

05. 생성자와 소멸자, 분할 컴파일

[실습 5-1]

- 다음 main() 함수가 잘 작동하도록 Rectangle 클래스를 작성하고 프로그램을 완성하라. Rectangle 클래스는 width와 height의 두 멤버 변수와 3 개의 생성자, 그리고 isSquare() 함수를 가진다.

```
int main() {  
    Rectangle rect1;  
    Rectangle rect2(3, 5);  
    Rectangle rect3(3);  
  
    if(rect1.isSquare()) cout << "rect1은 정사각형이다." << endl;  
    if(rect2.isSquare()) cout << "rect2는 정사각형이다." << endl;  
    if(rect3.isSquare()) cout << "rect3는 정사각형이다." << endl;  
}
```

```
rect1은 정사각형이다.  
rect3는 정사각형이다.
```

[실습 5-2]

- Oval 클래스는 주어진 사각형에 내접하는 타원을 추상화한 클래스이다. Oval 클래스의 멤버는 모두 다음과 같다. Oval 클래스를 선언부와 구현부로 나누어 작성하라.

- 정수값의 사각형 너비와 높이를 가지는 width, height 변수 멤버
- 너비와 높이값을 매개변수로 받는 생성자
- 너비와 높이를 1로 초기화하는 매개변수 없는 생성자
- width와 height를 출력하는 소멸자
- 타원의 너비를 리턴하는 getWidth() 함수멤버
- 타원의 높이를 리턴하는 getHeight() 함수멤버
- 타원의 너비와 높이를 변경하는 set(int w, int h) 함수멤버
- 타원의 너비와 높이를 화면에 출력하는 show() 함수멤버

Oval 클래스를 활용하는 코드의 사례와 결과는 다음과 같다.

```
int main() {  
    Oval a, b(3, 4);  
    a.set(10, 20);  
    a.show();  
    cout << b.getWidth() << ", " << b.getHeight() << endl;  
}
```

```
width = 10, height = 20  
3, 4  
Oval 소멸 : width = 3, height = 4  
Oval 소멸 : width = 10, height = 20
```

[실습 5-3]

- 다음 클래스에 중복된 생성자를 디폴트 인자를 가진 하나의 생성자로 작성하고 테스트 프로그램을 작성하라.

```
class MyVector{
    int *mem;
    int size;
public:
    MyVector();
    MyVector(int n, int val);
    ~MyVector() { delete [] mem; }

    MyVector::MyVector() {
        mem = new int [100];
        size = 100;
        for(int i=0; i<size; i++) mem[i] = 0;
    }

    MyVector::MyVector(int n, int val) {
        mem = new int [n];
        size = n;
        for (int i=0; i<size; i++) mem[i] = val;
    }
}
```

[실습 5-4]

- 아래의 소스를 헤더 파일과 cpp 파일로 분리하여 재작성하라.

```
#include <iostream>
using namespace std;

class Adder { // 덧셈 모듈 클래스
    int op1, op2;
public:
    Adder(int a, int b);
    int process();
};

Adder::Adder(int a, int b) {
    op1 = a; op2 = b;
}

int Adder::process() {
    return op1 + op2;
}
```

```
class Calculator { // 계산기 클래스
public:
    void run();
};

void Calculator::run() {
    cout << "두 개의 수를 입력하세요>>";
    int a, b;
    cin >> a >> b; // 정수 두 개 입력
    Adder adder(a, b); // 덧셈기 생성
    cout << adder.process(); // 덧셈 계산
}

int main() {
    Calculator calc; // calc 객체 생성
    calc.run(); // 계산기 시작
}
```

두 개의 수를 입력하세요>>5 -20
-15