

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний Технічний Університет України  
«Київський Політехнічний Інститут»  
Факультет інформатики та обчислювальної техніки  
Кафедра обчислювальної техніки

Лабораторна робота №7  
з дисципліни «Інженерія програмного забезпечення»  
на тему: «Шаблони поведінки. Шаблони Memento, State, Command, Interpreter»

Виконав:  
студент 2-го курсу ФІОТ  
групи ІВ-71  
Мазан Я.В.  
Номер залікової книжки: 7109  
Варіант: 5

Перевірив:  
доцент кафедри ОТ  
Антонюк А.І.

## Завдання

Варіант – 5

Визначити специфікації класів, що подають чергу HTTP-запитів на обробку. Реалізувати можливість виключення запитів з черги без обробки, та зміни позиції запиту через зміну значення пріоритету.

## Код програми

Main.java:

```
package com.lab111.labwork7;
/**
 * Command pattern is a data driven design pattern and falls under behavioral
 * pattern category. Main class.
 *
 * @author Yan Mazan
 * @version 1.0
 */
public class Main {
    public static void main(String[] args) {
        Request request = new Request("myRequest");
        ExclusionRequest exR = new ExclusionRequest(request);
        ChangeRequestPriority chR = new ChangeRequestPriority(request);
        Queue queue = new Queue();
        queue.takeOrder(exR);
        queue.takeOrder(chR);
        queue.execOrders();
    }
}
```

ExclusionRequest.java:

```
package com.lab111.labwork7;
/**
 * Concrete command classes and implementing Order interface which will do
 * actual command processing.
 *
 * @author Yan Mazan
 * @version 1.0
 */
public class ExclusionRequest implements Order {
    private Request request;
    public ExclusionRequest(Request request) {
        this.request = request;
    }
    @Override
    public void execute() {
        request.exclusion();
    }
}
```

ChangeRequestPriority.java:

```
package com.lab111.labwork7;
/**
 * Concrete command classes and implementing Order interface which will do
 * actual command processing.
 *
 * @author Yan Mazan
 * @version 1.0
 */
```

```

public class ChangeRequestPriority implements Order {
    private Request request;
    public ChangeRequestPriority(Request request) {
        this.request = request;
    }
    public void execute() {
        request.changePriority();
    }
}

```

Queue.java:

```

package com.lab111.labwork7;
import java.util.ArrayList;
import java.util.List;
/**
 * Class is created which acts as a invoker object.
 *
 * @author Yan Mazan
 * @version 1.0
 */
public class Queue {
    private List<Order> orderList = new ArrayList<>();
    public void takeOrder(Order order) {
        orderList.add(order);
    }
    public void execOrders() {
        for (Order order : orderList) {
            order.execute();
        }
        orderList.clear();
    }
}

```

Request.java:

```

package com.lab111.labwork7;
/**
 * Class which acts as a request.
 *
 * @author Yan Mazan
 * @version 1.0
 */
public class Request extends Queue {
    public String name;
    public Request(String name) {
        this.name = name;
    }
    public void exclusion() {
        System.out.println("Method name: exclusion. Request name: " + this.name);
    }
    public void changePriority() {
        System.out.println("Method name: changePriority. Request name: " + this.name);
    }
}

```

Order.java:

```

package com.lab111.labwork7;
/**
 * Class which acts as a request.
 *
 * @author Yan Mazan
 * @version 1.0
 */
public class Request extends Queue {

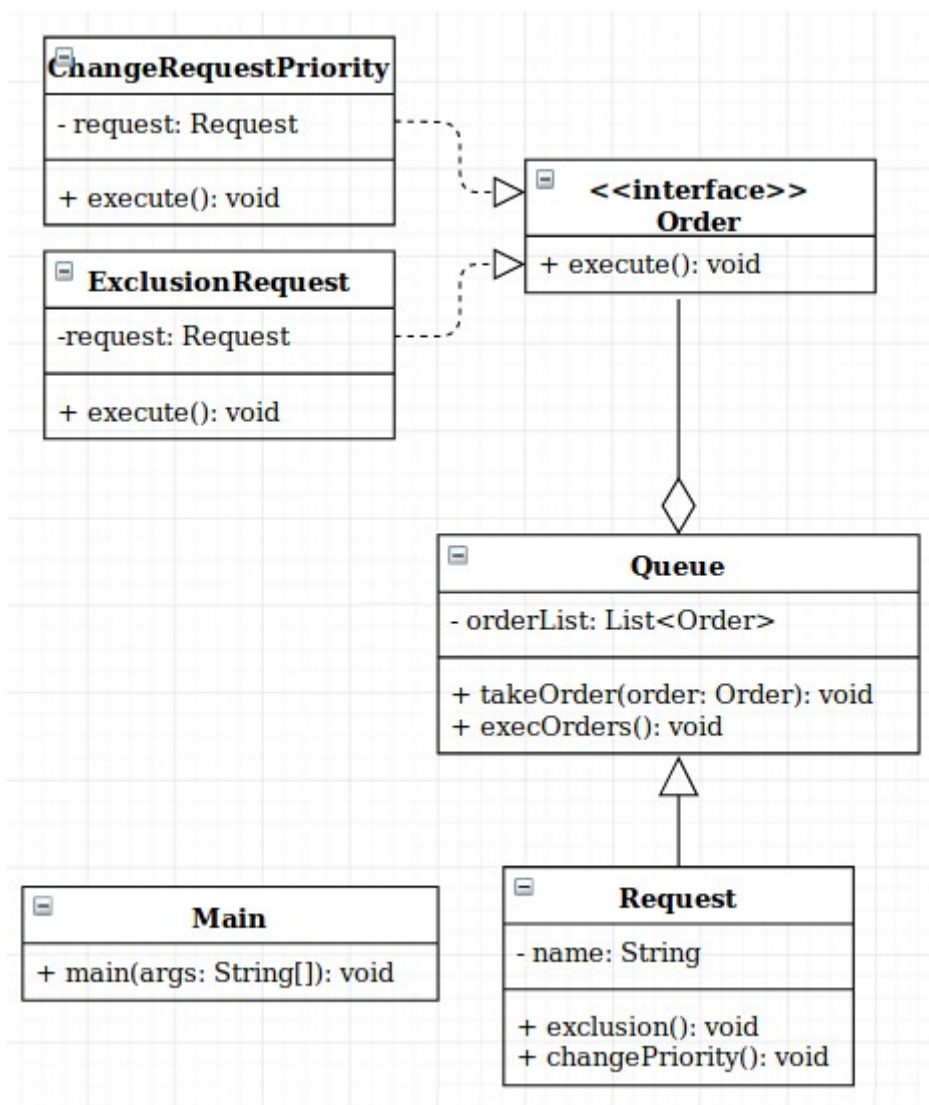
```

```

public String name;
public Request(String name) {
    this.name = name;
}
public void exclusion() {
    System.out.println("Method name: exclusion. Request name: " + this.name);
}
public void changePriority() {
    System.out.println("Method name: changePriority. Request name: " + this.name);
}
}

```

## Діаграма класів



## Тестування програми

```

/usr/lib/jvm/java-11.0-openjdk-amd64/bin/java -java:
Method name: exclusion. Request name: myRequest
Method name: changePriority. Request name: myRequest

Process finished with exit code 0

```

## **Висновок**

Під час виконання лабораторної роботи я вивчив шаблони проектування Memento, State, Command та Interpreter: для виконання даного мені варіанту було використано шаблон Command. Під час виконання роботи проблем не виникло.