

### ■ 전화번호 관리 프로그램 01단계 문제

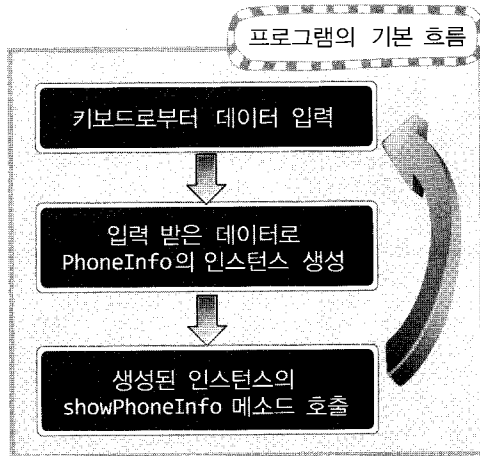
클래스를 하나 정의하자. 클래스의 이름은 PhoneInfo이다. 그리고 이 클래스에는 다음의 데이터들이 문자열의 형태로 저장 가능해야 하며, 저장된 데이터의 적절한 출력이 가능하도록 메소드도 정의되어야 한다.

- 이름            name            String
- 전화번호        phoneNumber   String
- 생년월일       birthday        String

단 생년월일 정보는 저장할 수도, 저장하지 않을 수도 있게끔 생성자를 정의하자. 생년월일 정보를 모른다고 해서 저장이 불가능하면 말이 되겠는가? 그리고 정의된 클래스의 확인을 위한 main 메소드도 간단히 정의하자.

## 12-3

# 단계별 프로젝트 : 전화번호 관리 프로그램 02단계



### ❖ 실행의 예

C:\>java PhoneBookVer02

선택하세요...

1. 데이터 입력
2. 프로그램 종료

선택 : 1

이름 : 홍길동

전화번호 : 222-3333

생년월일 : 99년 12월 25일생

입력된 정보 출력...

name : 홍길동

phone : 222-3333

birth : 99년 12월 25일생

선택하세요...

1. 데이터 입력
2. 프로그램 종료

선택 :

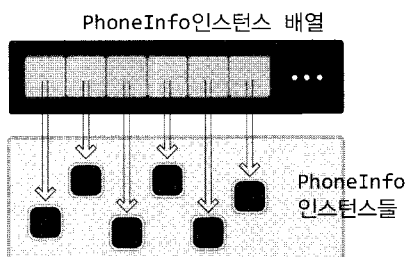
배열을 이용해서, 프로그램 사용자가 입력하는 정보가 최대 100개까지 유지되도록 프로그램을 변경하자. 그리고 다음의 기능을 추가로 삽입하자.

- 저장 이름, 전화번호, 생년월일 정보를 대상으로 저장의 과정을 진행한다.
- 검색 이름을 기준으로 데이터를 찾아서 해당 데이터의 정보를 출력해준다.
- 삭제 이름을 기준으로 데이터를 찾아서 삭제의 과정을 진행한다.

우리가 저장하는 데이터 중에는 동명이인(同名異人)의 데이터가 존재하지 않는다고 가정하겠다. 그리고 데이터의 삭제는 다음의 형태로 이뤄져야 한다.

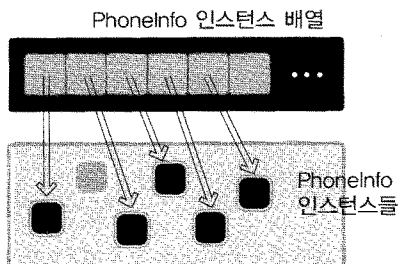
“배열의 중간에 저장된 데이터를 삭제할 경우에는, 해당 요소의 뒤에 저장된 요소들을 한 칸씩 앞으로 이동시키는 형태로 삭제를 진행한다.”

예를 들어서 다음의 형태로 배열이 구성되었다고 가정해보자.



[그림 13-13 : 삭제의 과정 1]

이 상황에서 두 번째 요소에 저장된 데이터를 삭제하는 경우에는 인스턴스 배열에 저장된 참조의 관계를 다음과 같이 변경하면 된다.



[그림 13-14 : 삭제의 과정 2]

위의 배열상황에서 다시 네 번째 배열 요소에 저장된 데이터를 삭제하는 경우에는 인스턴스 배열에 저장된 참조의 관계를 다음과 같이 변경하면 된다.

## ■ 전화번호 관리 프로그램 03단계 프로그램의 실행 예

데이터의 입력, 검색, 삭제 기능별 실행의 예를 각각 보이도록 하겠다. 여기서는 비록 하나의 데이터를 기준으로 실행의 예를 보이지만, 문제에서 제시했듯이, 프로그램은 둘 이상의 데이터를 기준으로 동작 가능해야 한다.

### ❖ 실행의 예1 : 데이터의 입력

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 : 1

데이터 입력을 시작합니다..

이름 : 이인석

전화번호 : 555-7777

생년월일 : 99년 04월 07일생

데이터 입력이 완료되었습니다.

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 :

#### ❖ 실행의 예2 : 데이터의 검색

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 : 2

데이터 검색을 시작합니다..

이름 : **이인석**

name : 이인석

phone : 555-7777

birth : 99년 04월 07일생

데이터 검색이 완료되었습니다.

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 :

#### ❖ 실행의 예3 : 데이터의 삭제

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 : 3

데이터 삭제를 시작합니다..

이름 : **이인석**

데이터 삭제가 완료되었습니다.

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 :

전화번호 관리 프로그램 03단계를 통해서 프로젝트의 기본 틀을 적당히 완성하였다. 따라서 이를 토대로 프로젝트를 진행하면, 필자가 요구하는 유형에서 크게 벗어나지 않을 것이다.

## ■ 전화번호 관리 프로그램 04단계 문제

본문에서 언급했던 내용과 유사한 형태의 시나리오를 프로젝트에 반영해 보겠다. 앞서 프로젝트 03단계에서는 PhoneInfo 클래스를 다음과 같이 정의하였다.

```
class PhoneInfo
{
    String name;
    String phoneNumber;
    String birth;    // birth, 그리고 birth와 관련된 코드를 삭제합시다!
    . . . .
}
```

이중에서 인스턴스 변수 birth는 생성자의 오버로딩을 실습하기 위한 목적으로 삽입한 것이니, 이번 프로젝트를 기점으로 이 변수를 삭제하기로 하겠다. 물론 관련 메소드들도 수정을 하자. 이 과정에서 PhoneInfo 클래스가 많이 가벼워질 것이다. 대신 다음 두 클래스를 추가로 삽입하자.

- PhoneUnivInfo                  대학 동기들의 전화번호 저장
- PhoneCompanyInfo            회사 동료들의 전화번호 저장

각각의 클래스에 정의되어야 하는 인스턴스 변수의 종류는 다음과 같다.

```
PhoneUnivInfo
• 이름        name        String
• 전화번호    phoneNumber String
• 전공        major       String
• 학년        year        int
```

```
PhoneCompanyInfo
• 이름        name        String
• 전화번호    phoneNumber String
• 회사        company    String
```

## ■ 전화번호 관리 프로그램 04단계 프로그램의 실행 예

03단계 프로젝트와는 데이터 입력의 과정에서만 차이를 보인다. 따라서 이를 중심으로 실행의 예를 보이도록 하겠다.

### ❖ 실행의 예1 : 일반적인 데이터 입력의 예

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 : 1

데이터 입력을 시작합니다..

1. 일반, 2. 대학, 3. 회사

선택>> 1

이름 : 이운영

전화번호 : 555-7777

데이터 입력이 완료되었습니다.

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 :

❖ 실행의 예2 : 대학동기의 전화번호 입력의 예

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제

데이터 입력을 시작합니다..

1. 일반, 2. 대학, 3. 회사

선택>> 2

이름 : 정대학

전화번호 : 444-9999

전공 : 컴퓨터공학

학년 : 3

데이터 입력이 완료되었습니다.

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 :



❖ 실행의 예3 : 회사동료의 전화번호 입력의 예

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 : 1

데이터 입력을 시작합니다..

1. 일반, 2. 대학, 3. 회사

선택>> 3

이름 : 최회사

전화번호 : 8282-1234

회사 : 맛있는 비타민

데이터 입력이 완료되었습니다.

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 :

❖ 실행의 예4 : 대학동기의 정보 조회의 예

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 : 2

데이터 검색을 시작합니다..

이름 : 정대학

name : 정대학

phone : 444-9999

major : 컴퓨터공학

year : 3

데이터 검색이 완료되었습니다.

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 :

## ■ 전화번호 관리 프로그램 06단계 문제

우리가 직접 예외의 상황을 정의하고, 해당 예외의 표현을 위한 예외 클래스를 정의해서 단계별 프로젝트에 반영해보고자 한다. 필자가 제안하는 예외의 상황은 다음 두 가지이다.

- ▶ 초기 메뉴 선택에서 1, 2, 3, 4 이외의 값을 입력하는 예외상황
- ▶ 데이터 입력의 과정에서(inputData 메소드 내에서) 1, 2, 3 이외의 값을 입력하는 예외상황

이 두 가지 예외상황 모두 그 유형이 동일하니, MenuChoiceException이라는 이름의 예외 클래스를 하나 정의해서 위의 두 상황에 모두 활용하기로 하겠다. 그리고 위의 두 가지 중 어느 예외상황이 발생을 하건, 프로그램의 흐름은 '초기 메뉴 선택'으로 이동하는 것을 원칙으로 하자.

## ■ 전화번호 관리 프로그램 06단계 프로그램의 실행 예

총 두 가지 예외상황을 정의하였는데, 어떠한 예외가 발생을 하건 프로그램의 흐름은 '초기 메뉴 선택'으로 이동을 해야 한다. 아래 실행의 예를 통해서 이를 보이도록 하겠다.

### ❖ 실행의 예1 : 초기 메뉴 선택에서의 예외상황

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 : 5

5에 해당하는 선택은 존재하지 않습니다.  
메뉴 선택을 처음부터 다시 진행합니다.

선택하세요...

1. 데이터 입력
2. 데이터 검색
3. 데이터 삭제
4. 프로그램 종료

선택 :