

# Visitor: C++ Code Examples

22<sup>nd</sup> LECTURE

엄현상(Eom, Hyeonsang)  
School of Computer Science and Engineering  
Seoul National University

# Outline

- **Visitor: C++ Code Examples**

# Code Examples

- Visitor

```
#include <iostream>
```

```
using namespace std;
```

```
class Item;
```

```
class ItemA;
```

```
class ItemB;
```

```
class ItemVisitor;
```

```
class Pricing1Visitor;
```

```
class Pricing2Visitor;
```

텍스트

“Design Patterns: Elements of Reusable  
Object-Oriented Software,” Erich Gamma,  
Richard Helm, Ralph Johnson, John  
Vlissides, Addison Wesley, 1995

# Code Examples Cont'd

- Visitor Cont'd

```
class Item {  
public:  
    virtual ~Item() {};  
  
    void SetPrice(double p) { _price = p; };  
    double GetPrice() { return _price; };  
  
    virtual double NetPrice() { return _price; };  
    virtual double DiscountPrice() { return 0.6*_price; };  
  
    virtual void Accept(ItemVisitor&) = 0;  
protected:  
    Item() {};  
private:  
    double _price;  
};
```

# Code Examples Cont'd

- Visitor Cont'd

```
class ItemA : public Item {  
public:  
    ItemA() {};  
    virtual void Accept(ItemVisitor& v) { v.VisitItemA(this); }  
};
```

레퍼런스 타입은 포인터라서 전체를 카피하지 않고, 파라미터 등에 있어 전달시에 효율적으로 쓸 수 있음. 여기서 주의 레퍼런스? 포인터?

```
class ItemB : public Item {  
public:  
    ItemB() {};  
    virtual void Accept(ItemVisitor& v) { v.VisitItemB(this); }  
};
```

실제로 쓰일때는 v.으로 변수처럼 쓰이지만, **reference type**이다.

# Code Examples Cont'd

- Visitor Cont'd

```
class ItemVisitor {  
public:  
    virtual void VisitItemA(Item*) {}; 이 친구는 포인터다.  
    virtual void VisitItemB(Item*) {};  
protected:  
    ItemVisitor() {};  
};
```

# Code Examples Cont'd

- Visitor Cont'd

```
class Pricing1Visitor : public ItemVisitor {
public:
    Pricing1Visitor() { _total = 0.0; };
    double GetTotalPrice() { return _total; };
    virtual void VisitItemA(ItemA* pa) { _total += pa->NetPrice(); };
    virtual void VisitItemB(ItemB* pb) { _total += pb-
>DiscountPrice(); };
private:
    double _total;
};
```

텍스트

# Code Examples Cont'd

- Visitor Cont'd

```
class Pricing2Visitor : public ItemVisitor {
public:
    Pricing2Visitor() { _total = 0.0; };

    double GetTotalPrice() { return _total; };

    virtual void VisitItemA(ItemA* pa) { _total += pa-
>DiscountPrice(); };
    virtual void VisitItemB(ItemB* pb) { _total += pb->NetPrice(); };

private:
    double _total;
};
```



# Code Examples Cont'd

- Visitor Cont'd

```
int main () {  
    ItemA a;  
    a.SetPrice(10);  
    ItemB b;  
    b.SetPrice(20);  
  
    Pricing1Visitor p1;  
    a.Accept(&p1); // p1.VisitItemA(&a);  
    b.Accept(&p1); // p1.VisitItemB(&b);  
    cout << "Pricing: p1.GetTotalPrice() = for a & b"  
<< p1.GetTotalPrice() << endl;  
  
    Pricing2Visitor p2;  
    a.Accept(&p2); // p2.VisitItemA(&a);  
    b.Accept(&p2); // p2.VisitItemB(&b);  
    cout << "Pricing: p2.GetTotalPrice() = for b & b"  
<< p2.GetTotalPrice() << endl;  
}
```

잘못된 부분을 찾으시오 -> n %를 지워야 한다?