Template Method Pattern

하위 클래스에서 구체적으로 처리.

- 상위 클래스에서 처리의 뼈대를 세우고, 구체적인 처리를 하위 클래스에서 실행.

Architecture, Modeling, Design Pattern, Behavioral

GUI 앱에서 Menu, Sidebar를 열기/닫기/크기 얻기 동작을 한다고 가정.

AbstractClass

```
/**
 * Template 역할
 */
public abstract class AbstractLayout {
    private int width;
    private int height;
    public AbstractLayout(int width, int height) {
        this.width = width;
        this.height = height;
    }
    public abstract void open();
    public abstract void close();
    public final int[] getMesasure() {
        return new int[] {width, height};
    }
}
```

ConcreteClass

```
public class MenuLayout extends AbstractLayout {

    public MenuLayout(int width, int height) {
        super(width, height);
    }

    @Override
    public void open() {
        System.out.println("Menu opened!");
    }

    @Override
    public void close() {
```

```
        System.out.println("Menu closed!");
    }
}
```

```
public class SidebarLayout extends AbstractLayout {

    public SidebarLayout(int width, int height) {
        super(width, height);
    }

    @Override
    public void open() {
        System.out.println("Sidebar opened!");
    }

    @Override
    public void close() {
        System.out.println("Sidebar closed!");
    }
}
```

Client

```
public class Main {
    public static void main(String[] args) {
        AbstractLayout menuLayout = new MenuLayout(100, 100);
        AbstractLayout sidebarLayout = new SidebarLayout(100, 500);
        print(menuLayout);
        print(sidebarLayout);
    }
    private static void print(AbstractLayout layout) {
        layout.open();
        layout.close();
        for(int value : layout.getMesasure()) {
            System.out.println(value);
        }
        System.out.println();
    }
}

/*
Menu opened!
Menu closed!
100
100
```

```
Sidebar opened!
Sidebar closed!
100
500
*/
```

From:
https://ledyx.xyz/ - **Ledyx WIKI**

Permanent link:
https://ledyx.xyz/doku.php?id=design_pattern:template_method_pattern

Last update: **2021/02/07 03:30**