

창의적 소프트웨어 설계 실습 문제 08 – hw08-1

제출 기한

11 월 6 일 화 23:59 PM

- I. hw8-1(mkdir hw8-1)라는 폴더를 만들고 GitLab 에 push
- II. hw8-1 디렉토리에 Makefile 과 각 문제에서 요구하는 파일들을 작성
- III. **'make'명령을 수행하여 숙제가 모두 빌드**
- IV. 최종 버전을 GitLab 에 commit
- V. **시간과 파일명**, 입력과 출력 방식 반드시 지키기

문제 1

숫자를 하나 지정하면 입력 받은 숫자, 입력 받은 숫자의 제곱, 입력받은 숫자의 세제곱을 계산하는 Number, Square, Cube 클래스는 아래의 정의를 가지며, 주석에 지시된 대로 나머지 부분을 작성한다.

```
class Number
{
    protected:
        int _num;
    public:
        Number() { _num = 0; };
        Number(int num) { _num = num; };
        void setNumber(int num)
        {
            _num = num;
        }
        int getNumber()
        {
            return _num;
        }
};
```

class Square // Number 를 상속받을 수 있게 코드 추가

```
{
public:
    int getSquare(); // setNumber()로 지정된 숫자의 제곱을 리턴하도록 소스 파일에 구현
};
```

class Cube // Square 를 상속받을 수 있게 코드 추가

```
{
public:
    int getCube (); // setNumber()로 지정된 숫자의 세제곱을 리턴하도록 소스 파일에 구현
};
```

main 함수에서는 사용자의 입력을 반복적으로 받으며, 다음과 같은 입력을 설명된 방식으로 처리한다.

- number 숫자
 - : Number 객체를 하나 만들어 아래 실행 예의 형식으로 getNumber()의 리턴값을 출력
- square 숫자
 - : Square 객체를 하나 만들어 아래 실행 예의 형식으로 getNumber(), getSquare()의 리턴값을 출력
- cube 숫자
 - : Cube 객체를 하나 만들어 아래 실행 예의 형식으로 getNumber(), getSquare(), getCube()의

리턴값을 출력

- quit

- 입력을 종료

이외의 명령어는 들어오지 않음.

파일명 : number (number.h number.cc number_main.cc)

입력 / 출력

```
$ ./number
number 3
getNumber(): 3
square 2
getNumber(): 2
getSquare(): 4
cube 4
getNumber(): 4
getSquare(): 16
getCube(): 64
quit
$
```

문제 2.

Rectangle 클래스를 상속받아 정사각형(Square), 정사각형이 아닌 사각형(NonSquare) 클래스를 생성.

Rectangle, Square, NonSquare 클래스는 아래와 같은 정의를 가지며, 주석에 지시된 대로 나머지 부분을 작성한다.

```
class Rectangle {
public:
    Rectangle(int height, int width); // 필요한 데이터를 멤버변수로 저장하도록 소스 파일에 구현

    int getWidth() const;    // 사각형의 너비를 반환하도록 소스 파일에 구현
    int getHeight() const;   // 사각형의 높이를 반환하도록 소스 파일에 구현
    int getArea() const;     // 사각형의 넓이를 구하도록 소스 파일에 구현
    int getPerimeter() const; // 사각형의 둘레를 구하도록 소스 파일에 구현
private:
    // 필요한 멤버변수를 정의
};

class Square: public Rectangle {
public:
    Square (int width);      // 부모 클래스의 생성자를 적절히 호출하도록 소스 파일에 구현
    void print() const;      // Square 객체의 정보를 출력 (아래 실행 예 참조)
};

class NonSquare: public Rectangle {
public:
    NonSquare (int height, int width); // 부모 클래스의 생성자를 적절히 호출하도록 소스
    파일에 구현
    void print() const; // NonSquare 객체의 정보를 출력 (아래 실행 예 참조)
};
```

main 함수에서는 사용자의 입력을 반복적으로 받으며, 다음과 같은 입력을 설명된 방식으로 처리한다.

- nonsquare 높이 너비
 - NonSquare 객체를 하나 만들어 아래 실행 예의 형식으로 해당 객체의 정보, 넓이, 둘레를 출력한다.
- square 한_변의_길이

- Square 객체를 하나 만들어 아래 실행 예의 형식으로 해당 객체의 정보, 넓이, 둘레를 출력한다.

- quit

- square, nonsquare, quit 이외의 명령어는 들어오지 않음

파일명 : rectangle (rectangle.h rectangle.cc rectangle_main.cc)

입력 / 출력

```
$ ./rectangle
nonsquare 3 5
3x5 NonSquare
Area: 15
Perimeter: 16
square 7
7x7 Square
Area: 49
Perimeter: 28
quit
$
```