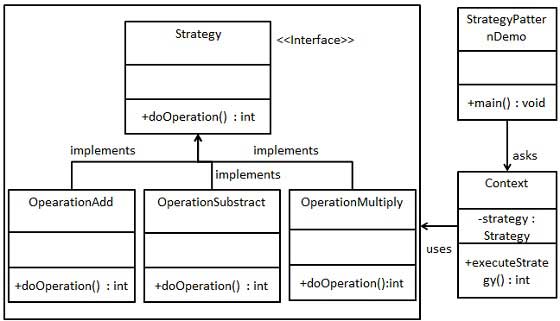
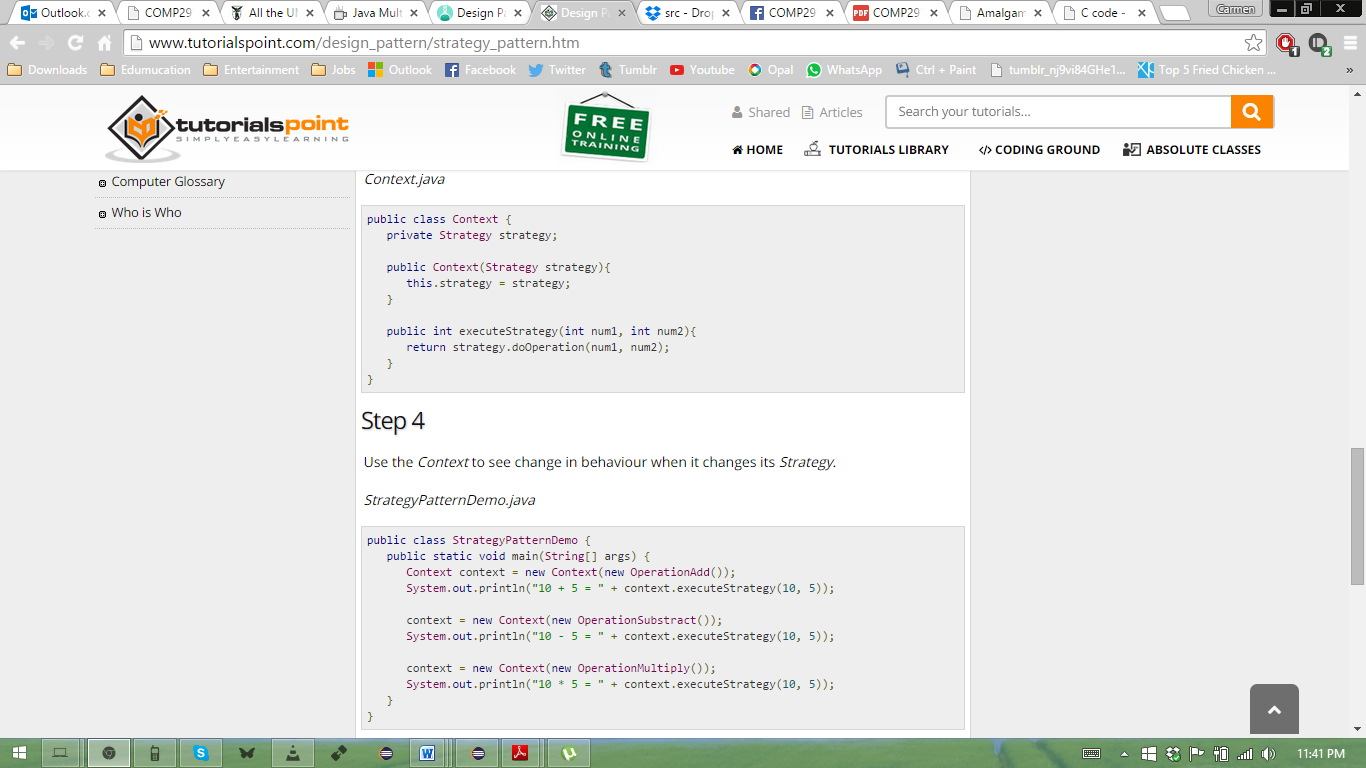
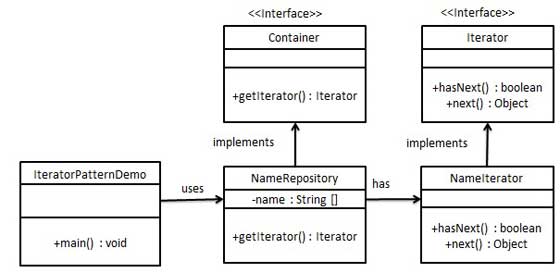
**Strategy Pattern:**

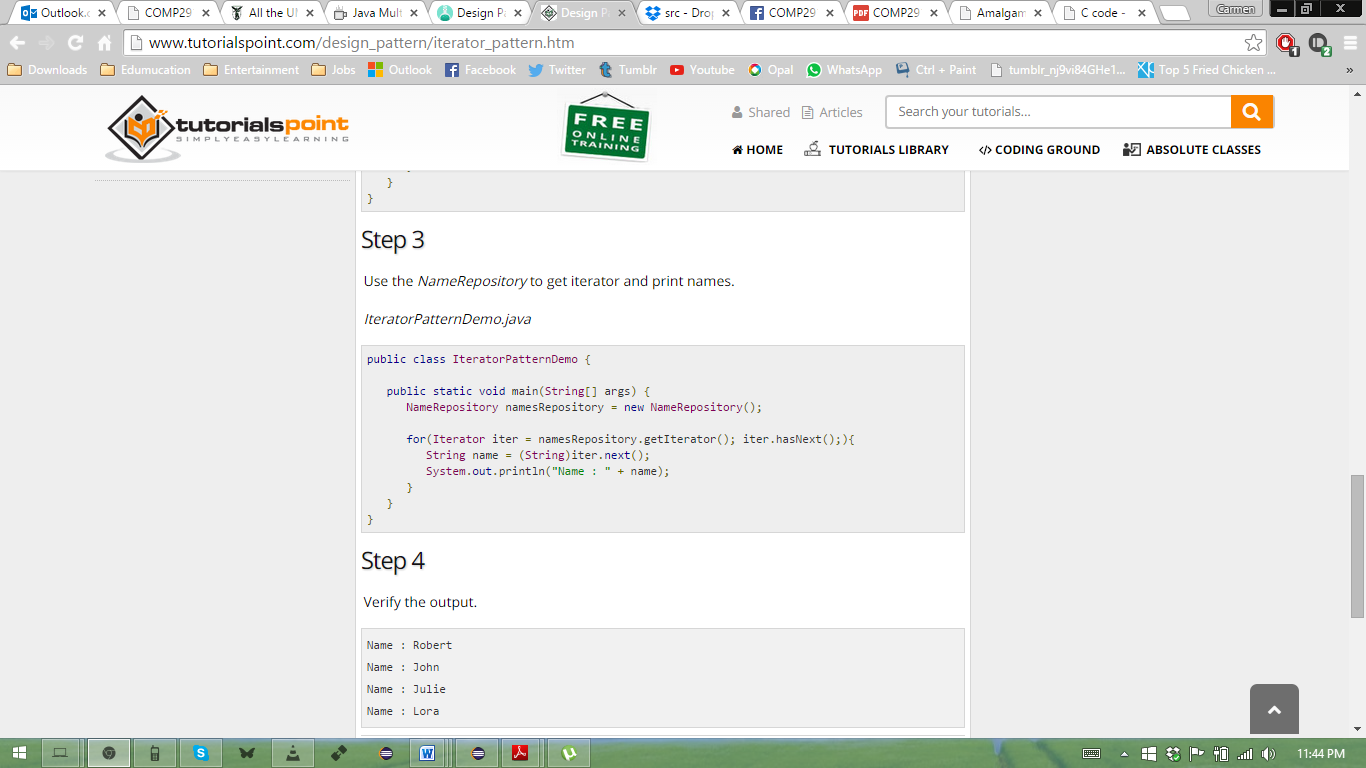




**A\* Class(Strategy strategy) 🡪 Method executeStrategy(int num1, num2) that returns strategy.doOperation(num1, num2) 🡪 OperationAdd implements Strategy interface**

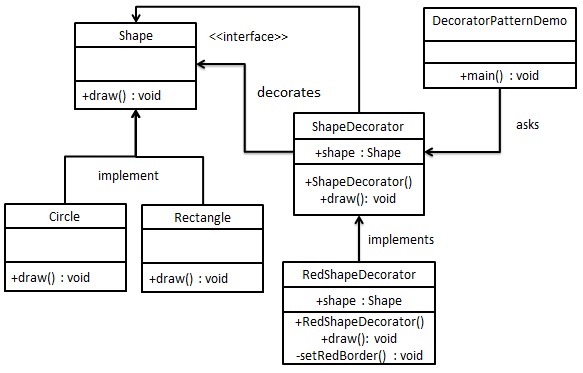
**Iterator Pattern:**

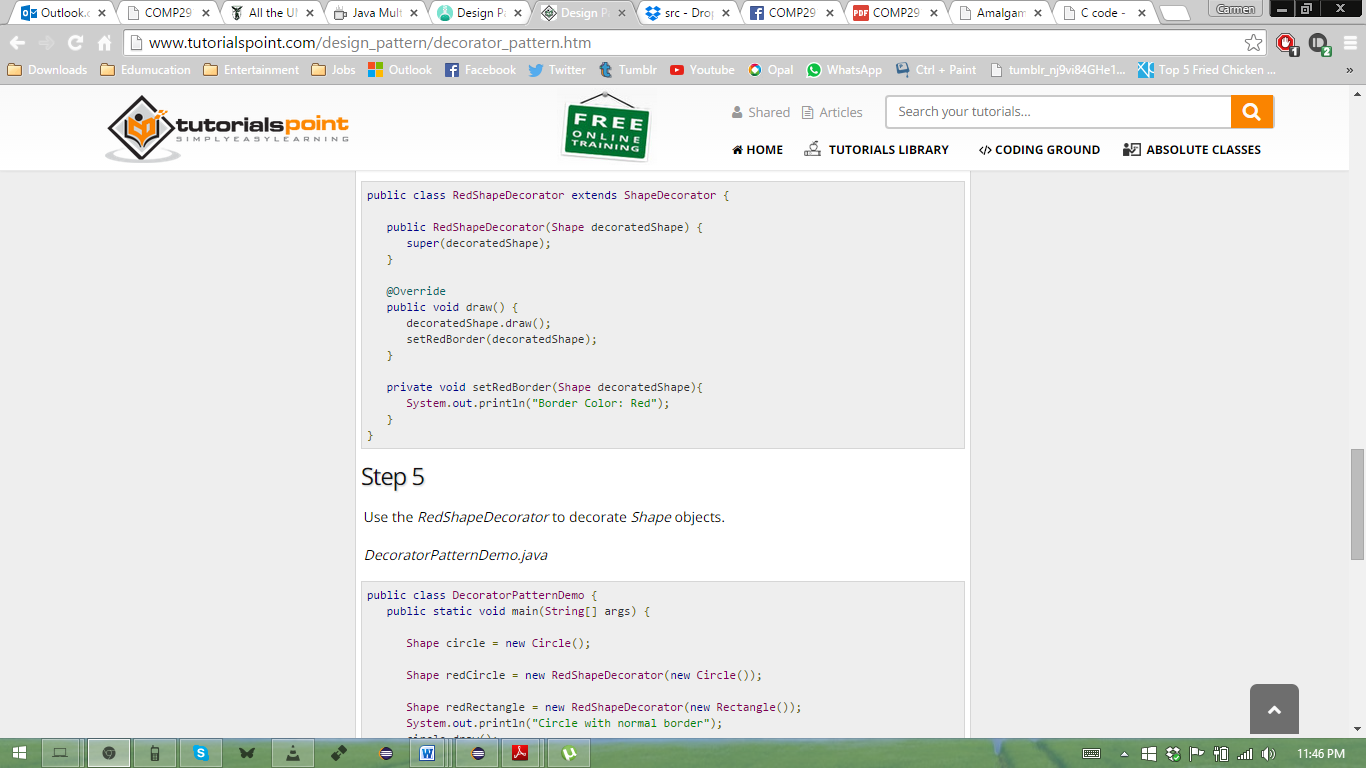




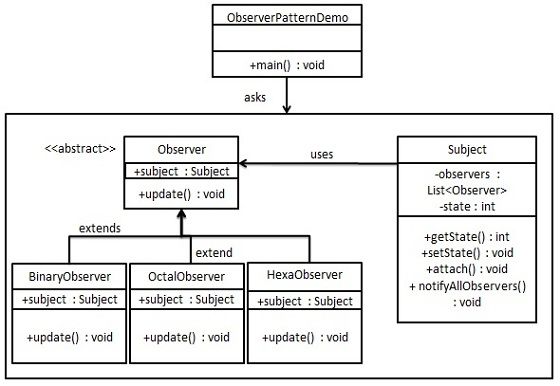
**A class that implements an interface with a getIterator() method 🡪 In that class, is a private class NameIterator that implements iterator interface 🡪 when getIterator() is called, it returns new NameIterator();**

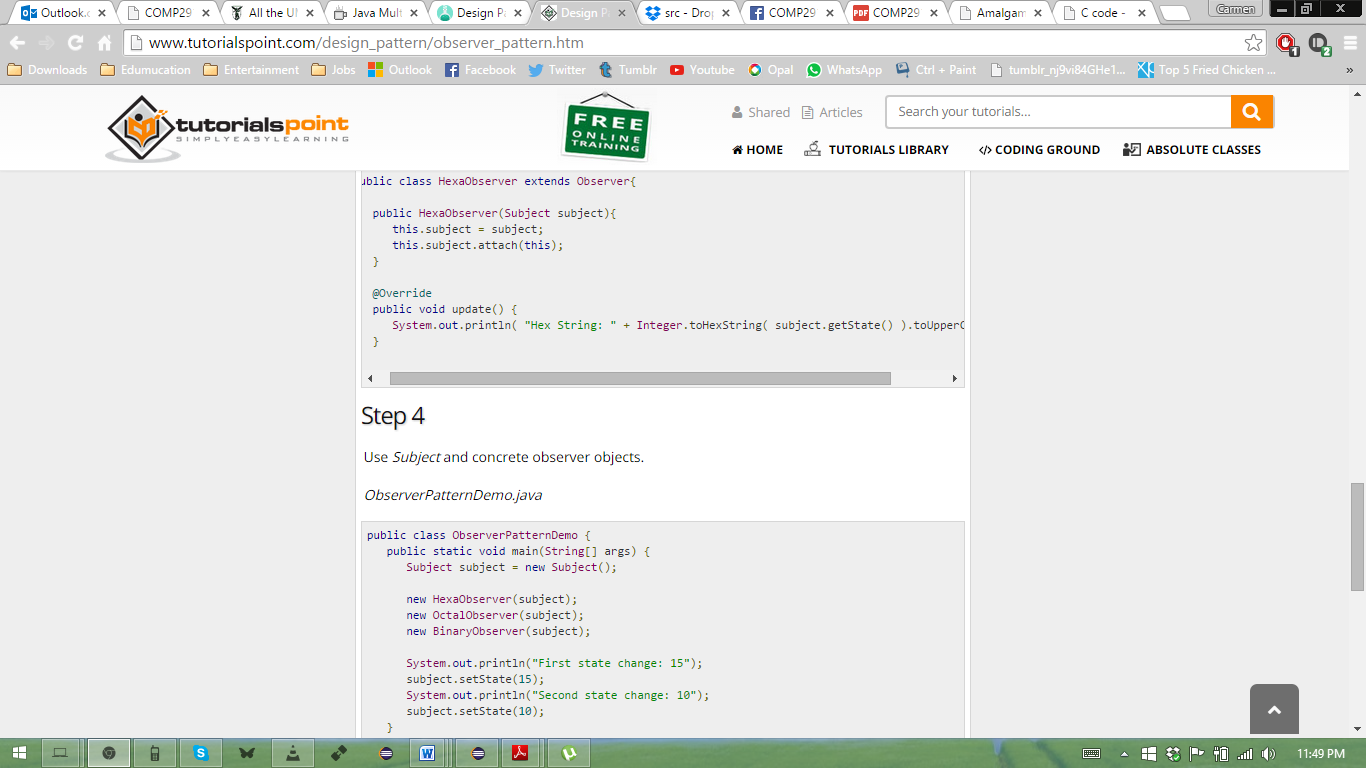
**Decorator Pattern:**



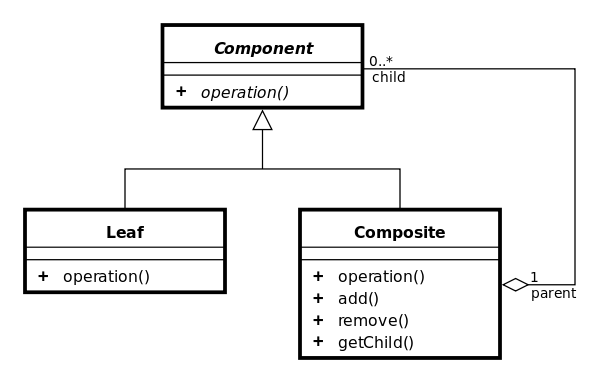


**Observer Pattern:**





**Composite Pattern:**



**public** **abstract** **class** ComponentDecorator **implements** Component {

**protected** Component decoratedComponent;

**public** ComponentDecorator(Component comp) {

**this**.decoratedComponent = comp;

}

**public** **int** getCost() {

**return** decoratedComponent.getCost();

}

}

**public** **class** HalfOffDecorator **extends** ComponentDecorator {

**public** HalfOffDecorator(Component comp) {

**super**(comp);

}

@Override

**public** **int** getCost() {

**return** applyDiscount(**super**.getCost());

}

**private** **int** applyDiscount(**int** cost) {

**return** cost/2;

}

}