of crime table)
CREATE TABLE evidence (
    fir_no VARCHAR(5),
    evidences TEXT NOT NULL,
    PRIMARY KEY (fir_no, evidences),
    FOREIGN KEY (fir_no) REFERENCES crime(fir_no)
    ON DELETE CASCADE ON UPDATE CASCADE
);


--4. Commits Table (Many-to-Many : criminal - crime)
CREATE TABLE commits (
    criminal_id VARCHAR(5),
    fir_no VARCHAR(5),
    PRIMARY KEY (criminal_id, fir_no),
    FOREIGN KEY (criminal_id) REFERENCES criminal(criminal_id)
    ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (fir_no) REFERENCES crime(fir_no)
    ON DELETE CASCADE ON UPDATE CASCADE
);


--5. Officer Table
```

```sql
CREATE TABLE officer (
    badge_no VARCHAR(5) PRIMARY KEY,
    name VARCHAR(25) NOT NULL,
    gender CHAR(1) NOT NULL CHECK (gender IN ('M', 'F', 'O')),
    jurisdiction VARCHAR(15) NOT NULL,
    department VARCHAR(15) NOT NULL,
    rank VARCHAR(15)
);


--6. Officer-Contact Table (Multi Value attribute of officer table)
CREATE TABLE officer_contact (
    badge_no VARCHAR(5),
    contact_no VARCHAR(10),
    PRIMARY KEY (badge_no, contact_no),
    FOREIGN KEY (badge_no) REFERENCES officer(badge_no)
      ON DELETE CASCADE ON UPDATE CASCADE
);


--7. InvestigatedBy table (Many-to-Many : officer - crime)
CREATE TABLE investigated_by (
    fir_no VARCHAR(5),
    badge_no VARCHAR(5),
    PRIMARY KEY (fir_no, badge_no),
    FOREIGN KEY (fir_no) REFERENCES crime(fir_no)
      ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (badge_no) REFERENCES officer(badge_no)
      ON DELETE CASCADE ON UPDATE CASCADE
);


--8. Case Table
CREATE TABLE "case" ( -- "case" because simple case give error
    case_id VARCHAR(5) PRIMARY KEY,
    status VARCHAR(15) DEFAULT 'Pending',
    verdict TEXT,
    verdict_date DATE,
    fir_no VARCHAR(5) NOT NULL,
    FOREIGN KEY (fir_no) REFERENCES crime(fir_no)
```

```
    ON DELETE CASCADE ON UPDATE CASCADE
);


--9. Prisoner Table
CREATE TABLE prisoners (
    prisoner_id VARCHAR(5) PRIMARY KEY,
    prisoner_name VARCHAR(50) NOT NULL,
    prison_name VARCHAR(50) NOT NULL,
    prison_cell_no VARCHAR(15) NOT NULL,
    status VARCHAR(15) NOT NULL,
    release_date DATE,
    sentence_duration INT CHECK (sentence_duration > 0) -- In days
);


--10. Convicts Table (Many-to-Many : case - prisoners)
CREATE TABLE convicts (
    case_id VARCHAR(5),
    prisoner_id VARCHAR(5),
    PRIMARY KEY (case_id, prisoner_id),
    FOREIGN KEY (case_id) REFERENCES "case"(case_id)
      ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (prisoner_id) REFERENCES prisoners(prisoner_id)
      ON DELETE CASCADE ON UPDATE CASCADE
);


--11. Victim Table
CREATE TABLE victim (
    victim_no VARCHAR(5) PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    gender CHAR(1) NOT NULL CHECK (gender IN ('M', 'F', 'O')),
    age INT CHECK (age > 0),
    gov_id VARCHAR(15) UNIQUE
);


--12. Victim-Contact Table (Multi Value attribute of victim table)
CREATE TABLE victim_contact (
```

```
    victim_no VARCHAR(5),
    contact_no VARCHAR(10),
    PRIMARY KEY (victim_no, contact_no),
    FOREIGN KEY (victim_no) REFERENCES victim(victim_no)
      ON DELETE CASCADE ON UPDATE CASCADE
);


--13. Affects Table (Many-to-Many : crime - victim)
CREATE TABLE affects (
    victim_no VARCHAR(5),
    fir_no VARCHAR(5),
    PRIMARY KEY (victim_no, fir_no),
    FOREIGN KEY (victim_no) REFERENCES victim(victim_no)
      ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (fir_no) REFERENCES crime(fir_no)
      ON DELETE CASCADE ON UPDATE CASCADE
);


--14. Witness Table
CREATE TABLE witness (
    witness_no VARCHAR(5) PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    gender CHAR(1) NOT NULL CHECK (gender IN ('M', 'F', 'O')),
    age INT CHECK (age > 0),
    gov_id VARCHAR(15) UNIQUE,
    statement TEXT NOT NULL,
    base_address TEXT,
    city VARCHAR(15) NOT NULL,
    district VARCHAR(15) NOT NULL,
    state VARCHAR(15) NOT NULL
);


--15. Witness-Contact Table (Multi Value attribute of witness table)
CREATE TABLE witness_contact (
    witness_no VARCHAR(5),
    contact_no VARCHAR(10),
    PRIMARY KEY (witness_no, contact_no),
```

```
    FOREIGN KEY (witness_no) REFERENCES witness(witness_no)
      ON DELETE CASCADE ON UPDATE CASCADE
);


--16. ObservedBy Table (Many-to-Many : crime - witness)
CREATE TABLE observed_by (
    witness_no VARCHAR(5),
    fir_no VARCHAR(5),
    PRIMARY KEY (witness_no, fir_no),
    FOREIGN KEY (witness_no) REFERENCES witness(witness_no)
    ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (fir_no) REFERENCES crime(fir_no)
    ON DELETE CASCADE ON UPDATE CASCADE
);


--17. Suspect Table
CREATE TABLE suspect (
    suspect_no VARCHAR(5) PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    gender CHAR(1) NOT NULL CHECK (gender IN ('M', 'F', 'O')),
    suspect_basis TEXT NOT NULL,
    age INT CHECK (age > 0),
    relationship_with_victim VARCHAR(15),
    base_address TEXT,
    city VARCHAR(15) NOT NULL,
    district VARCHAR(15) NOT NULL,
    state VARCHAR(15) NOT NULL
);


--18. Suspect-Contact Table (Multi Value attribute of suspect table)
CREATE TABLE suspect_contact (
    suspect_no VARCHAR(5),
    contact_no VARCHAR(10),
    PRIMARY KEY (suspect_no, contact_no),
    FOREIGN KEY (suspect_no) REFERENCES suspect(suspect_no)
    ON DELETE CASCADE ON UPDATE CASCADE
);
```

```sql
--19. AssociateWith Table (Many-to-Many : crime - suspect)
CREATE TABLE associate_with (
    suspect_no VARCHAR(5),
    fir_no VARCHAR(5),
    PRIMARY KEY (suspect_no, fir_no),
    FOREIGN KEY (suspect_no) REFERENCES suspect(suspect_no)
    ON DELETE CASCADE ON UPDATE CASCADE,
    FOREIGN KEY (fir_no) REFERENCES crime(fir_no)
    ON DELETE CASCADE ON UPDATE CASCADE
);
```