

## 실습문제 7-4

1

- 교재 435페이지, 실습문제 4
- 프로젝트 이름: `prac07_04`
- 클래스 이름: `RainfallStatistics_학번`

## 실습문제 7-4 소스

2

```

필요한 클래스들 import      // 예제 7-5 참조

RainfallStatistics_학번 클래스 {

    // print(): 정수만 다루는 벡터인 n 내의 모든 원소를 출력하고, 평균도 구하여 출력함
    //      예제 7-4를 참조할 것
    print(v) 함수 { // v는 정수만 다루는 벡터의 변수임, 리턴 값 없음
        // print()는 static 함수인 main에서 호출할 수 있게 선언되어야 함, 실습 3-3 참조
        원소의 합을 저장할 정수형 변수 sum의 선언 및 초기화
        정수만 다루는 Iterator 변수 it 선언 및 초기화 (v의 Iterator 객체를 구해서 초기화)
        while (it가 다음 원소를 가지고 있는 동안) {
            정수형 변수 n 선언 및 초기화(it의 다음 원소로)
            n 값 출력
            n을 sum에 더함
        }
        줄바꾸기 출력
        "현재 평균 "+평균_값 출력 // 평균_값: v의 원소의 개수 알아야 함, 표 7-1 참조
    }
    main() 함수 // 뒤 페이지 참조
}

```

## 실습문제 7-4 소스

3

```
main() 함수 {
    scanner 변수 선언 및 초기화;
    정수만 다루는 Vector 변수 v 선언 및 초기화(new로 객체 생성) // 예제 7-1 참조
    무한 루프 {
        "강수량 입력 (0 입력시 종료)>> " 출력
        정수형 변수 n 선언 및 초기화(scanner에서 정수 값 하나 읽어서)
        n 값 체크하여 무한 루프 탈출 (if, break 사용)
        n을 v에 추가
        print() 함수 호출
    }
    scanner 닫기
}
```

## 실습문제 7-6

4

- 교재 437페이지, 실습문제 6
- 프로젝트 이름: [prac07\\_06](#)
- 클래스 이름: [LocationApp\\_학번](#)
  - ▣ Location 클래스 작성
  - ▣ LocationManager 클래스 작성
  - ▣ LocationApp\_학번 클래스 작성
    - 이 클래스는 main 함수만 가지고 있음

## 실습문제 7-6 소스

5

```

필요한 클래스들 import      // 예제 7-5 참조
// main이 포함된 클래스만 public으로 선언하고 나머지 클래스는
// 디폴트 클래스로 선언, 즉 public 선언하지 말 것
Location 클래스 {
    문자열 변수 city, double형 변수 longitude(경도)와 latitude(위도) 선언
    // 위 세 변수는 외부에서 접근할 수 없고 이 클래스 내에서만 접근 가능
    생성자(city, longitude, latitude) {
        // 세 함수 인자를 각각 상응하는 클래스 멤버에 저장(this 사용)
    }
    printLocation() 함수 정의 { // 외부에서 호출 가능; 리턴 값 없음
        도시이름, 경도, 위도를 한 줄에 출력; // 출력 시 각 값(열) 사이에 탭("\t")을
        출력하여 각 열이 같은 위치에 출력되게 하라
    }
}
LocationManager 클래스 // 뒤 페이지 참조

LocationApp_학번 클래스 {
    main 함수 {
        LocationManager 변수인 man 선언 및 초기화(new로 생성)
        man 객체의 멤버 함수인 run() 호출
    }
}

```

## 실습문제 7-6 소스

6

```

LocationManager 클래스 {
    scanner 변수 선언 및 초기화 // 예제 7-5 참조
    HashMap<String, Location>형의 변수 dept 선언 및 초기화(new로 생성)//예제7-5

    // 아래 모든 함수들의 리턴 값은 없음
    read() 함수 {
        // 뒤 페이지 참조
    }
    printAll() 함수 {
        // 뒤 페이지 참조
    }
    processQuery() 함수 {
        // 뒤 페이지 참조
    }
    // 위의 모든 데이터 및 함수는 외부에서 접근 못하고
    // 이 클래스 내에서만 접근 가능함
    run() 함수 { // 외부에서 호출 가능
        위 세 함수들을 순서적으로 호출한다.
    }
}

```

## 실습문제 7-6 소스

7

```

read() 함수 {
    "도시,경도,위도를 입력하세요." 출력
    네 번 반복 {
        ">> " 출력
        문자열 변수 text 선언 및 초기화(scanner에서 한 줄을 통째로 읽어) 표 2-4
        StringTokenizer 변수 st 선언 및 초기화(new로 객체 생성; 생성자에 인자로 text와
        구분자 ","를 인자로 넘겨 줌 // 표 6-8, 6-9 참조
        문자열 변수 city 선언 및 초기화(다음 토큰을 얻은 후 앞뒤 공백 문자를 제거한 후
        초기화 함; 예, String city = st.nextToken().trim(); // 표 6-9, 6-5
        문자열 변수 log와 lat의 선언 및 초기화(위 city처럼 다음 토큰을 얻은 후 앞뒤 공
        백 문자를 제거 한 후 초기화)
        // log와 lat에는 실수 값이 문자열로 저장되어 있으므로
        // 이를 실수형으로 변환하여야 함;
        // 예제 3-13과 교재 353 페이지 [문자열을 기본 타입으로 변환] 참조
        실수형 변수 logitude와 latitude 선언 및 초기화(log와 lat 문자열을 실수로 변환
        하여 각각 초기화 함)
        Location 변수 loc 선언 및 초기화 (new로 객체 생성; 이때 생성자에 city, logitude,
        latitude를 인자로 넘겨 줌)
        해시맵(LocationManager의 dept 멤버 변수)에 (city와 loc) 요소 삽입 // 예제 7-5
    }
}

```

## 실습문제 7-6 소스

8

```

printAll() 함수 {
    // 교재 414 페이지 [해시 맵의 전체 검색] 부분을 참조하여 이 함수를 구현하라.
    문자열 집합 변수 keys 선언 및 초기화(해시 맵 dep에서 키 집합을 얻어서)
    문자열 Iterator 변수 it 선언 및 초기화(집합 keys에서 iterator 구해서)
    "-----" 출력
    루프 돌기(it가 다음 값을 가지고 있는 동안) {
        도시 이름 변수 city 선언 및 초기화(it에서 다음 키 값을 얻어)
        Location 변수 loc 선언 및 초기화(해시 맵 dept에서 도시 이름 city를 '키' 값으로
        주고 상응하는 '값' 얻어) // 교재 414 페이지
        // [해시 맵의 전체 검색] 부분을 참조하되 데이터 타입과 변수이름만 변경
        도시이름, 경도, 위도를 출력하는 loc의 멤버 함수 호출
    }
    "-----" 출력
}

```

## 실습문제 7-6 소스

9

```
processQuery() 함수 {
    무한 루프 {
        "도시 이름 >> " 출력
        도시 이름 변수 city 선언 및 초기화(scanner에서 한 행을 통째로 읽어) 표 2-4
        도시 이름이 "그만"과 동일하면 (객체가 같은지 체크, equals() 함수) 리턴
        Location 변수 loc 선언 및 초기화(해시 맵 dept에서 도시 이름 city를 키 값으로
        주고 상응하는 '값' 얻어 초기화) // 앞 페이지 printAll() 참조
        도시가 해시맵에 없다면 (loc 값이 무엇과 같을까?) // 표 7-4 get() 함수 참조
            city+"는 없습니다." 출력
        그렇지 않다면 // 해시맵에서 city에 상응하는 '값' loc이 존재
            도시이름, 경도, 위도를 출력하는 loc의 함수 호출 // Location 클래스 참조
    }
}
```