**class**

**class를 이번에 처음 배우게 되었는데 이해하기 어렵기도 하고 중요한 개념이라 기초를 다질 필요가 있다고 생각합니다.**

#include <iostream>

using namespace std;

class Count{

int num;

public:

void Reset(){num=0;}

void Increment(){num++;}

void Print(){cout<<num<<" ";}

};

int main()

{

Count n1 ,n2;

n1.Reset();

n1.Increment();n1.Increment();

n2.Reset();n2.Increment();

n1.Print();n2.Print();

return 0;

}

**constructor**

**constructor의 개념을 이해할 때 필요한 코드라고 생각되어 중요한 코드라고 생각합니다.**

#include <iostream>

using namespace std;

class Count{

int num;

public:

Count(){num=0;}

Count(int n){num=(n<0)?0:n;}

void Increment(){num++;}

void Print(){cout<<num<<" ";}

};

int main()

{

Count n1 ,n2(10);

n1.Increment();n1.Increment();

n2.Increment();

n1.Print();n2.Print();

return 0;

}

**destructor, static member**

**destructor의 개념을 이해하기 위해 필요한 코드라 중요하다고 생각합니다.**

#include <iostream>

using namespace std;

class Count{

int num;

public:

static int total;

static void PrintTotal(){

cout<<"total ="<<total<<endl;

}

Count();

~Count();

void Increment() {num++;total++;}

void Print(){cout<<"num"<<num<<",";PrintTotal();}

};

int Count::total = 0;

int main()

{

Count n1;

n1.Increment();n1.Increment();

{

Count n2;

n2.Increment();

}

Count::PrintTotal();

return 0;

}

Count::Count(){

num=0;Print();

}

Count::~Count(){

Print();

total-=num;

}

**Function overriding**

#include <iostream>

using namespace std;

class Student{

public:

void PrintHello(){cout<<"Hello, I am a ";}

void PrintStudent(){cout<<"student\n";}

};

class Freshman:public Student{

public:

void PrintStudent(){cout<<"freshman\n";}

};

int main()

{

Student a;

Freshman b;

a.PrintHello(); a.PrintStudent();

b.PrintHello(); b.PrintStudent();

return 0;

}

**virtual function**

**virtual의 사용법을 알 수 있는 코드라 중요하다고 생각합니다.**

#include <iostream>

using namespace std;

class Student{

public:

virtual void PrintHello(){cout<<"Hello, I am a ";}

virtual void PrintStudent(){cout<<"student\n";}

void Print() {PrintHello();PrintStudent();}

};

class Freshman:public Student{

public:

void PrintStudent(){cout<<"freshman\n";}

};

class Sophomore:public Student{

public:

void PrintHello(){cout<<"Hi, I am a ";}

void PrintStudent(){cout<<"freshman\n";}

};

int main()

{

Freshman a;

Sophomore b;

a.Print();b.Print();

return 0;

}