

9장. 실전연습4

박 종 혁 교수
(서울과기대 컴퓨터공학과)

Tel: 970-6702
Email: jhpark1@seoultech.ac.kr

실습문제(1)

- Person 클래스를 설계하라. Person 클래스는 이름, 주소, 전화 번호를 멤버 변수로 가진다. 하나 이상의 생성자를 정의하고 각 멤버 변수에 대하여 접근자와 설정자 함수를 작성하라. 이어서 Person을 상속받아서 Customer를 작성하여 보자. Customer는 고객 번호(id)와 마일리지(mileage)를 멤버 변수로 가지고 있다. 한 개 이상의 생성자를 작성하고 적절한 접근자 함수와 설정자 함수를 작성하라.

```
4 class Person{
5     string name;
6     string address;
7     string tel;
8 };
```

```
이름을 입력하세요 :김철수
주소를 입력하세요 :서울시
연락처를 입력하세요 :010-1101-2202
ID를 입력하세요 :110
마일리지를 입력하세요 :1500
-----고객 정보-----
이름: 김철수
주소: 서울시
연락처: 010-1101-2202
고객ID: 110
마일리지: 1500
```

실습문제(1)

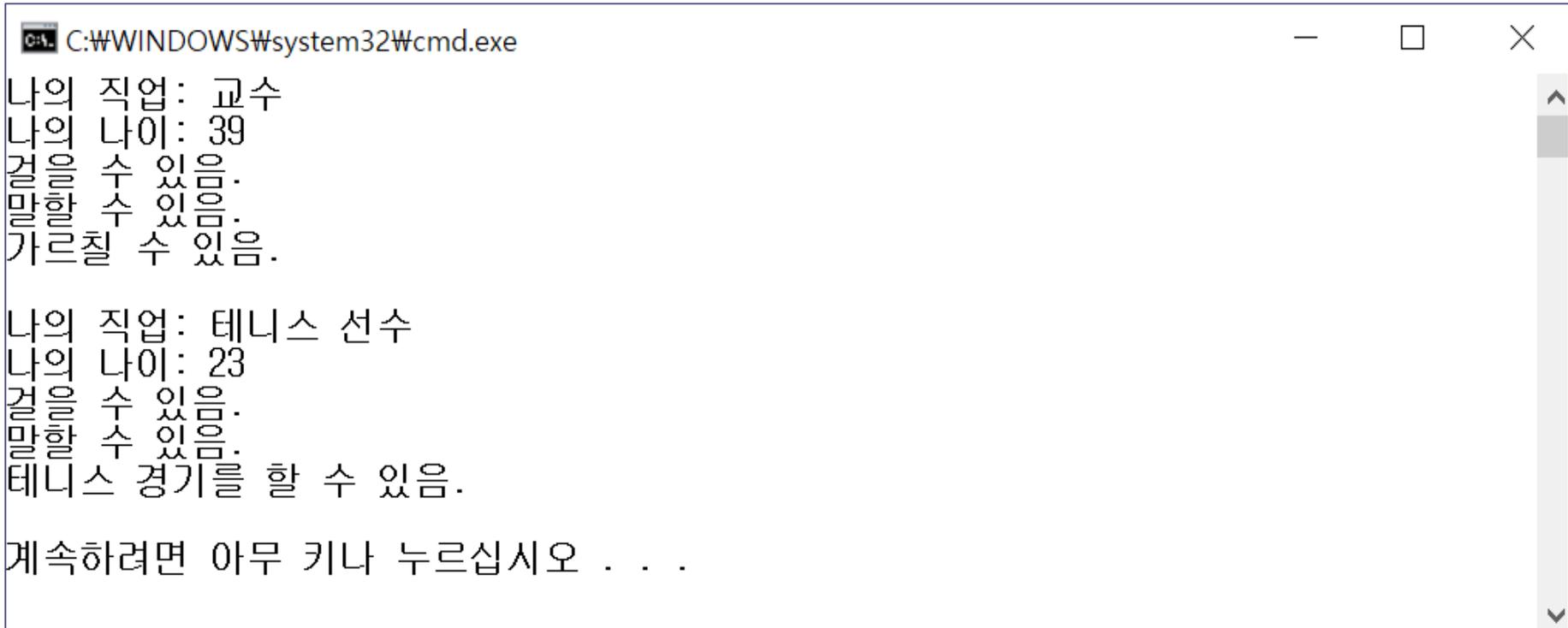
```
1 #include <iostream>
2 using namespace std;
3
4 class Person{
5     string name;
6     string address;
7     string tel;
8
9     public:
10    Person(string name1, string address1, string tel1){
11        name=name1;
12        address=address1;
13        tel=tel1;
14    }
15    void setName(string name1){
16        name=name1;
17    }
18    void setAddress(string address1){
19        address=address1;
20    }
21    void setTel(string tel1){
22        tel=tel1;
23    }
24    string getName(){
25        return name;
26    }
27    string getAddress(){
28        return address;
29    }
30    string getTel(){
31        return tel;
32    }
33 };
```

실습문제(1)

```
35 class Customer : public Person{
36     int ID;
37     int mileage;
38
39     public:
40     Customer(string name1, string address1, string tel1, int ID1, int mileage1):Person(name1, address1, tel1){
41         ID=ID1;
42         mileage=mileage1;
43     }
44     void show_Customer(){
45         cout << "-----고객 정보-----" << endl;
46         cout <<"이름: " << getName() << endl;
47         cout <<"주소: " << getAddress() << endl;
48         cout <<"연락처: " << getTel() << endl;
49         cout <<"고객ID: " << ID << endl;
50         cout <<"마일리지: " << mileage << endl;
51     }
52 };
53
54 int main(){
55     string name1, address1, tel1;
56     int ID1, mileage1;
57
58     cout << "이름을 입력하세요 :"; cin >> name1;
59     cout << "주소를 입력하세요 :"; cin >> address1;
60     cout << "연락처를 입력하세요 :"; cin >> tel1;
61     cout << "ID를 입력하세요 :"; cin >> ID1;
62     cout << "마일리지를 입력하세요 :"; cin >> mileage1;
63
64     Customer c(name1, address1, tel1, ID1, mileage1);
65
66     c.show_Customer();
67
68 }
```

실습문제(2)

- 다음과 같은 실행결과를 출력할 수 있도록 프로그램을 작성한다. (클래스의 상속 개념을 활용 하시오.)



```
C:\WINDOWS\system32\cmd.exe
나의 직업: 교수
나의 나이: 39
걸을 수 있음.
말할 수 있음.
가르칠 수 있음.

나의 직업: 테니스 선수
나의 나이: 23
걸을 수 있음.
말할 수 있음.
테니스 경기를 할 수 있음.

계속하려면 아무 키나 누르십시오 . . .
```

실습문제(3)

- 일반적인 도형을 나타내는 Shape라는 클래스에 추가로 getArea() 함수를 가상함수로 정의하라. getArea()는 도형의 면적을 구한다. Rect(사각형), Circle(원), Triangle(삼각형) 클래스는 Shape 클래스를 상속받아서 작성된다. Shape* 포인터 배열을 다양한 도형 객체로 채운다. 배열 안에 있는 도형들의 면적을 계산하는 프로그램을 작성해보자.

```
class Shape {  
    int x, y;  
    virtual double getArea() = 0;  
    ...  
};
```

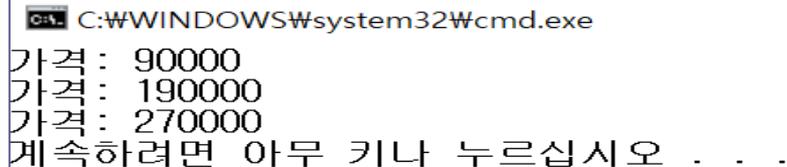


```
C:\WINDOWS\system32\cmd.exe  
도형 #0의 면적: 2747  
도형 #1의 면적: 14949.5  
도형 #2의 면적: 1798  
계속하려면 아무 키나 누르십시오 . . .
```

실습문제(4)

- 전자제품을 나타내는 HomeAppliance 클래스를 작성한다. HomeAppliance 클래스는 가격을 반환하는 getPrice() 라는 가상 함수를 가진다. Television 클래스와 Refrigerator 클래스는 HomeAppliance 클래스를 상속받는다. 제품의 종류에 따라서 할인율이 달라진다. 예를 들어서 텔레비전은 10% 할인, 냉장고는 5% 할인할 수 있다. 몇 개의 객체를 생성한 후에 getPrice() 함수를 호출하여서 가격을 출력하여 보자.

```
class HomeAppliance {  
    int price;  
    virtual double getPrice() = 0;  
    ...  
};
```

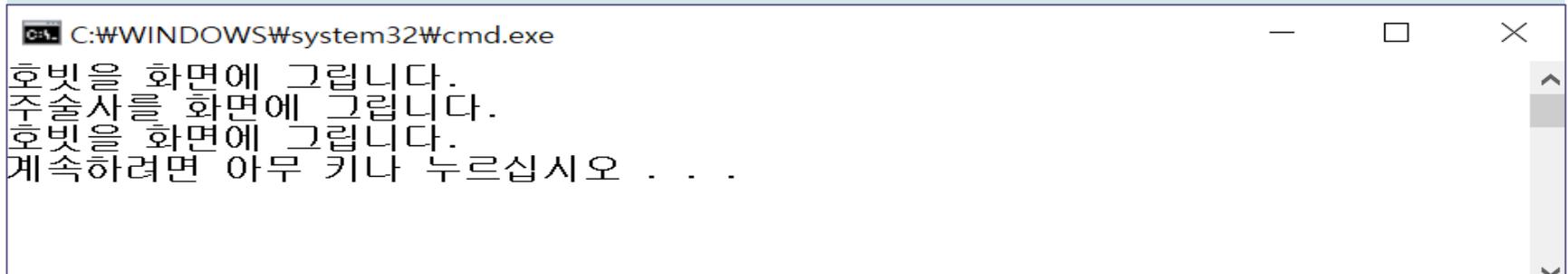


```
C:\WINDOWS\system32\cmd.exe  
가격: 90000  
가격: 190000  
가격: 270000  
계속하려면 아무 키나 누르십시오 . . .
```

실습문제(5)

- 간단한 게임 프로그램을 다형성을 이용하여서 작성하여 보자. 게임에는 많은 캐릭터들이 있다. 예를 들어서 호빗(Hobbit), 주술사(Sorcerer) 등이 있다. 이들 클래스들이 공통의 부모 클래스(GameCharacter)로부터 상속을 받았다고 가정하라. 이 클래스는 draw()라는 가상함수를 가지고 있다. 자식 클래스들은 이 draw()메소드를 재정의 한다. 이들 캐릭터들은 포인터 배열에 주소값이 저장된다. 화면을 다시 그리기 위하여 프로그램은 주기적으로 모든 캐릭터의 draw()를 호출한다. 예를 들어서 호빗의 경우, draw()가 호출되면 콘솔에 “호빗을 그립니다.”라고 출력된다..

```
class GameCharacter {
Public:
    GameCharacter() {}
    virtual void draw() = 0;
    ...
};
```



```
C:\WINDOWS\system32\cmd.exe
호빗을 화면에 그립니다.
주술사를 화면에 그립니다.
호빗을 화면에 그립니다.
계속하려면 아무 키나 누르십시오 . . .
```

실습문제(6)

- Point 클래스를 상속받아서 ThreeDPoint 클래스를 정의해보자. ThreeDPoint 클래스는 3차원 공간 상의 점을 나타내고 int z; 멤버 함수를 추가로 가진다.

```
class Point {
    int x, y;
};
...
int main()
{
    ThreeDPoint p(10, 10, 10);
    p.print();
}
```



```
C:\WINDOWS\system32\cmd.exe
(10, 10, 10)
계속하려면 아무 키나 누르십시오 . . .
```

- 위의 프로그램을 컴파일 할 수 있도록 생성자, 접근자, 설정자 등의 함수를 추가하라.
- + 연산자를 중복 정의하여서 다음과 같은 연산이 가능하도록하라.

```
ThreeDPoint p1(10, 10, 10);
ThreeDPoint p2(10, 10, 10);
ThreeDPoint p3;
p3 = p1 + p2
```

참고문헌

- 뇌를 자극하는 C++ 프로그래밍, 이현창, 한빛미디어, 2011
- 열혈 C++ 프로그래밍(개정판), 윤성우, 오렌지미디어, 2012
- 객체지향 원리로 이해하는 ABSOLUTE C++ , 최영근 외 4명 , 교보문고, 2013
- C++ ESPRESSO, 천인국, 인피니티북스, 2011
- 명품 C++ Programming, 황기태 , 생능출판사, 2018
- 어서와 C++는 처음이지, 천인국, 인피니티북스, 2018



Q&A