

Санкт-Петербургский национальный исследовательский университет

информационных технологий, механики и оптики

Кафедра вычислительной техники

Курсовая работа

по дисциплине «Базы данных»

Часть 3

Выполнили:

Нго Ши Тханг

Нгу Фыонг Ань

Группа Р3110

Преподаватель:

Письмак Алексей Евгеньевич

Санкт-Петербург, 2018

Ход работы:

```
CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;
USE `mydb` ;
```

```
-----
-- Table `mydb`.`Subway_Station`
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Subway_Station` (
  `Station_ID` INT NOT NULL,
  `Name` VARCHAR(45) NOT NULL,
  `Number_of_Entrances` INT NOT NULL,
  `Number_of_Wayout` INT NOT NULL,
  `Open_Time` TIME NOT NULL,
  `Close_Time` TIME NOT NULL,
  `Count_Train` INT NOT NULL,
  PRIMARY KEY (`Station_ID`));
```

```
-----
-- Table `mydb`.`Train`
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Train` (
  `Train_ID` INT NOT NULL,
  `Model` VARCHAR(255) NOT NULL,
  `Total_Seat` INT NOT NULL,
  `Subway_Station_Station_ID` INT NOT NULL,
  PRIMARY KEY (`Train_ID`, `Subway_Station_Station_ID`),
  INDEX `fk_Train_Subway_Station_idx` (`Subway_Station_Station_ID` ASC),
  CONSTRAINT `fk_Train_Subway_Station`
    FOREIGN KEY (`Subway_Station_Station_ID`)
      REFERENCES `mydb`.`Subway_Station` (`Station_ID`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION);
```

-- Table `mydb`.`Trip`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Trip` (  
  `Trip_ID` INT NULL,  
  `Arriving_Time` DATETIME NOT NULL,  
  `Departing_Time` DATETIME NOT NULL,  
  `Distance` INT NOT NULL,  
  `Departing_From` VARCHAR(45) NOT NULL,  
  `Arrive_To` VARCHAR(45) NOT NULL,  
  `Total_Seat` INT NOT NULL,  
  `Number_of_Station` INT NOT NULL,  
  `Train_Train_ID` INT NOT NULL,  
  `Train_Subway_Station_Station_ID` INT NOT NULL,  
  PRIMARY KEY (`Trip_ID`, `Train_Train_ID`, `Train_Subway_Station_Station_ID`),  
  INDEX `fk_Trip_Train1_idx` (`Train_Train_ID` ASC, `Train_Subway_Station_Station_ID`  
  ASC),  
  CONSTRAINT `fk_Trip_Train1`  
    FOREIGN KEY (`Train_Train_ID`, `Train_Subway_Station_Station_ID`)  
    REFERENCES `mydb`.`Train` (`Train_ID`, `Subway_Station_Station_ID`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION);
```

-- Table `mydb`.`Passenger`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Passenger` (  
  `Passenger_ID` INT NOT NULL,  
  `Fname` VARCHAR(45) NOT NULL,  
  `Mname` VARCHAR(45) NOT NULL,  
  `Lname` VARCHAR(45) NOT NULL,  
  `Age` BIT NOT NULL,  
  `Contact` VARCHAR(45) NOT NULL,  
  `Genger` VARCHAR(1) NOT NULL,  
  PRIMARY KEY (`Passenger_ID`));
```

-- Table `mydb`.`Position`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Position` (  
  `Position_ID` INT NOT NULL,  
  `Name` VARCHAR(255) NOT NULL,  
  `Requirement` VARCHAR(225) NULL,  
  `Salary` BIGINT NOT NULL,  
  `Number_of_Employees` INT NOT NULL,  
  PRIMARY KEY (`Position_ID`));
```

-- Table `mydb`.`Department`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Department` (  
  `Department_Location` VARCHAR(225) NOT NULL,  
  `Station_ID` INT NOT NULL,  
  `Name` VARCHAR(255) NOT NULL,  
  `Number_of_Employees` INT NOT NULL,  
  PRIMARY KEY (`Department_Location`, `Station_ID`),  
  INDEX `fk_Department_Subway_Station1_idx` (`Station_ID` ASC),  
  CONSTRAINT `fk_Department_Subway_Station1`  
    FOREIGN KEY (`Station_ID`)  
      REFERENCES `mydb`.`Subway_Station` (`Station_ID`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION);
```

-- Table `mydb`.`Manager`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Manager` (  
  `Manager_ID` INT NOT NULL,  
  `Name` VARCHAR(255) NOT NULL,  
  `Position` VARCHAR(225) NULL,  
  PRIMARY KEY (`Manager_ID`));
```

-- Table `mydb`.`Employees`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Employees` (  
  `Employee_ID` INT NOT NULL,  
  `Fname` VARCHAR(45) NOT NULL,  
  `Mname` VARCHAR(45) NOT NULL,  
  `Lname` VARCHAR(45) NOT NULL,  
  `Age` BIT NOT NULL,  
  `Gender` VARCHAR(1) NOT NULL,  
  `Position` VARCHAR(225) NOT NULL,  
  `Position_ID` INT NOT NULL,  
  `Department_Location` VARCHAR(225) NOT NULL,  
  `Manager_ID` INT NOT NULL,  
  PRIMARY KEY (`Employee_ID`, `Position_ID`, `Department_Location`, `Manager_ID`),  
  INDEX `fk_Employees_Position1_idx` (`Position_ID` ASC),  
  INDEX `fk_Employees_Department1_idx` (`Department_Location` ASC),  
  INDEX `fk_Employees_Manager1_idx` (`Manager_ID` ASC),  
  CONSTRAINT `fk_Employees_Position1`  
    FOREIGN KEY (`Position_ID`)  
      REFERENCES `mydb`.`Position` (`Position_ID`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,
```

```
CONSTRAINT `fk_Employees_Department1`  
  FOREIGN KEY (`Department_Location`)  
  REFERENCES `mydb`.`Department` (`Department_Location`)  
  ON DELETE NO ACTION  
  ON UPDATE NO ACTION,  
CONSTRAINT `fk_Employees_Manager1`  
  FOREIGN KEY (`Manager_ID`)  
  REFERENCES `mydb`.`Manager` (`Manager_ID`)  
  ON DELETE NO ACTION  
  ON UPDATE NO ACTION);
```

```
-----  
-- Table `mydb`.`Ticket`  
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Ticket` (  
  `Ticket_id` INT NOT NULL,  
  `Type_of_Ticket` VARCHAR(255) NOT NULL,  
  `Price` INT NULL,  
  `Number_of_Trip` INT NULL,  
  `Time_limit_day` INT NULL,  
  PRIMARY KEY (`Ticket_id`));
```

```
-----  
-- Table `mydb`.`Payment`  
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Payment` (  
  `Payment_ID` INT NOT NULL,  
  `Passenger_ID` INT NOT NULL,  
  `Time` DATETIME NULL,  
  `Money` INT NULL,  
  `Type_of_Payment` VARCHAR(225) NULL,  
  `Ticket_id` INT NOT NULL,  
  PRIMARY KEY (`Payment_ID`, `Passenger_ID`, `Ticket_id`),  
  INDEX `fk_Payment_Passenger1_idx` (`Passenger_ID` ASC),  
  INDEX `fk_Payment_Ticket1_idx` (`Ticket_id` ASC),  
  CONSTRAINT `fk_Payment_Passenger1`  
    FOREIGN KEY (`Passenger_ID`)  
    REFERENCES `mydb`.`Passenger` (`Passenger_ID`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_Payment_Ticket1`  
    FOREIGN KEY (`Ticket_id`)  
    REFERENCES `mydb`.`Ticket` (`Ticket_id`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION);
```

```
-----
```

-- Table `mydb`.`Trip_has_Passenger`

CREATE TABLE IF NOT EXISTS `mydb`.`Trip_has_Passenger` (
 `Trip_ID` INT NOT NULL,
 `Train_ID` INT NOT NULL,
 `Station_ID` INT NOT NULL,
 `Passenger_ID` INT NOT NULL,
 PRIMARY KEY (`Trip_ID`, `Train_ID`, `Station_ID`, `Passenger_ID`),
 INDEX `fk_Trip_has_Passenger_Passenger1_idx` (`Passenger_ID` ASC),
 INDEX `fk_Trip_has_Passenger_Trip1_idx` (`Trip_ID` ASC, `Train_ID` ASC, `Station_ID`
ASC),
 CONSTRAINT `fk_Trip_has_Passenger_Trip1`
 FOREIGN KEY (`Trip_ID`, `Train_ID`, `Station_ID`)
 REFERENCES `mydb`.`Trip` (`Trip_ID`, `Train_ID`,
`Train_Subway_Station_ID`)
 ON DELETE NO ACTION
 ON UPDATE NO ACTION,
 CONSTRAINT `fk_Trip_has_Passenger_Passenger1`
 FOREIGN KEY (`Passenger_ID`)
 REFERENCES `mydb`.`Passenger` (`Passenger_ID`)
 ON DELETE NO ACTION
 ON UPDATE NO ACTION);