

Week 6-1 : Constructor, Destructor

Copy Constructor

Part1. Constructor

- 생성자의 외형

클래스의 이름과 함수의 이름이 동일.

반환형이 선언되어 있지 않으며, 실제로 반환하지 않음.

함수의 일종으로 오버로딩과 매개변수에 디폴트 값 설정이 가능.

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

class Constructor
{
int num1;
int num2;

public:
Constructor()
{
num1=0;
num2=0;
}
Constructor(int n)
{
num1=n;
num2=0;
}
Constructor(int n1, int n2)
{
num1=n1;
num2=n2;
}
}
```

```
/* 디폴트 매개변수 생성자
Constructor(int n1=0, int n2=0)
{
    num1=n1;
    num2=n2;
}
*/

void ShowData() const
{
    cout<<num1<<' '<<num2<<endl;
}
};

int main(void)
{
    Constructor sc1;
    sc1.ShowData();

    Constructor sc2(100);
    sc2.ShowData();

    Constructor sc3(100, 200);
    sc3.ShowData();
    return 0;
}
```

sc1, sc2, sc3 객체를 생성하면서 생성자를 거치는데 각각 오버로드 된 생성자를 거침을 확인할 수 있다. 디폴트 매개변수의 생성자를 위해서는 나머지 생성자들을 다 지워야 하며, 실행결과는 동일하다.

- 멤버 이니셜라이저를 통한 초기화

멤버변수로 선언된 객체의 생성자 호출에 활용됨.

몸체부분에서 초기화를 시키는 것이 아니라, 매개변수 옆에서 초기화를 하는 형태

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

```
class Constructor
{
int num1;
int num2;

public:

Constructor(int n1, int n2) : num1(n1), num2(n2)
{

}

void ShowData() const
{
cout<<num1<<' '<<num2<<endl;
}
};

int main(void)
{
Constructor sc(100,200);
sc.ShowData();

return 0;
}
```

Part2. Destructor

생성자에서 할당한 리소스의 소멸에 사용

new 연산자를 이용해서 할당해 놓은 메모리 공간이 있다면, 소멸자에서 delete 연산을 이용해서 이 메모리를 소멸

참고>> new와 delete

c언어의 malloc과 free에 대응되는 것으로 c++에서는 new와 delete를 사용한다. 객체를 생성할 때는 반드시 new를 사용하여야 한다.

```
#include <iostream>
```

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

class Book
{
private:
char * bookName;
int bookNum;
public:
Book(char * tempName, int tempNum)
{
int len=strlen(tempName)+1;
bookName=new char[len];
strcpy(bookName, tempName);
bookNum=tempNum;
}

void ShowBookInfo() const
{
cout<<"도서명 : "<<bookName<<endl;
cout<<"도서번호 : "<<bookNum<<endl;
}

~Book()
{
delete []bookName;
cout<<"destructor"<<endl;
}
};

int main(void)
{
Book book1("Computer Programming", 2001001);
Book book2("This is C++", 400010);
book1.ShowBookInfo();
book2.ShowBookInfo();
return 0;
}
```

[실습하기]

1. In the first example, Think about generating sc1 object using 'Simpleclass sc1();' instead of 'Simpleclass sc1;'.
2. Make a member manager program.
 - Member class includes private name, age, department variables.
 - Initialize objects using constructor, also make a destructor.
 - Make a public set() method ; use this method, when you input member's information.
 - In the main, make a person type array object which has 3 indexes.
 - Input the member's information using for loop.