

# Task 2

## 1. 创建基本表

```
CREATE TABLE STUDENT (  
    sno INT,  
    sname VARCHAR(10),  
    ssex VARCHAR(10),  
    sage INT  
);  
CREATE TABLE COURSE (  
    cno INT,  
    cname VARCHAR(10),  
    credit DEC(10,2)  
);  
CREATE TABLE ELECTIVE (  
    sno INT,  
    cno INT,  
    grade INT  
);
```

## 2. 编写 SQL 语句完成以下内容

### (1) 查询学生编号为 10 的学生的姓名信息

```
SELECT *  
FROM STUDENT  
WHERE sno = 10;
```

### (2) 将 STUDENT 基本表中的学号设置为主键

```
ALTER TABLE STUDENT  
ADD PRIMARY KEY (sno);
```

### (3) 为 ELECTIVE 中的学生编号和课程编号创建 UNIQUE 索引

```
CREATE UNIQUE INDEX ELECTIVE_index  
ON ELECTIVE (sno,cno);
```

### (4) 创建一个视图，显示学生的姓名、课程名称以及获得的分数

```
CREATE VIEW stu_grade AS  
SELECT STUDENT.sname,COURSE.cname,grade  
FROM ELECTIVE  
JOIN STUDENT ON ELECTIVE.sno = STUDENT.sno  
JOIN COURSE ON ELECTIVE.cno = COURSE.cno;
```

## 3. 编写一个函数，返回某个学生的分数总和

```
CREATE FUNCTION sumGrade (  
s_sno INT  
RETURNS INT AS $$  
DECLARE sum_grade INT;  
BEGIN  
    SELECT SUM(grade) INTO sum_grade  
    FROM STUDENT  
    JOIN ELECTIVE  
    ON STUDENT.sno = ELECTIVE.sno  
    WHERE STUDENT.sno = s_sno  
    GROUP BY sno;  
    RETURN g_sum;  
END;  
$$ language plpgsql;
```

#### 4. 为 STUDENT 表创建一个触发器，当删除学生信息时，同步删除 ELECTIVE 表中学生的选课信息

```
CREATE FUNCTION deleteInfo()  
RETURNS TRIGGER AS $delete_student_info$  
BEGIN  
    DELETE ELECTIVE  
    WHERE sno = old.sno;  
END;  
$delete_student_info$ language plpgsql;  
CREATE TRIGGER delete_student_info  
AFTER DELETE ON STUDENT  
FOR EACH ROW EXECUTE PROCEDURE deleteInfo();
```