REPORT



|  |  |
| --- | --- |
| 과 목 : | 객체지향프로그래밍 |
| 제출일자 : | 2023.03.28 |
| 담당교수 : | 황성호 |
| 학 과 : | 멀티디자인학과 |
| 학 번 : | 201522405 |
| 이 름 : | 최준하 |

1.(예제 3-7)

#include <iostream>

using namespace std;

class Circle {

public:

int radius;

Circle();

Circle(int r);

~Circle();

double get\_area();

};

Circle::Circle() {

radius = 1;

cout << "반지름" << radius << "원 생성" << endl;

}

Circle::Circle(int r) {

radius = r;

cout << "반지름" << radius << "원 생성" << endl;

}

Circle::~Circle() {

cout << "반지름" << radius << "원 소멸" << endl;

}

double Circle::get\_area() {

return 3.14 \* radius \* radius;

}

int main() {

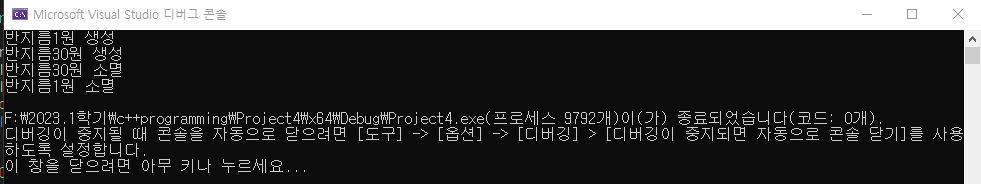
Circle donut;

Circle pizza(30);

return 0;

}

🡪출력 결과



2. (예제 3-8)

#include <iostream>

using namespace std;

class Circle {

public:

int radius;

Circle();

Circle(int r);

~Circle();

double get\_area();

};

Circle::Circle() {

radius = 1;

cout << "반지름 " << radius << " 원 생성" << endl;

}

Circle::Circle(int r) {

radius = r;

cout << "반지름 " << radius << " 원 생성" << endl;

}

Circle::~Circle() {

cout << "반지름 " << radius << " 원 소멸" << endl;

}

double Circle::get\_area() {

return 3.14 \* radius \* radius;

}

Circle global\_donut(1000);

Circle global\_pizza(2000);

void f() {

Circle f\_conut(100);

Circle f\_pizza(200);

}

int main() {

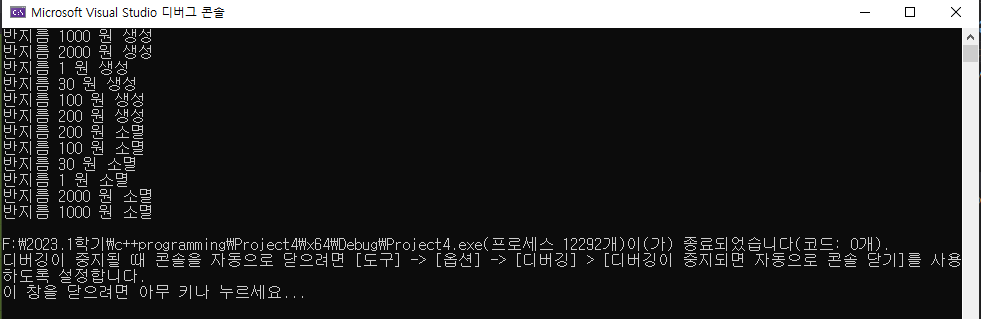
Circle main\_donut;

Circle main\_pizza(30);

f();

}

🡪출력 결과



3. (예제 3-9)

#include <iostream>

using namespace std;

class private\_access\_error {

private:

int a;

void f();

private\_access\_error();

public:

int b;

private\_access\_error(int x);

void g();

};

private\_access\_error::private\_access\_error() {

a = 1;

b = 1;

}

private\_access\_error::private\_access\_error(int x) {

a = x;

b = x;

}

void private\_access\_error::f() {

a = 5;

b = 5;

}

void private\_access\_error::g() {

a = 6;

b = 6;

}

int main() {

//private\_access\_error obj\_a; //error

private\_access\_error obj\_b(100);

//obj\_b.a = 10; //error

obj\_b.b = 20;

//obj\_b.f(); //error

obj\_b.g();

}

🡪출력 결과

오류

4. (예제 3-10)

#include <iostream>

using namespace std;

struct struct\_circle

{

private:

int radius;

public:

struct\_circle(int r) {

radius = r;

} // 구조체 자동 인라인 생성

double get\_area();

};

double struct\_circle::get\_area() {

return 3.14 \* radius \* radius;

}

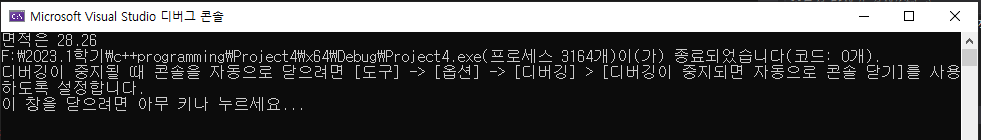
int main() {

struct\_circle waffle(3);

cout << "면적은 " << waffle.get\_area();

}

🡪출력 결과



5. (예제 3-11)

헤더1\_덧셈 클라스:

#ifndef ADDER\_H

#define ADDER\_H

class adder {

int op1, op2;

public:

adder(int a, int b);

int process();

};

#endif // !ADDER\_H

헤더2\_계산기 클라스:

#ifndef CALCULATOR\_H

#define CALCULATOR\_H

class calculater {

public:

void run();

};

#endif // !CALCULATOR\_H

cpp파일1\_덧셈:

#include "add\_module.h"

adder::adder(int a, int b) {

op1 = a; op2 = b;

}

int adder::process() {

return op1 + op2;

}

cpp파일2\_계산

#include <iostream>

#include "add\_module.h"

#include "calculater.h"

using namespace std;

void calculater::run() {

cout << "두 개의 수를 입력>> ";

int a, b;

cin >> a >> b;

adder \_adder(a, b);

cout << \_adder.process();

}

Main 함수:

#include "calculater.h"

int main() {

calculater calc;

calc.run();

}

🡪출력 결과

