Team #3

Design Pattern

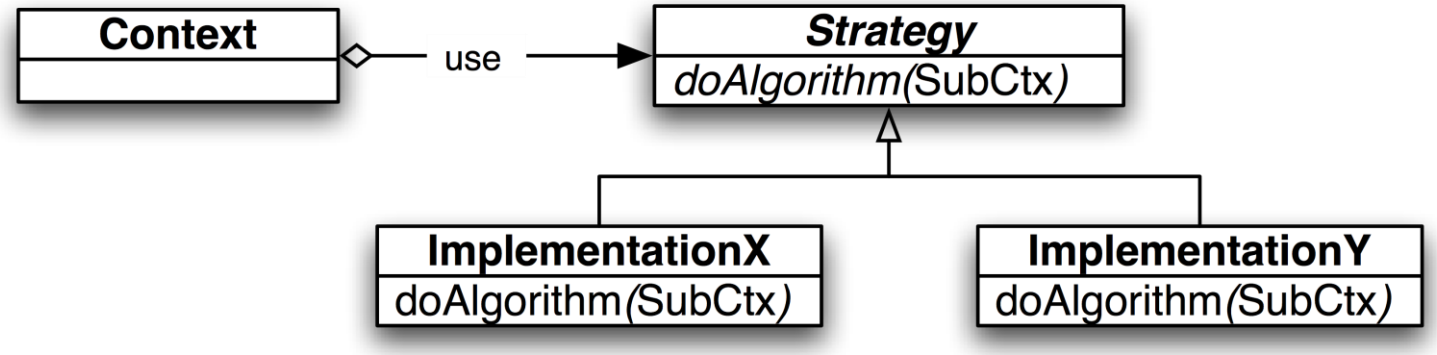
Strategy Pattern

- **Bill Capps**
- **Doug Hoskisson**
- **Rakan Alanazi**

UMKC

Strategy Pattern

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



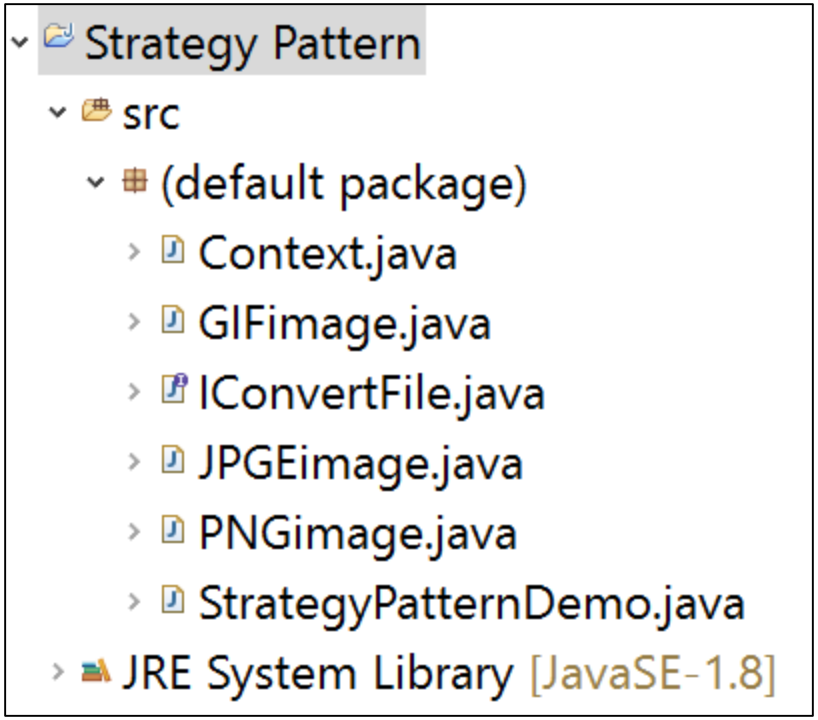
Strategy Pattern

- **Pros**
  - Enables the clients to choose the required algorithm, without using a "switch" statement or a series of "if-else" statements.
  - The strategy pattern application can switch strategies at run-time.

- **Cons**
  - Increase the number of objects in an application.
  - The client must be aware of all the strategies to select the right one for the right situation.

UMKC

Implementation



PNGimage Class

```
1
2 public class PNGimage implements IConvertFile    {
3
4     @Override
5     public String doConvert(String file) {
6         return " has been converted to PNG image";
7     }
8
9 }
```

JPGEimage Class

```
1
2 public class JPGEimage implements IConvertFile {
3
4     @Override
5     public String doConvert(String file) {
6         return " has been converted to JPGE image";
7
8     }
9
10 }
```

GIFimage

```
1
2 public class GIFimage implements IConvertFile {
3
4     @Override
5     public String doConvert(String file) {
6         return " has been converted to GIF image";
7     }
8
9 }
10
```

Context and IConvertFile Classes

```java
public class Context {
    private IConvertFile strategy;

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

    public String executeStrategy(String file){
        return strategy.doConvert(file);
    }
}
```

```java

public interface IConvertFile {
        public String doConvert(String file );

}

```

Demo Classes

```
1  public class StrategyPatternDemo {
2      public static void main(String[] args) {
3          Context context = new Context(new PNGimage());
4          System.out.println("img1"+context.executeStrategy("image"));
5
6          context = new Context(new JPGEimage());
7          System.out.println("img1"+context.executeStrategy ("image"));
8
9          context = new Context(new GIFimage());
10         System.out.println("img1"+context.executeStrategy("image"));
11     }
12 }
```

The Output

Problems Javadoc Declaration Console

```
<terminated> StrategyPatternDemo [Java Application] C:\Program Files\Java\
img1 has been converted to PNG image
img1 has been converted to JPGE image
img1 has been converted to GIF image
```

UMKC