

대구 경찰청 사건 현황 관리 시스템

21812221 구본식(팀장)
21812248 이창협
21812206 김민재
22012198 남경민
22012212 양소은

목 차

01.

서비스 소개

02.

ERD

03.

경찰관

04.

관리자

05.

시연

06.

Q & A

대구에서 발생하는 사건들은 대구 경찰청 아래 경찰서에서 팀별로 사건들을 처리한다.

서비스 관리자는 발생한 사건들을 관리 및 등록, 키워드 별 통계를 내고 통계 결과를 볼 수 있다.

대구 경찰청 소속 경찰관들은 키워드(ex. 사건 종류, 가해자, 팀별 처리사건 등) 을 이용해 모든 사건들을 조회 할 수 있다.

사건의 진행을 맡은 경찰관들은 사건 진행중, 종료 시점에 보고서를 작성할 수 있다.

이때 다음 요구조건을 만족하는 대구경찰청 사건 현황 관리 시스템을 위한 시스템을 설계한다.

- 서비스 사용자

: 대구경찰청 소속 경찰관

1. 완료된 사건 또는 진행되는 사건에 대해 정보를 기입하고 조회할 수 있다.

- 서비스 관리자: 대구경찰청

: 대구경찰청 경찰 행정 관리 담당관

1. 경찰관 관리 (등록, 수정, 삭제)를 한다.

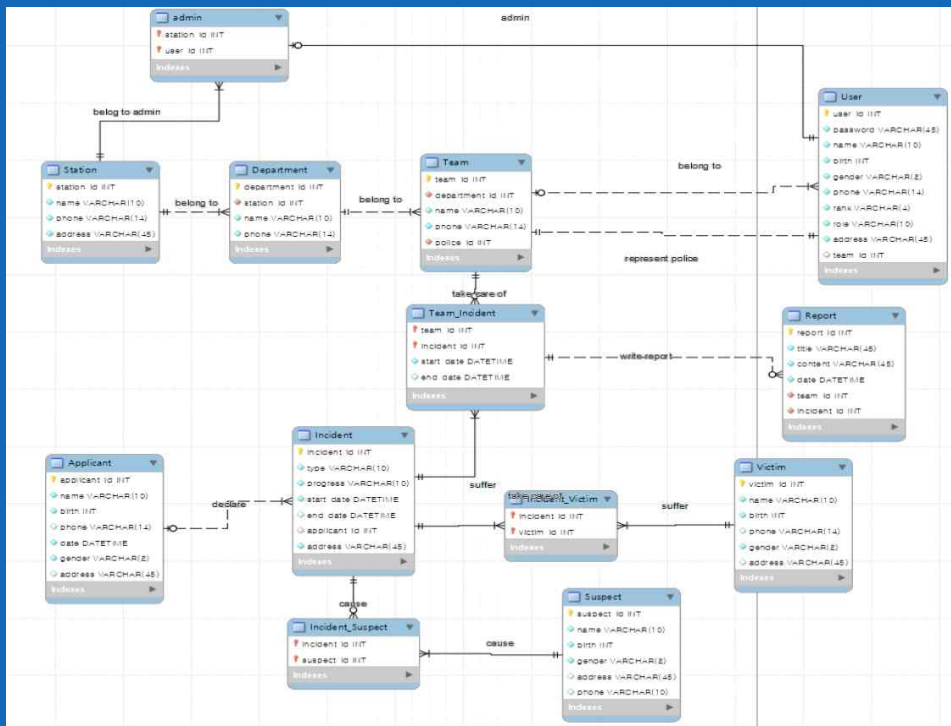
2. 접수된 사건을 등록하고 신고자 등록 및 팀을 배정한다.

3. 사건에 대한 통계를 낼수 있다.

ex) 서, 부서, 팀 별 통계

EERD 전체 구조

- Station : 서
- Department : 부서
- Team : 팀
- admin : 관리자
- User : 경찰관
- Team_Incident : 팀별 사건
- Report : 보고서
- Incident : 사건
- Victim : 피해자
- incident_Victim : 사건별 피해자
- Applicant : 신고자
- Incident_Suspect : 사건별 가해자
- Suspect : 가해자



<03.경찰관>

데이터베이스

1. 사건 조회 및 결과 - 사건ID를 통한 조회.php

사건 조회

서: 부서: 팀: 진행도: 사건종류: 주소: 검색

사건 ID: 검색

<사건 정보를 받아오는 프로시저>

```
USE PoliceDB;
DROP PROCEDURE IF EXISTS search_Incident_Other_info_Proc;
DELIMITER $$
CREATE PROCEDURE
search_Incident_Other_info_Proc(IN IncidentId INT)
BEGIN
    select i.incident_id , i.type ,
           i.start_date, i.end_date, i.progress, i.address ,i.applicant_id
    from Incident i
    where i.incident_id= IncidentId;
END $$
DELIMITER
```

- 사건의 ID를 통해 사건의 정보를 검색

<사건을 맡은 팀 리스트를 받아오는 프로시저>

```
USE PoliceDB;
DROP PROCEDURE IF EXISTS search_Incident_Team_info_Proc;
DELIMITER $$
CREATE PROCEDURE
search_Incident_Team_info_Proc(IN IncidentId INT)
BEGIN
    select
        ti.team_id,
        CONCAT(s.name," ", d.name," ",t.name) as Unit ,
        CONCAT(DATE_FORMAT(ti.start_date, '%Y-%m-%d %H:%i:%s')," ~ ",
        IFNULL(DATE_FORMAT(ti.end_date,'%Y-%m-%d %H:%i:%s')," 진행중")) as period
    from Team_Incident ti join Team t on ti.team_id = t.team_id
    join Department d on d.department_id= t.department_id
    join Station s on s.station_id=d.station_id
    where ti.incident_id=IncidentId;
END $$
DELIMITER
```

- "사건테이블"과 "사건_팀 테이블"은 "1:N"이기 때문에 두 테이블간에 조인을 통해 한번에 가져오게 되면 사건의 정보가 뺏기기 되는 현상을 막기 위해 별개의 쿼리문을 이용하여 받아옴.
(하나의 쿼리로 가져온 후 애플리케이션 단(PHP) 에서 처리도 가능)

- PHP에서 select 한 데이터의 후 처리 로직을 줄이기 위해 필요한 데이터 형태로 response 함

- 두 프로시저 호출을 하나의 PHP에서 구현

<03.경찰관>

데이터베이스

1. 사건 조회 및 결과 - 다양한 필터링조건 통한 조회.php

<사건 정보를 받아오는 프로시저>

```
USE PoliceDB;
DROP PROCEDURE IF EXISTS search_IncidentByFilter_Other_info_Proc;
DELIMITER $$
CREATE PROCEDURE
search_IncidentByFilter_Other_info_Proc(
    IN s_n varchar(10),IN d_n varchar(10),
    IN t_n varchar(10),IN ip_n varchar(10),
    IN it_n varchar(10),IN id_n varchar(45)
)
BEGIN
    select i.incident_id , i.type, #사건ID는 "사건상세 페이지로 이동할 때 사용"
    i.start_date,i.end_date, i.progress, i.address ,i.applicant_id
    from Incident i
    join Team_Incident ti on i.incident_id= ti.incident_id
    join Team t on ti.team_id = t.team_id
    join Department d on d.department_id = t.department_id
    join Station s on s.station_id = d.station_id
    where
    s.name = ifnull(s_n, s.name)
    and d.name = ifnull(d_n, d.name)
    and t.name = ifnull(t_n, t.name)
    and i.progress = ifnull(ip_n, i.progress)
    and i.type = ifnull(it_n, i.type)
    and (INSTR(i.address, ifnull(id_n,"")) > 0)
    group by i.incident_id;
END $$
DELIMITER
```

사건 조회

서: 부서: 팀: 진행도: 사건종류: 주소:

사건 ID :

- "서, 부서, 팀, 진행도, 사건종류, 주소" 카테고리에서 원하는 필터링 조건 입력을 통해 사건 검색 가능.
- 선택하지 않은 카테고리는 "null"로 판별.
- 해당 프로시저를 통해 사건의 정보를 받아오고 위에서 설명한 "사건을 맡은 팀 리스트" 조회 프로시저를 통해 팀 리스트를 받아옴.

1. 사건 조회 및 결과 - (사건ID, 다양한 필터링 조건).php

<상태별(권한 별, 신고자 유무 별) 동적 html>

#경찰관 일경우 -> 사건을 맡고 있는 팀인지 확인

```
$check = false;
if($role=="Police") {

    if($progress_s[$i] != "수사완료") {

        for($k=0; $k<count($team_list[$i]); $k++) {
            if($teamId == $team_list[$i][$k]['team_id']){
                $check = true;
                break;
            }
        }
        if($check == true) { #사건을 맡은 팀인경우
            if(is_null($applicant_ids[$i])) {
                echo "<td>";
                echo "<form method = 'get' , action = 'Incident_AddApplicant
                echo "<input type = 'text' name = 'incident_id' value = '". $i
                echo "<input type = 'text' name = 'role' value = '". $role ." "
                echo "<input type = 'text' name = 'teamId' value = '". $teamId
                echo "<button type = 'submit'> 신고자 등록 </button>";
                echo "</form>";
                echo "</td>";
            }
            echo "<td>";
            echo "<form method = 'get' , action = 'Incident_AddSuspect.php'>
            echo "<input type = 'text' name = 'incident_id' value = '". $incid
            echo "<input type = 'text' name = 'role' value = '". $role ." " hidd
            echo "<input type = 'text' name = 'teamId' value = '". $teamId ." "
            echo "<button type = 'submit'> 가해자 추가 </button>";
```

- 로그인 후 부터 매 페이지(PHP) 이동시에 사용자의 정보를 "hidden"으로 전달 (Stateless하기 때문)

- 수사가 완료된 사건에 대해서는 모든 버튼 비활성화

- 로그인한 사용자가 사건을 맡고 있는 팀에 소속될 경우에는 "신고자추가, 보고서 쓰기, 가해자 추가, 피해자 추가" 버튼이 활성화

- 사건을 맡고 있지 않은 팀의 사용자 일 경우에는 위의 버튼을 비활성화

- 사건에 신고자가 한명만 존재하기 때문에 신고자가 이미 존재하는 사건에는 "신고자 추가 버튼" 비활성화

- 관리자일 경우 위의 버튼은 사용할 수 없고 사건에 대해서 "팀추가 버튼"만 활성화(완료된 사건 제외)

<03.경찰관>

데이터베이스

1. 사건 조회 및 결과 - 결과

(1) 사건 ID를 통한 검색 - 경찰관

사건 조회

서: 부서: 팀:

사건 ID :

<-->

| 사건 ID | 종류 | 시작일시 | 종료일시 | 진행도 | 주소 | 팀 | |
|--|----|---------------------|------|------|-------------|-------------|-------------------------------------|
| 59 | 절도 | 2022-11-16 07:14:00 | | 배정완료 | 대구광역시 동구... | 서구경찰서 형사... | <input type="button" value="팀 추가"/> |
| 서구경찰서 형사과 1팀 2022-11-17 07:14:00 ~ 2022-12-17 07:14:00 서구경찰서 형사과 3팀 2022-11-18 10:13:00 ~ 2022-11-27 08:14:00 | | | | | | | |

(2) 다양한 필터링 조건을 통한 검색 - 경찰관

서: 부서: 팀: 진행도: 사건종류: 주소:

서: 부서: 팀: 진행도: 사건종류: 주소:

서: 부서: 팀: 진행도: 사건종류: 주소:

| | | | | | | | | | | |
|----|----|---------------------|--|------|----------|----------|--|---------------------------------------|---------------------------------------|---------------------------------------|
| 59 | 절도 | 2022-11-16 07:14:00 | | 배정완료 | 대구광역시... | 서구경찰서... | <input type="button" value="신고자 등록"/> | <input type="button" value="가해자 추가"/> | <input type="button" value="피해자 추가"/> | <input type="button" value="보고서 쓰기"/> |
| 41 | 살인 | 2022-11-03 00:32:00 | | 수사중 | 서구 서구... | 서구경찰서... | 서구경찰서 형사과 1팀 2022-11-17 07:14:00 ~ 2022-12-17 07:14:00 서구경찰서 형사과 3팀 2022-11-18 10:13:00 ~ 2022-11-27 08:14:00 | | | |
| 48 | 살인 | 2022-09-04 19:20:16 | | 수사중 | 서구 달서... | 서구경찰서... | | | | |
| 49 | 살인 | 2022-06-04 20:20:16 | | 수사중 | 서구 달서... | 서구경찰서... | <input type="button" value="가해자 추가"/> | <input type="button" value="피해자 추가"/> | <input type="button" value="보고서 쓰기"/> | |

(3) 관리자

| 사건 ID | 종류 | 시작일시 | 종료일시 | 진행도 | 주소 | 팀 | |
|-------|----|---------------------|------|-----|-------------|-------------|-------------------------------------|
| 46 | 사기 | 2021-08-04 17:20:16 | | 수사중 | 남구 봉곡로 한... | 남구경찰서 수사... | <input type="button" value="팀 추가"/> |
| 47 | 사기 | 2021-09-04 18:20:16 | | 수사중 | 남구 봉곡로 한... | 남구경찰서 수사... | <input type="button" value="팀 추가"/> |

2. 사건 상세 - 사건상세.php

1. 사건 정보

```
CREATE PROCEDURE
search_Incident_Other_plus_Applicant_info_Proc(IN IncidentId INT)

BEGIN
    select i.incident_id , i.type , i.start_date, i.end_date,
    i.progress, i.address ,
    a.applicant_id, concat(a.name,'(',a.birth,')') as a_info
    from Incident i
    left join Applicant a on i.applicant_id = a.applicant_id
    where i.incident_id= IncidentId;
END $$
DELIMITER
```

2. 팀 리스트 정보

```
CREATE PROCEDURE
search_Incident_Team_info_Proc(IN IncidentId INT)
BEGIN
    select
    ti.team_id,
    CONCAT(s.name," ", d.name," ",t.name) as Unit ,
    CONCAT( DATE_FORMAT(ti.start_date, '%Y-%m-%d %H:%i:%s'), " ~ ",
    IFNULL( DATE_FORMAT(ti.end_date, '%Y-%m-%d %H:%i:%s'), "전행중")) as period
    from Team_Incident ti join Team t on ti.team_id = t.team_id
    join Department d on d.department_id= t.department_id
    join Station s on s.station_id=d.station_id
    where ti.incident_id=IncidentId;
END $$
DELIMITER
```

3. 가해자 리스트 정보

```
BEGIN
    select s.suspect_id, concat(s.name,'(',s.birth,')') as a_info
    from Incident_Suspect ins join Suspect s on ins.suspect_id = s.suspect_id
    where ins.incident_id=IncidentId;
END $$
DELIMITER
```

4. 피해자 리스트 정보

```
BEGIN
    select v.victim_id, concat(v.name,'(',v.birth,')') as a_info
    from Incident_Victim iv join Victim v on iv.victim_id = v.victim_id
    where iv.incident_id=IncidentId;
END $$
DELIMITER
```

2. 사건상세 - 사건 상세 php

5. 보고서 리스트 정보

```
BEGIN
select r.report_id, concat(s.name," ", d.name," ",t.name,
"(작성일:", DATE_FORMAT(r.re_date, '%Y-%m-%d %H:%i:%s'), ")") as r_info
from Team_Incident ti join Report r
on ti.incident_id = r.incident_id and ti.team_id = r.team_id
join Team t on t.team_id = ti.team_id
join Department d on d.department_id = t.department_id
join Station s on d.station_id = s.station_id
where ti.incident_id = IncidentId;
END $$
DELIMITER
```

- 팀, 가해자, 피해자, 보고서는 사건 정보와 "1:N" 관계이므로
사건 정보 뺀뒤기를 막기위해 각 쿼리로 구성
- php에서 최소한의 로직을 위해 쿼리 응답으로 필요한 형식으로 response 하게 구성
- 왼쪽과 같이 하나의 php안에 구성

<?php

```
$con = mysqli_connect("policedb.cg9boi0b9vlu.ap-northeast-2.rds.amazonaws.com",
"admin", "rnqgstlr123!", "PoliceDB") or die("MySQL 접속 실패!");

$role = $_GET["role"];
$teamId = $_GET["teamId"];
$incident_id = $_GET["incidentId"];

$sql = "CALL search_Incident_Other_plus_Applicant_info_Proc('".$incident_id."')";
$sql .= "CALL search_Incident_Team_info_Proc('".$incident_id."')";
$sql .= "CALL search_Incident_Suspect_info_Proc('".$incident_id."')";
$sql .= "CALL search_Incident_Victim_info_Proc('".$incident_id."')";
$sql .= "CALL search_Incident_Report_info_Proc('".$incident_id."')";
```

```
$level = 0;
if(mysqli_multi_query($con, $sql)){
    do{
        if($result = mysqli_store_result($con)){
            #사건 정보 쿼리
            if($level == 0) {
                while($row = mysqli_fetch_array($result)){
                    $Incident_info[0] = $row['incident_id'];
                    $Incident_info[1] = $row['type'];
                    $Incident_info[2] = $row['start_date'];
                    $Incident_info[3] = $row['end_date'];
                    $Incident_info[4] = $row['progress'];
                    $Incident_info[5] = $row['address'];

                    #신고자 정보 - 단일!
                    $Incident_info[6] = $row['applicant_id'];
                    $Incident_info[7] = $row['a_info'];
                }
            }
            #팀 정보 쿼리
        }
        else if($level == 1){
            while($row = mysqli_fetch_array($result)){
                $team_ids[$t count] = $row['team_id'];
            }
        }
    } while($result = mysqli_store_result($con));
}
```

2. 사건 상세 - (사건 상세.php 및 결과)

<권한 별 동적 html>

```
<?php
#경찰관 일 경우 -> 사건을 맡고 있는 팀인지 확인
#관리자도 수정 못함.
$check = false;
if($role=="Police") {
    for($k=0; $k<$t_count; $k++) {
        if($teamId == $team_ids[$k]){
            $check = true;
            break;
        }
    }

    if($check == true) {
        echo "<a href=/PolicePHP/Incident_Update1.php?
            incidentId=".$Incident_info[0]."&role=".$role."&teamId=".$teamId.
            "> 수정하로 가기 </a>";
    }
}

#뒤에서 사용하기 위한 변수
if($check == false)
    $check = 0;
else
    $check = 1;
?>
```

- 로그인한 사용자(경찰관)가 사건 담당팀 소속일 경우 "사건 수정" 버튼 활성화
- 관리자는 비활성화

<결과>

사건 상세 페이지

[<--홈으로 수정하로 가기](#)

| | |
|----------|---|
| 사건 ID | 4 |
| 사건 종류 | 살인 |
| 팀 | 북구경찰서 형사과 1팀 2022-11-01 02:01:00 ~ 2024-11-01 02:01:00 |
| 시작일시 | 2022-11-01 02:01:00 |
| 종료일시 | |
| 사건 발생 주소 | 북구 칠곡중앙대로 52길 |
| 진행도 | 배정완료 |
| 신고자 | 구자철(1996-10-28) |
| 가해자 | 나상호(1998-02-28) |
| 피해자 | 이창수(2001-03-05) 김테베(1990-09-28) 김인지(2002-02-02) |
| 보고서 | 북구경찰서 형사과 1팀(작성일:2022-11-27 19:27:36) |

- 사건의 신고자, 가해자, 피해자, 보고서는 각 링크를 통해 상세로 이동
- 사용자가 사건을 맡은 팀 소속일 경우 "수정하기" 버튼 활성화
- 관리자 수정하기 버튼 비활성화

<03.경찰관>

데이터베이스

3. 사건 수정 및 결과

< 수정 프로시저 >

CREATE PROCEDURE

update_Incident_Proc(

IN incidentId INT,

IN u_end_date datetime,

IN u_progress varchar(10)

)

BEGIN

UPDATE Incident

SET progress = IFNULL(u_progress, progress),

end_date = IFNULL(u_end_date, end_date)

WHERE incident_id = incidentId;

END \$\$

DELIMITER

- 사건 정보 중 종료일시, 진행도만 수정 가능
- UI 에서 "null"값 입력을 고려하여 null값 입력시 기존 값으로 저장.

<결과>

사건 상세 페이지

| | |
|----------|--|
| 사건 ID | 4 |
| 사건 종류 | 살인 |
| 팀 | 북구경찰서 형사과 1팀 2022-11-01 02:01:00 ~ 2024-11-01 02:01:00 |
| 시작일시 | 2022-11-01 02:01:00 |
| 종료일시 | |
| 사건 발생 주소 | 북구 칠곡중앙대로 52길 |
| 진행도 | 배정완료 ▼ |
| 신고자 | 구자철(1996-10-28) |
| 가해자 | 나상호(1998-02-28) |
| 피해자 | 이창수(2001-03-05) 김대배(1990-09-28) 김인지(2002-02-02) |
| 보고서 | 북구경찰서 형사과 1팀(작성일:2022-11-27 19:27:36) 북구경찰서 형사과 1팀(작성일:2022-11-27 20:35:50) |

수정완료

사건 수정 결과

사건 정보가 정상적으로 수정 완료되었습니다.

[<--홈으로](#)

4. (보고서, 신고자, 피해자, 가해자) 등록.php

1. 신고자, 피해자, 가해자

```

DELIMITER $$
CREATE PROCEDURE insertApplicant(IN Incident_Id INT, IN name varchar(10), IN birth DATE, IN phone VARCHAR(14)
, IN date DATETIME, IN gender VARCHAR(2), IN address VARCHAR(45))
BEGIN
    INSERT INTO Applicant (name, birth, phone, date, gender, address)
    VALUES (name, birth, phone, date, gender, address);
    UPDATE Incident SET Incident.applicant_id = (SELECT applicant_id FROM Applicant Where applicant_id = LAST_INSERT_ID())
    WHERE incident_id = Incident_Id;
END $$
DELIMITER ;

```

- 신고자를 등록하면 사건 테이블도 업데이트를 해줘야한다.
- 프로시저를 통해 쿼리문의 집합을 일괄 처리하고 쿼리 모듈화를 통해 편리하게 MySQL 운영

2. 보고서

```

DROP PROCEDURE IF EXISTS insertReport;
DELIMITER $$
CREATE PROCEDURE insertReport(IN title VARCHAR(45), IN content VARCHAR(45), IN Team_Id INT, IN Incident_Id INT)
BEGIN
    insert into Report (title, content, re_date, team_id, incident_id)
    values (title, content, default, Team_Id, Incident_Id);
END $$
DELIMITER ;

```

| Field | Type | Null | Key | Default | Extra |
|-------------|-------------|------|-----|-------------------|-------------------|
| report_id | int | NO | PRI | <u>NULL</u> | auto_increment |
| title | varchar(45) | NO | | <u>NULL</u> | |
| content | varchar(45) | NO | | <u>NULL</u> | |
| re_date | datetime | NO | | CURRENT_TIMESTAMP | DEFAULT_GENERATED |
| ch_date | datetime | YES | | <u>NULL</u> | |
| team_id | int | NO | MUL | <u>NULL</u> | |
| incident_id | int | NO | | <u>NULL</u> | |

- 보고서 테이블 중 작성 시간 속성값은 default 값을 전달해주면 자동으로 현재 시간 값을 저장하도록 설계

4. (보고서, 신고자, 피해자, 가해자) 등록.ph

p

3. 결과

가해자 정보 등록

이름:

주소:

생년월일:

전화번호:

성별: ☒ 남성 ☐ 여성

!

 다음 옵션 중 하나를 선택하세요.

가해자 정보 등록 결과

가해자 정보 정상 등록 완료.

확인

수사 보고서 작성

제목:

!

 이 입력란을 작성하세요.

등록하기

수사 보고서 작성 결과

보고서 정상 등록 완료.

확인

- TABLE 에서 NOT NULL에 해당하는 속성값은 HTML required를 통해 반드시 값을 입력해야 등록할 수 있도록 설계

4. (보고서, 신고자, 피해자, 가해자) 등록.php

1. 신고자, 피해자, 가해자

```

DROP PROCEDURE IF EXISTS searchApplicant;
DELIMITER $$
CREATE PROCEDURE searchApplicant(IN Applicant_Id INT)
BEGIN
    SELECT * FROM Applicant WHERE Applicant.applicant_id = Applicant_Id;
END $$
DELIMITER ;

```

2. 보고서

```

DELIMITER $$
CREATE PROCEDURE searchReport(IN Report_Id INT)
BEGIN
    select r.report_id, CONCAT(s.name, " ", d.name, " ", t.name) as writer, re_date, ch_date, r.title, r.content
    from Report r join Team_Incident ti on r.team_id = ti.team_id and r.incident_id = ti.incident_id
    join Team t on t.team_id = ti.team_id
    join Department d on d.department_id = t.department_id
    join Station s on d.station_id = s.station_id
    where r.report_id = Report_Id;
END $$
DELIMITER ;

```

3. 결과

신고자 정보 상세

| | |
|------|---------------------|
| 이름 | 구자철 |
| 주소 | 북구 123로 크래프트 1동 |
| 생년월일 | 1996-10-28 |
| 전화번호 | 010-9999-9999 |
| 성별 | 남성 |
| 신고일시 | 2022-11-01 01:47:00 |

수정하기 |

신고자 정보 상세

| | |
|------|---------------------|
| 이름 | 황진영 |
| 주소 | 중구 중앙대로 미래빌 6동 |
| 생년월일 | 2022-09-13 |
| 전화번호 | 010-2452-2222 |
| 성별 | 여성 |
| 신고일시 | 2022-04-04 13:20:16 |

사용자가 사건을 맡은 팀의 소속일 경우 "수정하기" 버튼 활성화

수사 보고서 상세 페이지

| | |
|-----|---------------------|
| 작성자 | 북구경찰서 형사과 1팀 |
| 작성일 | 2022-11-27 19:27:36 |
| 제목 | 1차 보고서 |

수사 내용 :

현재 유력한 증거가 없어 사건에 난항을 겪고 있음.

수정하기 |

6. (보고서, 신고자, 피해자, 가해자) 수정.ph

p

1. 신고자, 피해자, 가해자

```

DELIMITER $$
CREATE PROCEDURE updateVictim(IN Victim_Id INT, IN name varchar(10), IN birth DATE
, IN phone VARCHAR(14), IN gender VARCHAR(2), IN address VARCHAR(45))
BEGIN
UPDATE Victim SET name = name, birth = birth, phone = phone, gender = gender, address = address
WHERE Victim.victim_id = Victim_Id;
END $$
DELIMITER ;

```

2. 보고서

```

DROP PROCEDURE IF EXISTS updateReport;
DELIMITER $$
CREATE PROCEDURE updateReport(IN Report_Id INT, IN title VARCHAR(45), IN content VARCHAR(45))
BEGIN
update Report set title=title, content=content , ch_date = now() where report_id = Report_Id;
END $$
DELIMITER ;

```

- 보고서 테이블에서 보고서 수정 시 수정 시간 속성값을 현재 값으로 갱신하도록 설계

3. 결과

피해자 정보 상세

| | |
|------|----------------|
| 이름 | 김대배 |
| 주소 | 남구 회왕로 망아파트 1동 |
| 생년월일 | 1990-09-28 |
| 전화번호 | 010-8777-7777 |
| 성별 | 여성 |

수정하기

피해자 정보 수정

이름:
주소:
생년월일:
전화번호:
성별: ☒ 남성 ☐ 여성

수정완료

피해자 정보 상세

| | |
|------|----------------|
| 이름 | 김대배 |
| 주소 | 남구 회왕로 망아파트 1동 |
| 생년월일 | 1990-11-07 |
| 전화번호 | |
| 성별 | 남성 |

수정하기

수사 보고서 수정 페이지

작성일자

작성일자

작성일자

작성일자

작성일자

작성일자

작성일자

작성일자

작성일자

작성일자

수사 내용

수사 내용

수사 내용

수사 내용

수사 내용

수사 보고서 작성 결과

수사 보고서 작성 결과

수사 보고서 작성 결과

수사 보고서 작성 결과

수사 보고서 작성 결과

수사 보고서 작성 결과

수사 보고서 작성 결과

수사 보고서 작성 결과

수사 보고서 작성 결과

수사 보고서 작성 결과

수사 보고서 작성 결과

1. 사건 팀 추가.php

<사건 팀 추가 프로시저>

- 한 사건에 대해 팀을 더 추가하고 싶을때 사용
(공동수사)

```
CREATE PROCEDURE team_add_Proc(
    IN incident_id INT,
    IN s_name varchar(10),
    IN d_name varchar(10),
    IN t_name varchar(10),
    IN start datetime,
    IN end datetime
)
BEGIN
    insert into Team_Incident values((select t.team_id from Station s
    join Department d on s.station_id = d.station_id
    join Team t on d.department_id = t.department_id
    where s.name = s_name and d.name = d_name and t.name = t_name),
    incident_id, start, end);
END
```

사건 조회 결과

<--출으로

| 사건 ID | 종류 | 시작일시 | 종료일시 | 진행도 | 주소 | 팀 | |
|-------|----|---------------------|------|-----|-------------|----------|------|
| 3 | 사기 | 2022-04-04 13:20:16 | | 수사중 | 중구 중앙대로 ... | 중구경찰서 수사 | 팀 추가 |

중구경찰서 수사과 2팀 2022-04-04 13:20:16 ~ 2024-04-04 13:20:16

사건 팀 추가

서

부서

팀

시작일시

종료일시

등록하기

서구경찰서

수사과

1팀

2020-10-20 14:00:00

2024-12-20 15:25:00

<--출으로

| 사건 ID | 종류 | 시작일시 | 종료일시 | 진행도 | 주소 | 팀 | |
|-------|----|---------------------|------|-----|-------------|-------------|------|
| 3 | 사기 | 2022-04-04 13:20:16 | | 수사중 | 중구 중앙대로 ... | 서구경찰서 수사... | 팀 추가 |

서구경찰서 수사과 1팀 2020-10-20 14:00:00 ~ 2024-12-20 15:25:00
중구경찰서 수사과 2팀 2022-04-04 13:20:16 ~ 2024-04-04 13:20:16

<04.관리자>

데이터베이스

2. 경찰관 조회, 등록.php

<경찰관 조회 프로시저>

```
USE PoliceDB;
DROP PROCEDURE IF EXISTS Police_search_Proc;
DELIMITER $$
CREATE PROCEDURE Police_search_Proc(
    IN s_name varchar(10),
    IN d_name varchar(10),
    IN t_name varchar(10),
    IN name varchar(10)
)
BEGIN
    select u.user_id, u.password, u.name, u.birth, u.gender, u.phone, u.rank, u.address,
    s.name as station, d.name as department, t.name as team from Station s
    join Department d on s.station_id = d.station_id
    join Team t on d.department_id = t.department_id
    join User u on t.team_id = u.team_id
    where s.name = ifnull(s_name, s.name) and d.name = ifnull(d_name, d.name)
    and t.name = ifnull(t_name, t.name) and u.name like
    ifnull(concat('%', name, '%'), u.name);
END $$
DELIMITER ;
```

- "서,부서,팀, 이름" 으로 동적으로 선택 가능
- 선택하지 않은 값은 null값에 해당하고 "ifnull"를 사용하여 null값을 처리

<경찰관 등록 프로시저>

```
USE PoliceDB;
drop procedure if exists police_add;
delimiter $$
create procedure police_add(in s_name varchar(10),in d_name varchar(10),in t_name varchar(10),in p_name varchar(10)
,in id int,in pw varchar(45),in birth date,in gender varchar(2),in phone varchar(14),in p_rank varchar(4),in address varchar(45))
begin
insert into User values(id, pw, p_name, birth,gender, phone,p_rank, "Police", address,
(select team_id from Station s join Department d on s.station_id = d.station_id join Team t
on d.department_id = t.department_id where s.name = s_name and d.name = d_name and t.name = t_name));
end $$
delimiter ;
```

- HTML required 사용해 입력 시, null 값 처리 (not null인 컬럼에 대해서)
- "insert" 하기전 입력 받은 "서, 부서, 팀" 이름을 통해 "팀id" 값 조회

| | |
|------|---|
| 서 | 수정구경찰서 ▼ |
| 부서 | 여성 청소년과 ▼ |
| 팀 | 3팀 ▼ |
| 이름 | |
| ID | |
| 비밀번호 | |
| 생년월일 | 연도-월-일 □ |
| 성별 | <input type="radio"/> 남성 <input type="radio"/> 여성 |
| 전화번호 | 123-1234-1234 |
| 직급 | 중경 ▼ |
| 주소 | 북구 123로 12 삼성주조 |

| 아이디 | 비밀번호 | 이름 | 생년월일 | 성별 | 전화번호 | 직급 | 주소 | 서 | 부서 | 팀 | |
|-----|------|-----|------------|----|---------------|----|--------------------|-------|-----|----|---------|
| 25 | aaa | 고경휘 | 1999-01-15 | 남성 | 010-3371-6132 | 경사 | 중구 경상감영길 우유파아파트 1동 | 남구경찰서 | 수사과 | 1팀 | 수정 및 삭제 |
| 26 | aaa | 귀호승 | 1999-01-16 | 남성 | 010-3381-6132 | 경사 | 중구 경상감영길 우유파아파트 2동 | 남구경찰서 | 수사과 | 1팀 | 수정 및 삭제 |

<04.관리자>

데이터베이스

2. 경찰관 조회, 등록.php

<경찰관 상세 정보 검색 프로시저>

```
USE PoliceDB;
DROP PROCEDURE IF EXISTS Police_update_delete_Proc;
DELIMITER $$
CREATE PROCEDURE Police_update_delete_Proc(
    IN user_id int
)
BEGIN
    select u.user_id, u.password, u.name, u.birth, u.gender, u.phone, u.rank, u.address, s.name as station, d.name as department, t.name as team
    from Station s join Department d on s.station_id = d.station_id
    join Team t on d.department_id = t.department_id
    join User u on t.team_id = u.team_id
    where u.user_id = user_id;
END $$
DELIMITER ;
```

- 경찰관 Id 를 통한 검색

- 경찰관 조회 결과에서 "수정 및 삭제" 버튼 클릭시 user_id 를 통해 이동

| | |
|------|--|
| 서 | 남구경찰서 ▼ |
| 부서 | 수사과 ▼ |
| 팀 | 1팀 ▼ |
| ID | 25 |
| 비밀번호 | aaa |
| 이름 | 고경휘 |
| 생년월일 | 1999-01-15 □ |
| 성별 | <input checked="" type="radio"/> 남성 <input type="radio"/> 여성 |
| 전화번호 | 010-3371-6132 |
| 직급 | 경사 ▼ |
| 주소 | 충구 경상감영길 우울아파트 |

<경찰관 수정 프로시저>

```
CREATE PROCEDURE Police_update_Proc(
    IN n_name varchar(10), IN n_birth date,
    IN n_gender varchar(2), IN n_phone varchar(14),
    IN u_rank varchar(10), IN n_address varchar(45),
    IN s_name varchar(10), IN d_name varchar(10),
    IN t_name varchar(10), IN n_user_id int
)
BEGIN
    Update User u set name = n_name, birth = n_birth, gender = n_gender,
    phone = n_phone,
    u_rank = u_rank, address = n_address,
    team_id = (select team_id from Station s
    join Department d on s.station_id = d.station_id
    join Team t on d.department_id = t.department_id
    where s.name = s_name and d.name = d_name and t.name = t_name)
    where user_id = n_user_id;
END $$
DELIMITER ;
```

- "required"를 통해 null값 처리

- update 하기전 "서,부서,팀"

이름을 통한 팀 Id 값 추출

- 모든 컬럼 수정 가능

<경찰관 삭제 프로시저>

```
USE PoliceDB;
DROP PROCEDURE IF EXISTS Police_delete_Proc;
DELIMITER $$
CREATE PROCEDURE Police_delete_Proc(
    IN d_user_id int
)
BEGIN
    delete from User where user_id = d_user_id;
END $$
DELIMITER ;
```

등록 순서 : 1.신고자 등록 -> 2. 팀 검색 -> 3. 팀 선택 및 정보 기입 -> 4. 사건 정보 기입

<1.신고자 등록 프로시저>

```
insert into Applicant  
values(null, name2, birth2, phone2, date2, gender2, address2);
```

1. 신고자 등록

신고자 없음

| | |
|------|--|
| 이름 | 김민자 |
| 주소 | 대구광역시 을하서로59 |
| 생년월일 | 2001-07-04 <input type="text"/> |
| 전화번호 | 010-4654-4564 |
| 신고일시 | 2022-11-17 오전 05:12 <input type="text"/> |
| 성별 | <input checked="" type="radio"/> 남성 <input type="radio"/> 여성 |

다음

- 신고자가 있는 경우 or 신고자가 없는 경우

<2. 팀검색 프로시저>

```
CREATE PROCEDURE incidentAdd3 (  
in s_name varchar(10),  
in d_name varchar(10))  
begin  
select t.team_id, t.name, (count(*)-1) as count  
from Station s join Department d on s.station_id = d.station_id  
join Team t on d.department_id = t.department_id  
left join Team_Incident ti on t.team_id = ti.team_id  
where s.name = s_name and d.name = d_name  
group by t.team_id;
```

- 서, 부서 이름을 검색
- 앞서 신고자 정보가 "신고자등록.php"에서 신고자 정보가 있을시 신고자 정보 등록

<3. 팀 검색 결과 및 정보 기입 프로시저>

```
create procedure incidentAdd3(in s_name varchar(10),in d_name varchar(10))
begin
    select t.team_id, t.name,
    SUM(case when ti.team_id is null or i.progress = "수사완료"
    then 0 else 1 end) as count
    from Station s join Department d on s.station_id = d.station_id
    join Team t on d.department_id = t.department_id
    left join Team_Incident ti on t.team_id = ti.team_id
    left join Incident i on i.incident_id = ti.incident_id
    where s.name = s_name and d.name = d_name
    group by t.team_id;

end $$
delimiter ;
```

- 앞서 입력받은 "서, 부서" 이름을 통해 팀리스트와 현재 팀이 처리하고 있는 사건 개수 추출.

2.2 팀선택

| | |
|------|------------------|
| 팀 | 1팀(1건) ▾ |
| 시작일시 | 1팀(1건) |
| 종료일시 | 2팀(2건) 3팀(0건) |

```
<select name="team_number" size="1">
<?php
for($j=0; $j<$count; $j=$j+1) {
    echo "<option value='". $team_ids[$j]. "'>". $team_names[$j].
        "(".$team_count[$j]. "건)" </option>>";
}
?>
</select>
```

- 팀id와 팀시작일시, 종료일시를 입력받아 "팀_사건" 테이블에 등록

<4. 사건 정보 기입 프로시저>

1) 앞서 신고자가 없는 경우

```
drop procedure if exists incident_add_0;
delimiter $$
create procedure incident_add_0(
in type varchar(10),in a_start datetime,in Address varchar(45),
in T_id int,in t_start datetime,in t_end datetime)
begin
insert into Incident
values(null, type, "배정완료", a_start, null, null, Address);

insert into Team_Incident
values(T_id, (select i.incident_id from Incident i
where i.incident_id = last_insert_id()),t_start, t_end);
end $$
delimiter ;
```

- 사건 테이블에 사건을 먼저 등록
- 추후 "팀_사건" 테이블에 앞서 등록한 "사건id" 를 가져온 후 팀_사건 테이블에 정보 등록
- 신고자가 없기 때문에 해당 컬럼은 null로 입력

<4. 사건 정보 기입 프로시저>

1) 앞서 신고자가 있는 경우

```
create procedure incident_add_1(  
in type varchar(10),in a_start datetime,  
in Address varchar(45),in T_id int,  
in t_start datetime,in t_end datetime)  
begin  
insert into Incident  
values(null, type, "배정완료", a_start, null,  
(select a.applicant_id from Applicant a  
where a.applicant_id = last_insert_id()), Address);  
  
insert into Team_Incident  
values(T_id,  
(select i.incident_id from Incident i  
where i.incident_id = last_insert_id()), t_start, t_end);  
end $$  
delimiter ;
```

- 앞서 프로시저와 비슷. 차이점은 사건등록시 신고자 컬럼에 이전 등록한 신고자Id값을 검색하여 넣음.

< 서, 부서,팀별 통계 결과를 받아오는 프로시저>

```
USE PoliceDB;

/*{서}, {서,부서}, {서,부서,팀}으로 총 3가지로 선택 가능
drop procedure if exists statistics1;
delimiter $$
create procedure statistics1(in s_n boolean,in d_n boolean,in t_n boolean)
begin
select
trim(concat(if(s_n is true, a.sname, " "),",",if(d_n is true, a.dname, " "),",",if(t_n is true, a.tname, " "))) as 'Unit',
(sum(case when a.type='살인' then 1 else 0 end)) as '살인',
(sum(case when a.type='사기' then 1 else 0 end)) as '사기',
(sum(case when a.type='강도' then 1 else 0 end)) as '강도',
(sum(case when a.type='폭행' then 1 else 0 end)) as '폭행',
(sum(case when a.type='성폭행' then 1 else 0 end)) as '성폭행',
(sum(case when a.type='도둑' then 1 else 0 end)) as '도둑'
from
(
select s.name as sname , d.name as dname , t.name as tname, i.type
from Incident i
join Team_Incident ti on i.incident_id= ti.incident_id
join Team t on ti.team_id = t.team_id
join Department d on d.department_id = t.department_id
join Station s on s.station_id = d.station_id
group by
if(s_n is true, s.name, s.name),if(d_n is true, d.name, s.name),if(t_n is true, t.name, s.name),i.incident_id
)a
group by
if(s_n is true, a.sname, a.sname),if(d_n is true, a.dname, a.sname),if(t_n is true, a.tname, a.sname)order by a.sname;
end $$
delimiter ;
```

- 최소 통계 범위 그룹을 위해서 "서"는 고정값으로 설정

- "부서, 팀" 을 동적으로 선택 가능, 선택하지 않은 값은 null값으로 처리

- 통계 결과를 "사건 타입" 별로 추출하기 위해 동적으로 선택한(부서,팀)를 서브쿼리를 사용하여 그룹핑을 먼저 시행.

- 부서만 선택한 경우 부서내에 있는 팀들중 같은 사건을 맡고 있는 상황은 같은 사건으로 개수를 처리해야하므로 서브쿼리내에서 마지막 그룹핑을 "사건ID"로 설정

< 지역별 통계 결과를 받아오는 프로시저>

```
USE PoliceDB;
# (구), (구, 로/동) 으로 선택 가능
drop procedure if exists statistics2;
delimiter $$
create procedure statistics2(in a_s boolean,in a_c boolean)
begin
select
if(a_c is true, SUBSTRING_INDEX(i.address," ", 2), SUBSTRING_INDEX(i.address," ", 1)) as 'address',
#,SUBSTRING_INDEX(i.address," ", 1) as '구',
sum(case when i.type='살인' then 1 else 0 end) as '살인',
sum(case when i.type='사기' then 1 else 0 end ) as '사기',
sum(case when i.type='음주' then 1 else 0 end ) as '음주',
sum(case when i.type='폭행' then 1 else 0 end ) as '폭행',
sum(case when i.type='성폭행' then 1 else 0 end ) as '성폭행',
sum(case when i.type='절도' then 1 else 0 end ) as '절도'
from Incident i
group by
if(a_c is true, SUBSTRING_INDEX(i.address," ", 2), SUBSTRING_INDEX(i.address," ", 1))
order by SUBSTRING_INDEX(i.address," ", 1); #구 기준으로 정렬
end $$
delimiter ;
```

- 최소 통계 범위 그룹을 위해서 "구"는 고정값으로 설정

- "로/동" 을 동적으로 선택 가능, 선택하지 않은 값은 null값으로 처리

- 통계 결과를 "사건 타입" 별로 추출하기 위해 선택한 값에 따라 그룹핑을 시행