



20191114 실습 8주차

박건호

KITRI BoB 8th

5422 RTDCS

devgunho.github.io



static

static 키워드는 C++ 언어에서 가장 혼동되는 키워드 중 하나다.
어디에서 사용되는지에 따라 다른 의미를 지니기 때문이다.

전역 변수에서 static 키워드가 전역 변수에서 사용될 때,
'내부 연결 속성'을 적용한다. 즉, 변수가 정의된 소스 파일 내에서 사용 가능함을 의미한다.

static 키워드는 블록 내에서 선언된 지역 변수에서도 사용할 수 있다.

지역 변수는 '자동 주기(auto duration)'를 가지며, 정의되는 시점에서 생성되고 초기화되며,
정의된 블록이 끝나는 지점에서 소멸한다.

static 키워드를 사용한 지역 변수는 완전히 다른 의미가 있다.

static 키워드를 사용한 지역 변수는 '자동 주기(auto duration)'에서 '정적 주기(static duration)'로 바뀐다.
이것을 정적 변수(static variable)라고도 부르는데,

생성된 Scope(=범위)가 종료한 이후에도 해당 값을 유지하는 변수다.

또한, 정적 변수는 한 번만 초기화되며 프로그램 수명 내내 지속된다.

1. static

아래 코드를 오른쪽과 같은 출력이 나오도록 수정

```
#include <iostream>

void incrementAndPrint()
{
    int value = 1; // 기본적으로 '자동 생명 주기'
    ++value;
    std::cout << value << '\n';
} // value는 여기서 소멸된다.

int main()
{
    incrementAndPrint();
    incrementAndPrint();
    incrementAndPrint();

    return 0;
}
```

Microsoft Visual Studio Debug Console

```
2
2
2
D:\Workspace#19_ComputerProgramming
h code 0.
Press any key to close this window .
```

```
int main()
{
    incrementAndPrint();
    incrementAndPrint();
    incrementAndPrint();

    return 0;
}
```

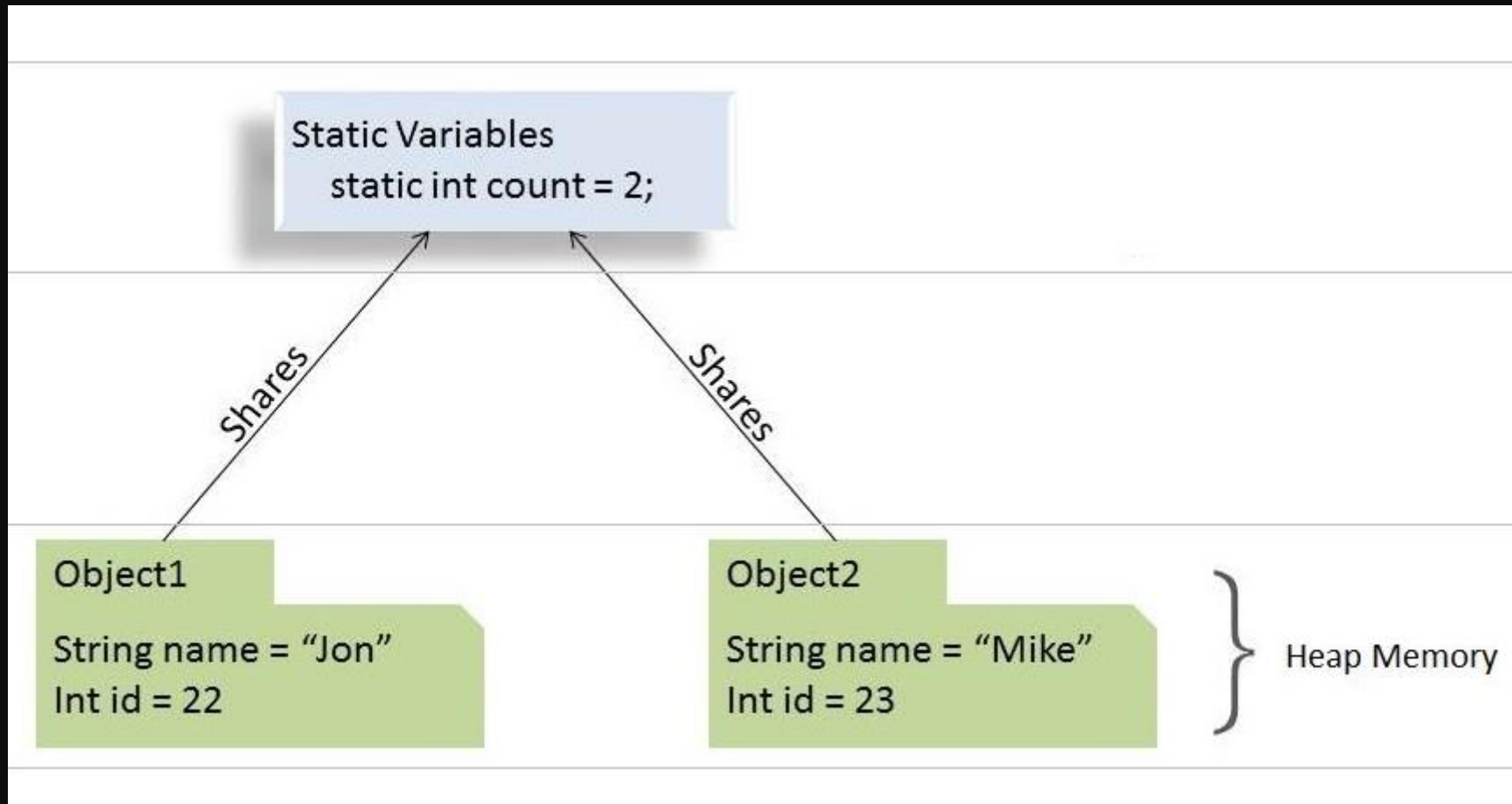
Microsoft Visual Studio

```
2
3
4
D:\Workspace#1
h code 0.
Press any key t
```





Static Members of a C++ Class





Static Members of a C++ Class

```
#include <iostream>

using namespace std;

class Led {
public:
    //Led();
    void turnOn();
    void turnOff();
    static int iCount; // 정적 멤버 변수
    //static int iCount = 0; // 오류 발생
private:
    bool light;
};

int Led::iCount = 0;
// 클래스 외부에서 전역 변수처럼 초기화시켜야 한다.
```

```
int main()
{
    Led led1, led2;

    // 정적 변수를 접근하는 방법은 두 가지가 있는데, 먼저 클래스 이름으로 접근하는 것이다.
    // 이때 범위 지정 연산자인::을 이용한다.

    cout << led1.iCount << endl;
    led1.iCount = 10; // 또는 인스턴스를 이용해서도 접근할 수 있다.
    cout << led1.iCount << endl;
    led2.iCount = 30; // 다른 객체에서 접근하여 값을 바꾸고 결과를 확인해본다.
    cout << led1.iCount << endl;

    /*
    비록 인스턴스 멤버 변수를 접근하는 문법과 똑같지만 서로 다른 인스턴스를 통해서 정적 변수를 참조했다고 할지라도
    결국 같은 하나의 정적 변수 iCount를 접근하는 것이다.
    */

    return 0;
}
```



2. LED Count

Led가 생성될 때마다,
생성된 Led를 count 할 수 있도록 생성자를 수정하고,
main함수가 종료되기 전에
소멸자가 호출되면서
count를 감소 시킬 수 있도록 구현하기

```
int main()
{
    Led led1(true);
    Led led2(true);
    Led led3(true);
    Led led4(false);
    Led led5(false);

    return 0;
}
```

출력

```
LED_Count : 1
LED_Count : 2
LED_Count : 3
LED_Count : 4
LED_Count : 5
LED_Count : 4
LED_Count : 3
LED_Count : 2
LED_Count : 1
LED_Count : 0
```



3. Constructor

```
1  #include <iostream>
2
3  using namespace std;
4
5  class AAA
6  {
7  public:
8      AAA() {
9          cout << "AAA() 호출" << endl;
10     }
11     AAA(int i) {
12         cout << "AAA(int i) 호출" << endl;
13     }
14     AAA(const AAA& a) {
15         cout << "AAA(const AAA& a) 호출" << endl;
16     }
17 };
18
19 int main()
20 {
21
22     return 0;
23 }
```

 Microsoft Visual Studio De

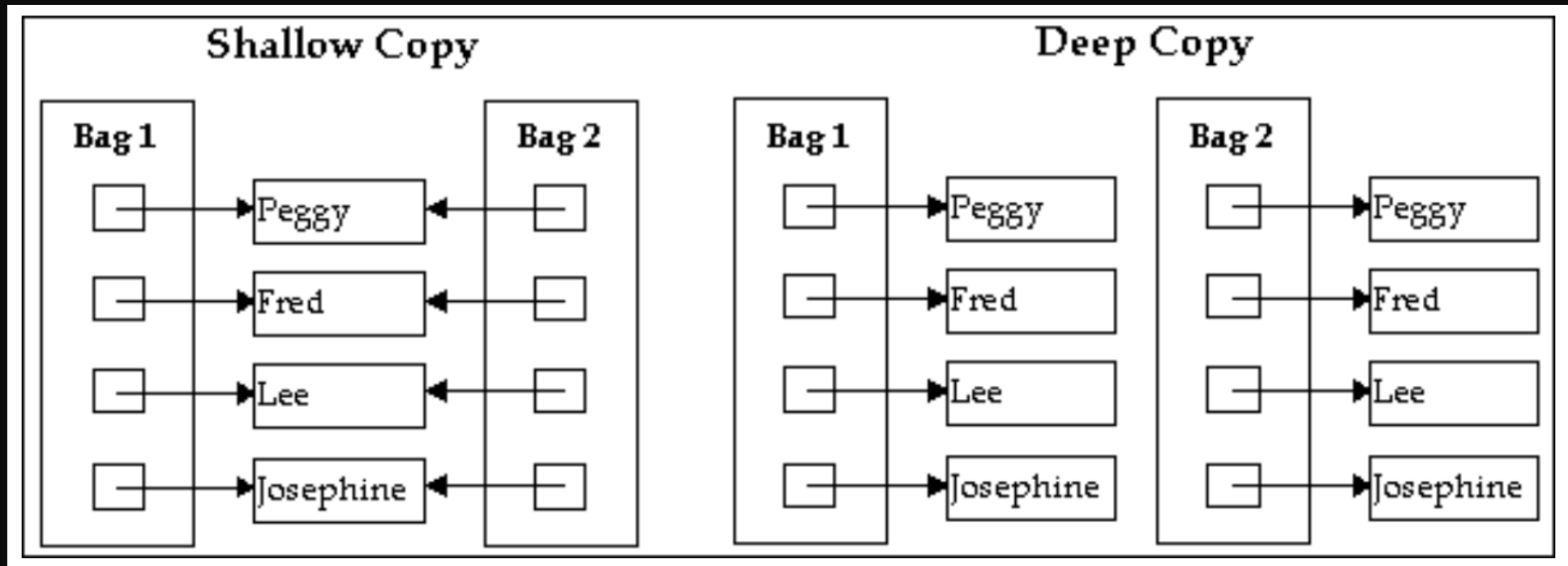
```
AAA() 호출
AAA(int i) 호출
AAA(const AAA& a) 호출
```

이미 구현되어 있는 AAA class의
각각의 생성자가 호출되도록 main을 작성하기



4. Deep Copy Constructor

깊은 복사 생성자를 만들 시에는, 포인터 변수와 동적 할당에 관한 부분을 고려해서 만들어야 한다.
복사 생성자를 만들 시, 반드시 매개변수는 레퍼런스로 전달해야 한다.





4. Deep Copy Constructor

깊은 복사 생성자를 만들 시에는, 포인터 변수와 동적 할당에 관한 부분을 고려해서 만들어야 한다.
복사 생성자를 만들 시, 반드시 매개변수는 레퍼런스로 전달해야 한다.

```
void ShowPersonInfo() const
{
    cout << "이름: " << name << endl;
    cout << "나이: " << age << endl;
}
~Person()
{
    delete[] name;
    cout << "소멸자 호출" << endl;
}
};
```

```
int main()
{
```

```
    Person man1("hong", 20);
    Person man2 = man1; // 복사 생성자 호출 (깊은 복사 실행), Person man2(man1)과 같은 표현
    man1.ShowPersonInfo();
    man2.ShowPersonInfo();

    return 0;
}
```

Microsoft Visual Studio Debug Console

```
이름: hong
나이: 20
이름: hong
나이: 20
소멸자 호출
소멸자 호출
```

```
C:\Users\gunho\Desktop\19_CP_Fall_Lecture_Code\WD
Press any key to close this window . . .
```



5. Find Min Dist. In Square

현재 위치 (x, y) 에서

직사각형의 왼쪽 아래 꼭짓점은 $(0, 0)$ 에 있고, 오른쪽 위 꼭짓점은 (w, h) 에 있다.

직사각형의 경계선까지 가는 거리의 최솟값을 구하시오.

첫째 줄에 $x\ y\ w\ h$ 가 주어진다.

w 와 h 는 1,000보다 작거나 같은 자연수이고, x 는 1보다 크거나 같고,

$w-1$ 보다 작거나 같은 자연수이고, y 는 1보다 크거나 같고, $h-1$ 보다 작거나 같은 자연수이다.

```
Microsoft Vi
6 2 10 3
1
```

```
Microsoft V
7 8 20 21
7
```

```
Microsoft Visual
49 49 100 100
49
```

```
Microsoft Visual S
61 61 100 100
39
```