Strategy Design Pattern

Discussion

In Strategy pattern, a class behavior or its algorithm can be changed at run time. This type of design pattern comes under behavior pattern.

In Strategy pattern, we create objects which represent various strategies and a context object whose behavior varies as per its strategy object. The strategy object changes the executing algorithm of the context object.
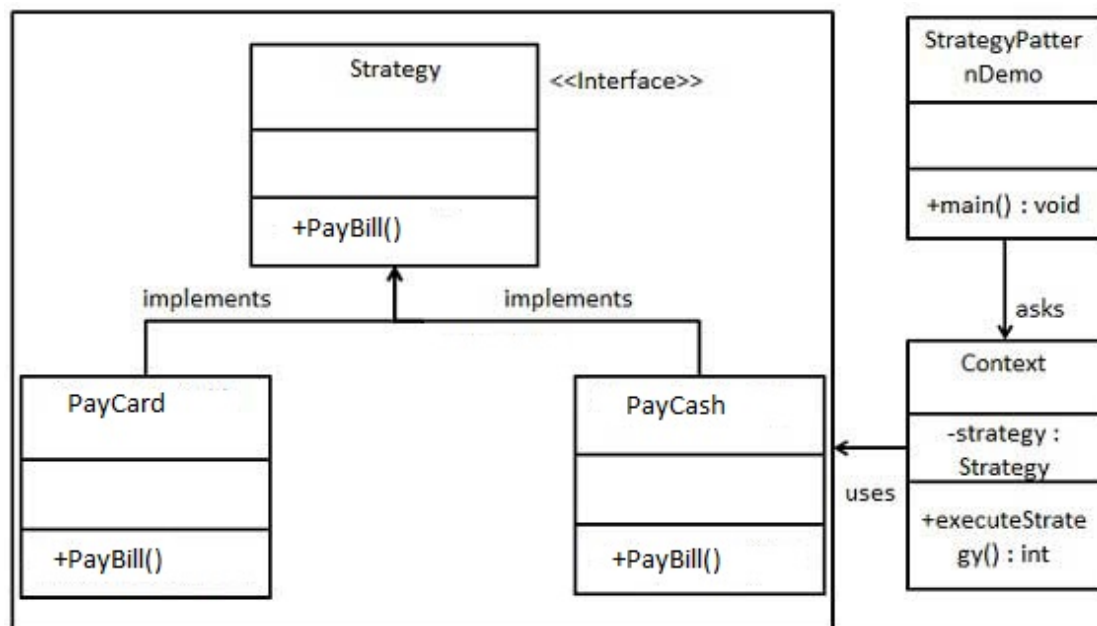
We are going to create a Strategy interface defining an action and concrete strategy classes implementing the Strategy interface. Context is a class which uses a Strategy.

StrategyPatternDemo, our demo class, will use Context and strategy objects to demonstrate change in Context behaviour based on strategy it deploys or uses.

Example

Consider Any shopping system. Payment strategy can be pay by cash or pay by card.

Diagram:

Implementation:

~~Class~~/ Interface: StratInterface

```java
// Create the Interace which has Paybill() functionality
package strategyP;

public interface  StratInterface {
      public void Paybill();

}
```

Class: Context

```java
//Create the Context which will take care of execution of PayBill() w.r.t. strategy
package strategyP;

public class Context {
        private StratInterface strategy;

        public Context(StratInterface strategy){
          this.strategy = strategy;
        }

        public void executeStrategy(){
          strategy.Paybill();
        }
    }
```

Class: PayCard

```java
//Concrete Class Pay by card implementing interface
package strategyP;

public class PayCard implements StratInterface{
        @Override
        public void Paybill() {
            System.out.println(" pay by card");
        }
    }
```

Class: PayCash

```java
//Concrete Class Pay by Cash implementing interface

package strategyP;

public class PayCash implements StratInterface{
        @Override
        public void Paybill() {
            System.out.println(" pay by cash");
```

```
            }
        }
```

//Use the Context to see change in behaviour when it changes its Strategy.

```
package strategyP;

public class StratDemo {
        public static void main(String[] args) {

            //Here we are using the strategy of PayCash and setting the context
            Context context = new Context(new PayCash());
            context.executeStrategy();

            //Here we are using the strategy of PayCard and setting the context
            context = new Context(new PayCard());
            context.executeStrategy();


        }
    }
```

Output:

```
pay by cash
pay by card
```