

유도 클래스의 생성 및 소멸

1108 서민준

유도 클래스의 생성과정 (1)

```
class SoBase
{
private:
    int baseNum;

public:
    SoBase() : baseNum(20)
    {
        std::cout << "SoBase()" << std::endl;
    }

    SoBase(int n) : baseNum(n)
    {
        std::cout << "SoBase(int n)" << std::endl;
    }

    void ShowBaseData() const
    {
        std::cout << baseNum << std::endl;
    }
};
```

기초 클래스

```
class SoDerived : public SoBase
{
private:
    int derivNum;

public:
    SoDerived() : derivNum(30)
    {
        std::cout << "SoDerived()" << std::endl;
    }

    SoDerived(int n) : derivNum(n)
    {
        std::cout << "SoDerived(int n)" << std::endl;
    }

    SoDerived(int n1, int n2) : SoBase(n1), derivNum(n2)
    {
        std::cout << "SoDerived(int n1, int n2)" << std::endl;
    }

    void ShowDerivData() const
    {
        ShowBaseData();
        std::cout << derivNum << std::endl;
    }
};
```

유도 클래스

유도 클래스의 생성과정 (2)

```
int main()
{
    std::cout << "case 1....." << std::endl;
    SoDerived dr1;
    dr1.ShowDer ivData();
    std::cout << "-----" << std::endl;

    std::cout << "case 2....." << std::endl;
    SoDerived dr2(12);
    dr2.ShowDer ivData();
    std::cout << "-----" << std::endl;

    std::cout << "case 3....." << std::endl;
    SoDerived dr3(23, 34);
    dr3.ShowDer ivData();
    std::cout << "-----" << std::endl;

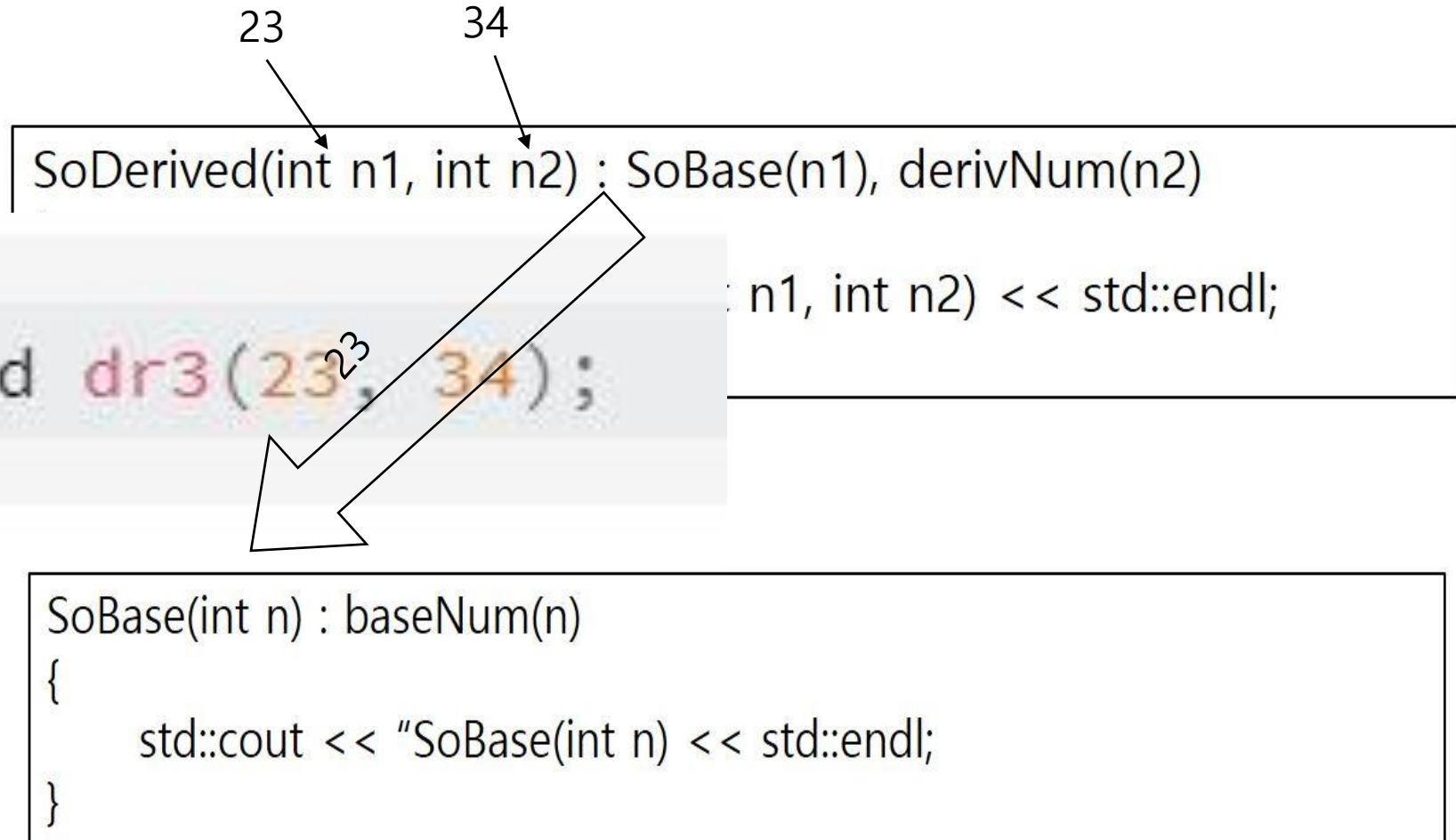
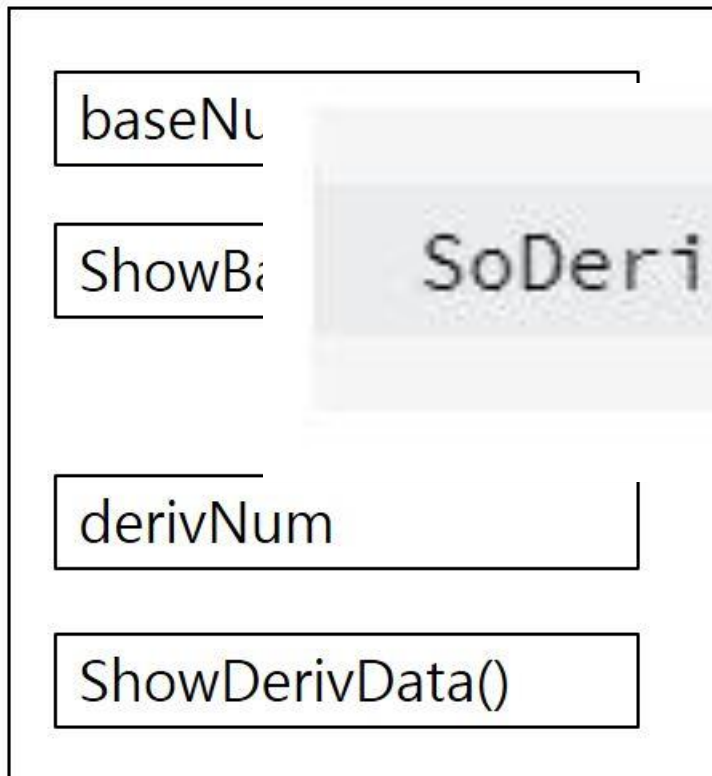
    return 1;
}
```

실행결과

```
Microsoft Visual Studio 디버그 콘솔
case 1.....
SoBase()
SoDerived()
20
30
-----
case 2.....
SoBase()
SoDerived(int n)
20
12
-----
case 3.....
SoBase(int n)
SoDerived(int n1, int n2)
23
34
-----
```

유도 클래스의 생성과정 (3)

SoDerived 객체



유도 클래스의 소멸 과정 (1)

```
class SoBase
{
private:
    int baseNum;

public:
    SoBase(int n) : baseNum(n)
    {
        std::cout << "SoBase() : " << baseNum << std::endl;
    }

    ~SoBase()
    {
        std::cout << "~SoBase() : " << baseNum << std::endl;
    }
};
```

기초 클래스

```
class SoDerived : public SoBase
{
private:
    int derivNum;

public:
    SoDerived(int n) : SoBase(n), derivNum(n)
    {
        std::cout << "SoDerived() : " << derivNum << std::endl;
    }

    ~SoDerived()
    {
        std::cout << "~SoDerived() : " << derivNum << std::endl;
    }
};
```

유도 클래스

유도 클래스의 소멸 과정 (2)

실행 결과

```
int main()
{
    SoDerived drv1(15);
    SoDerived drv2(27);

    return 1;
}
```

Microsoft Visual Studio 디버그 콘솔

```
SoBase() : 15
SoDerived() : 15
SoBase() : 27
SoDerived() : 27
~SoDerived() : 27
~SoBase() : 27
~SoDerived() : 15
~SoBase() : 15
```

3가지 형태의 상속

1108 서민준

protected 선언

private < protected < public

```
class Base
{
private:
    int num1;

protected:
    int num2;

public:
    int num3;

    void ShowData() const
    {
        std::cout << num1 << ", " << num2 << ", " << num3 << std::endl;
    }
};
```

기초 클래스

```
class Dervied : public Base
{
public:
    void ShowBaseMember() const
    {
        std::cout << num1 << ", "; // 컴파일 에러
        std::cout << num2 << ", "; // 컴파일 OK!
        std::cout << num3 << std::endl; // 컴파일 OK!
    }
};
```

유도 클래스

상속 형태 1 – protected 상속

```
class Base
{
private:
    int main()
    {
        Derived drv;
        std::cout << drv.num3 << std::endl; // 컴파일 에러 발생
    }
public:
    int
};
```

기초 클래스

```
class Dervied : protected Base
```

```
class Dervied : protected Base
{
    접근불가:
        int num1;

protected:
        int num2;

protected:
        int num3;
};
```

상속 형태 2 - private 상속

```
class Base
{
private:
    int num1;

protected:
    int num2;

public:
    int num3;
};
```

기초 클래스

상속

```
class Dervied : private Base
{
    // empty!
};
```

유도 클래스

메모리
구성

```
class Dervied : protected Base
{
    접근불가:
        int num1;

private:
    int num2;

private:
    int num3;
};
```

상속 형태 3 – public 상속

private을 제외한 나머지 멤버들을 그대로 상속해라!

C++의 상속은 다중상속과 같이 특별한 경우가 아니면 잘 사용하지 않는다.

상속을 위한 조건

1108 서민준

조건 1. IS-A 관계 (1)

서민준 -> 사람
서민준 -> 고등학생



서민준 is a 사람
서민준 is a 고등학생

상속관계에서는 기초 클래스와 유도 클래스 사이에서 **IS-A 관계**가 성립하여야 한다.

조건 1. IS-A 관계 (2)