

● 과제1

```
#include<iostream> // 입출력 헤더파일
using namespace std;//이름공간 설정

class Triangle {//클래스 생성
private://접근지정자
    int width, height;//멤버변수 세엇ㅇ
public://접근지정자
    Triangle();//생성자 생성
    double TriangleArea(double w, double h); //멤버함수 선언
    int setWidth(int w); //멤버함수 선언
    int setHeight(int h); //멤버함수 선언
    ~Triangle(); //소멸자 생성
}; //클래스 생성 종료

Triangle::Triangle() { // 생성자 초기화
    width = 1; //내부에서 변수 초기화
    height = 1; //내부에서 변수 초기화

    cout << "폭: " << width << ", 높이:" << height << "인 삼각형 생성" << endl; //
초기값 출력
}//생성자 종료

inline double Triangle::TriangleArea(double w,double h) { //인라인 함수 처리
    return width * height * (1.0 / 2.0); // 삼각형 면적 리턴값 출력
}// 인라인 함수 종료
inline int Triangle::setWidth(int w) { //인라인 함수 처리
    width = w; //밑변 초기화
}//인라인 함수 종료
inline int Triangle::setHeight(int h) { //인라인 함수 처리
    height = h; //높이 초기화
}//인라인 함수 종료

Triangle::~Triangle() { //소멸자 생성
    cout << "폭: " << width << ", 높이:" << height << "인 삼각형 소멸" << endl;
//소멸 출력
}

int main() {//메인함수 시작
    Triangle tri; //객체 생성
    tri.setWidth(3); // 객체의 밑변 선언
    tri.setHeight(5); // 객체의 높이 선언
    cout << "삼각형의 면적은 " << tri.TriangleArea(3.0, 5.0) << endl; //객체 면적
    출력
}
```

```
Microsoft Visual Studio 디버그 터미널
쪽 : 1, 높이 :1인 삼각형 생성
삼각형의 면적은 7.5
쪽 : 3, 높이 :5인 삼각형 소멸

C:\Users\leesoochan\OneDrive\바탕 화면\3학년 1학기\c++\코딩\제작\제작\제작.cpp
파일 내용은 다음과 같습니다.
이 창은 단수경험을 위한 것입니다. 그 외에는 그대로 두세요.
```

● 과제2

```
#include<iostream> //입출력 헤더파일
using namespace std; //이름공간 설정

class Tower { //클래스 설정
private: //접근지정자
    int height; //높이 설정
public: //접근지정자
    Tower(); //기본생성자 생성
    Tower( int a); //생성자 생성
    int getHeight(); //높이 멤버함수 선언
};

Tower::Tower(){ //생성자
    height = 1; //기본 높이 설정
}
Tower::Tower( int a) { //생성자 설정
    height = a; // 변수에 따른 높이 설정
}

int Tower::getHeight() { //높이 설정
    return height; //높이 설정
}

int main() { //메인함수 설정
    Tower myTower; //객체선언
    Tower seoulTower(100); // 객체선언
    cout << "높이는 " << myTower.getHeight() << "미터" << endl; //기본 타워 높이
    출력
    cout << "높이는 " << seoulTower.getHeight() << "미터" << endl; //서울타워 높이
    출력
}
```

높이는 1미터
높이는 100미터

C:\Users\leesoochan\OneDrive\바탕 화면\3학년 1학기
이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...|

● 과제3

```
#include<iostream> //입출력 헤더파일
using namespace std; //이름공간 설정

class coffeeMachine { //클래스 설정
private://접근지정자
    int coffee, water, sugar; //변수 선언

public://접근지정자
    coffeeMachine( int a, int b, int c); //생성자 생성
    void americano(); //아메리카노 추출 멤버함수
    void espresso(); //에스프레소 추출 멤버함수
    void sugarcoffee(); //설탕커피 추출 멤버함수
    void show(); //현재 머신 상태 추출 멤버함수
    void fill(); //재료 리필 멤버함수
};

coffeeMachine::coffeeMachine( int a, int b, int c){ //생성자
    coffee = a; //기본 커피 설정
    water = b; //기본 물 설정
    sugar = c; //기본 설탕 설정
}

void coffeeMachine::americano() { //아메리카노 추출시 사용
    coffee -= 1; //커피 사용량
    water -= 2; //물 사용량
    sugar -= 0; //설탕 사용량
}

void coffeeMachine::espresso() { //에스프레소 추출시 사용
    coffee -= 1; //커피 사용량
    water -= 1; //물 사용량
    sugar -= 0; //설탕 사용량
}

void coffeeMachine::sugarcoffee() { //설탕커피 추출시 사용
    coffee -= 1; //커피 사용량
    water -= 2; //물 사용량
    sugar -= 1; //설탕 사용량
}

void coffeeMachine::show() { //현재상태
```

```

        cout << "현재 커피머신 상태, 커피:" << coffee << "물: " << water << "설탕: "
<< sugar << endl;//재료현황 출력
}

void coffeemachine::fill() {//재료 리필
    coffee = 10;//커피 리필
    water = 10;//물 리필
    sugar = 10;//설탕 리필
}

int main() {//메인함수 설정
    coffeemachine java(5,10,3); //객체선언
    java.espresso();// 에스프레소 추출
    java.show();//현 상황 출력
    java.americano();// 아메리카노 추출
    java.show();//현 상황 출력
    java.sugarcoffee();// 설탕커피 추출
    java.show();//현 상황 출력
    java.fill();// 재료 리필
    java.show();//현 상황 출력
}

```

The screenshot shows the Microsoft Visual Studio Debug window with the title bar 'Microsoft Visual Studio 디버그'. The console output displays the state of the coffee machine after each command:

```

현재 커피머신 상태, 커피:4물: 9설탕: 3
현재 커피머신 상태, 커피:3물: 7설탕: 3
현재 커피머신 상태, 커피:2물: 5설탕: 2
현재 커피머신 상태, 커피:10물: 10설탕: 10

```

Below the output, the status bar shows the path: C:\Users\leesoochan\OneDrive\바탕 화면\3학년 1학기\c++\ and the message: 이(가) 종료되었습니다(코드: 0개). 이 창을 닫으려면 아무 키나 누르세요...|

● 과제4

```

#include<iostream>//입출력 헤더파일
using namespace std;//이름공간 설정

class Oval {//클래스 설정
private://접근지정자
    int width, height;//변수 선언

public://접근지정자
    Oval(int w, int h);//생성자 생성
    Oval();//기본 생성자
    ~Oval();//소멸자
    int getWidth(); //너비 받을 멤버함수
    int getHeight(); //높이받을 멤버함수
    void set(int w, int h); //세팅 멤버함수
}

```

```

        void show()//멤버함수
    };

Oval::Oval() { // 생성자 시작
    width = 1; //기본 너비
    height = 1; //기본 높이
}
Oval::Oval(int w, int h) { //생성자
    width = w;//너비 설정
    height = h;// 높이 설정
}

int Oval::getWidth() { // 너비 받을 멤버함수
    return width; //너비 반환
}

int Oval::getHeight() { //높이 받을 멤버함수
    return height;// 높이 반환
}

void Oval::set(int w, int h) { // 타원 설정 멤버함수
    width = w;// 너비 설정
    height = h;//높이 설정
}

void Oval::show() { //타원 너비와 높이 멤버함수
    cout << "타원의 너비: " << width << "타원의 높이: " << height << endl;// 타원
    너비 높이 출력
}

Oval::~Oval() { //소멸자 출력
    cout << "Oval 소멸: width = " << width << ", height = " << height << endl;
//소멸자 출력
}

int main() { //메인함수 시작
    Oval a, b(3, 4); //객체 선언
    a.set(10, 20); //타원 너비 높이 설정
    a.show(); //멤버함수
    cout << b.getWidth() << ", " << b.getHeight() << endl; //각 객체 너비 높이
    출력
}

```

Microsoft Visual Studio 디버그

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

C:\Users\leesoochan\OneDrive\바탕 화면\3학년 1학기\
이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...|
```

- 과제5

```
#include<iostream> //입출력 헤더파일
using namespace std; //이름공간 설정

class Ram{//클래스 설정
private://접근지정자
    char mem[100 * 1024]; //변수 선언
    int size; //사이즈 변수 선언

public://접근지정자
    Ram(); //생성자 생성
    ~Ram(); //소멸자 생성
    char read(int adress); //읽기변수 선언
    void write(int address, char value); //쓰기 변수 선언
};

Ram::Ram() { //생성자 시작
    mem[100 * 1024] = 0; //메모리 배열 초기화
    size = 100 * 1024; //사이즈 초기화
}

Ram::~Ram() { //소멸자 시작
    cout << "메모리 제거됨" << endl; //메모리 제거 출력
}

void Ram::write(int adress, char value) { //쓰기 멤버함수
    mem[adress] = value; //메모리 설정
}

char Ram::read(int adress) { //읽기 설정
    return mem[adress]; //배열주소에 맞는 메모리 반환
}

int main(){ //메인함수 시작
    Ram ram; //객체 선언
    ram.write(100, 20); //메모리 100번지 값 설정
    ram.write(101, 30); //메모리 101번지 값 설정
}
```

```
char res = ram.read(100) + ram.read(101); //각 주소값의 합 받기  
ram.write(102, res); //두 주소값 합 작성  
cout << "102번지의 값 = " << (int)ram.read(102) << endl; //두 주소 값의 합  
출력  
}
```

102번지의 값 = 50
메모리 제거됨

```
C:\Users\leesoochan\OneDrive\바탕 화면\3학년 1학기\c-  
(가) 종료되었습니다(코드: 0개).  
이 차트를 다운려면 아무 키나 누르세요 |
```