

Университет ИТМО
Факультет программной инженерии и компьютерной техники

Лабораторная работа
по Программированию №3

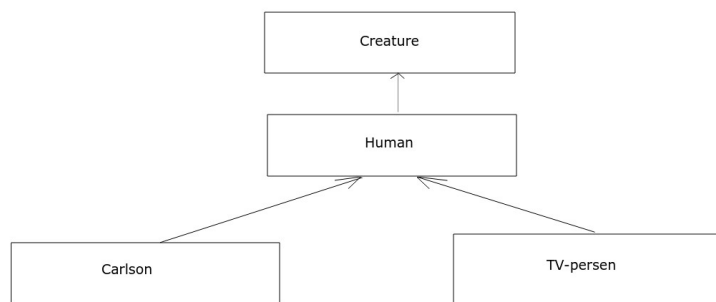
Преподаватель: Гаврилов А. В.
Выполнил: Щербаков Александр Валерьевич
Группа: Р3110
Вариант: 101112

Санкт-Петербург
2020

Задание:

Дикторша продолжала улыбаться Карлсону, и он улыбался ей в ответ. Потом он оттолкнул Малыша в сторону: Вдруг дикторша исчезла. Вместо нее на экране возникли два серьезных полных господина, которые все болтали и болтали. Карлсону это пришлось не по душе. Он начал нажимать на все кнопки и вертеть все ручки.

Диаграмма наследования классов



Исходный код программы

Carlson.java

```
package pack;

public final class Carlson extends Human {
    public Carlson(String name) {
        super(name);
    }
}
```

package pack;

Creature.java

```
public abstract class Creature implements shielded{
    private String name;

    public Creature(String name) {
        this.name = name;
    }
}
```

```

}

    public String getName() {
        return name;
    }

    public void push(Creature c) {
        System.out.println(this.toString() + " push " + c.toString());
    }

    @Override
    public String toString(){
        return this.getName();
    }
}

```

Human.java:

```

package pack;

public class Human extends Creature{

    public Human(String name) {
        super(name);
        condition = Condition.NORMAL;
    }
    public void smile() {
        System.out.println(this.toString() + " smiles");
    }
    public void smile(Human h) {
        System.out.println(this.toString() + " smiles at " + h.toString());
    }
    public void talk() {
        System.out.println(this.toString() + " talking");
    }

    public void pushButtons() {
        System.out.println(this.toString() + " push all buttons");
    }

    public Condition condition;
    @Override
    public void appeared() {
    }
    @Override
    public void disappeared() {
    }
}

enum Condition{
    NORMAL, IRRITATED
}

```

lab3.java

```

package pack;

public class lab3 {

    public static void main(String[] args) {
        Carlson carlson = new Carlson("Carlson");
        TV announcer = new TV("announcer");
    }
}

```

```

        Human baby = new Human("Baby");
        TV sirs = new TV("serious and fat gentlemen");

        announcer.smile();
        carlson.smile(announcer);

        carlson.push(baby);
        announcer.disappeared();

        sirs.appeared();
        sirs.talk();

        carlson.condition = Condition.IRRITATED;
        System.out.println("Carlson became irritated");

        carlson.pushButtons();
    }
}

```

Shielded.java:

```

package pack;

public interface shielded {
    public void appeared();
    public void disappeared();
}

```

TV.java:

```

package pack;

public final class TV extends Human{
    public TV(String name) {
        super(name);
    }
    public void appeared() {
        System.out.print(this.toString() + " appear on screen ");
    }

    public void disappeared() {
        System.out.println(this.toString() + " disappeared from the screen
");
    }
}

```

Результат работы программы

```

announcer smiles
Carlson smiles at announcer
Carlson push Baby
announcer disappeared from the screen
serious and fat gentlemen appear on screen serious and fat gentlemen talking
Carlson became irritated
Carlson push all buttons

```

Вывод по работе

Вывод: в ходе выполнения работы я изучил принципы SOLID и научился применять их в своей работе.