

# Design Patterns

## W9A1

**Topic:** - Write a Program to implement Adapter Design Pattern

**Github Link:-** <https://github.com/RajAditya01/Design-And-Pattern>

### Definition :-

Adapter pattern works as a bridge between two incompatible interfaces. This type of design pattern comes under structural pattern as this pattern combines the capability of two independent interfaces.

This pattern involves a single class which is responsible to join functionalities of independent or incompatible interfaces. A real life example could be a case of card reader which acts as an adapter between memory card and a laptop. You plugin the memory card into card reader and card reader into the laptop so that memory card can be read via laptop.

### Code: -

```
package adapter;
```

```
interface WebDriver {  
    public void getElement();  
    public void selectElement();  
}
```

```
class ChromeDriver implements WebDriver {
```

```
    @Override
```

```
    public void getElement() {  
        System.out.println("Get element from ChromeDriver");  
    }
```

```
    @Override
```

```
    public void selectElement() {  
        System.out.println("Select element from ChromeDriver");  
    }
```

}

}

**class** IEDriver {**public void** findElement() {

System.out.println("Find element from IEDriver");

}

**public void** clickElement() {

System.out.println("Click element from IEDriver");

}

}

**class** WebDriverAdapter **implements** WebDriver {

IEDriver ieDriver;

**public** WebDriverAdapter(IEDriver ieDriver) {**this**.ieDriver = ieDriver;

}

@Override

**public void** getElement() {

ieDriver.findElement();

}

@Override

**public void** selectElement() {

ieDriver.clickElement();

}

}

**public class** AdapterDesignPattern {**public static void** main(String[] args) {ChromeDriver a = **new** ChromeDriver();

a.getElement();

a.selectElement();

```
IEDriver e = new IEDriver();  
e.findElement();  
e.clickElement();
```

```
WebDriver wID = new WebDriverAdapter(e);  
wID.findElement();  
wID.selectElement();
```

```
}
```

```
}
```

```
}
```

Output: -

```
PS D:\Design-And-Pattern> cd "d:\Design-And-Pattern\" ;  
Get element from ChromeDriver  
Select element from ChromeDriver  
Find element from IEDriver  
Click element from IEDriver  
Find element from IEDriver  
Click element from IEDriver  
PS D:\Design-And-Pattern>
```