

The Government of the Russian Federation

**The Federal State Autonomous Institution of Higher Education
"National Research University - Higher School of Economics"**

National Research University «Higher School of Economics»

Faculty of Information Technology and Computer Engineering
Department of Computer Systems and Networks

Course title: Network computing

Practical training № 7. Multithreading.

Senior lecturer

Baybikova T.N.

Student:

Anna Kalinina

Date: 05.06.2019

Grade:

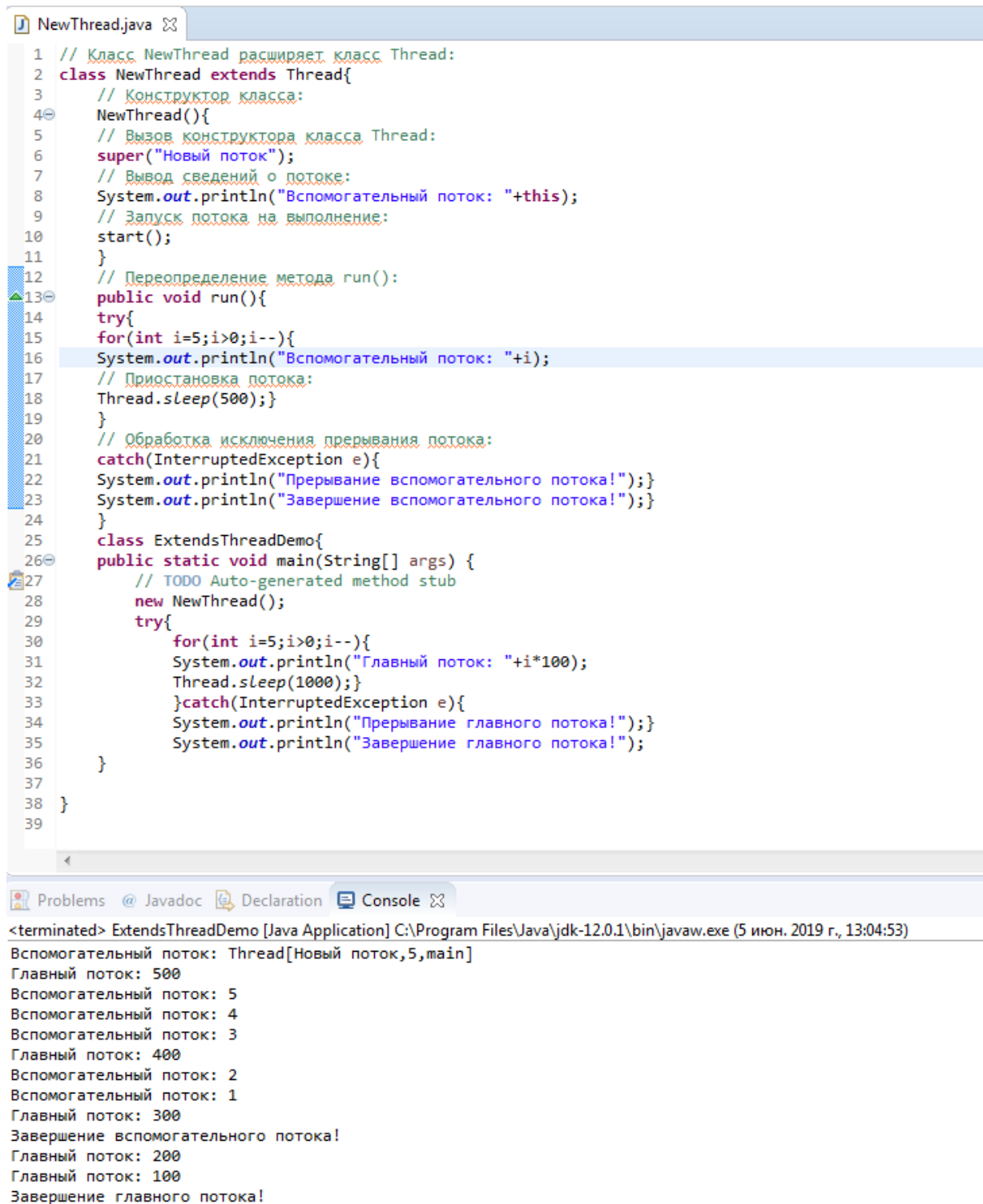
Moscow 2019

Practical training № 7. Multithreading.

Goal (цель): Create a thread and give a description of the methods and operations used for multithreading.

Variant: (вариант):

Listings (листинги) and **Figures** (рисунки):



```
NewThread.java
1 // Класс NewThread расширяет класс Thread:
2 class NewThread extends Thread{
3     // Конструктор класса:
4     NewThread(){
5         // Вызов конструктора класса Thread:
6         super("Новый поток");
7         // Вывод сведений о потоке:
8         System.out.println("Вспомогательный поток: "+this);
9         // Запуск потока на выполнение:
10        start();
11    }
12    // Переопределение метода run():
13    public void run(){
14        try{
15            for(int i=5;i>0;i--){
16                System.out.println("Вспомогательный поток: "+i);
17                // Приостановка потока:
18                Thread.sleep(500);}
19        }
20        // Обработка исключения прерывания потока:
21        catch(InterruptedException e){
22            System.out.println("Прерывание вспомогательного потока!");}
23        System.out.println("Завершение вспомогательного потока!");}
24    }
25    class ExtendsThreadDemo{
26    public static void main(String[] args) {
27        // TODO Auto-generated method stub
28        new NewThread();
29        try{
30            for(int i=5;i>0;i--){
31                System.out.println("Главный поток: "+i*100);
32                Thread.sleep(1000);}
33            }catch(InterruptedException e){
34                System.out.println("Прерывание главного потока!");}
35            System.out.println("Завершение главного потока!");
36        }
37    }
38 }
39
```

Problems @ Javadoc Declaration Console

<terminated> ExtendsThreadDemo [Java Application] C:\Program Files\Java\jdk-12.0.1\bin\javaw.exe (5 июн. 2019 г., 13:04:53)

Вспомогательный поток: Thread[Новый поток,5,main]
Главный поток: 500
Вспомогательный поток: 5
Вспомогательный поток: 4
Вспомогательный поток: 3
Главный поток: 400
Вспомогательный поток: 2
Вспомогательный поток: 1
Главный поток: 300
Завершение вспомогательного потока!
Главный поток: 200
Главный поток: 100
Завершение главного потока!

Listing:

```
// Класс NewThread расширяет класс Thread:
class NewThread extends Thread{
    // Конструктор класса:
    NewThread(){
        // Вызов конструктора класса Thread:
        super("Новый поток");
        // Вывод сведений о потоке:
        System.out.println("Вспомогательный поток: "+this);
        // Запуск потока на выполнение:
        start();
    }
    // Переопределение метода run():
    public void run(){
        try{
            for(int i=5;i>0;i--){
                System.out.println("Вспомогательный поток: "+i);
                // Приостановка потока:
                Thread.sleep(500);}
            }
        // Обработка исключения прерывания потока:
        catch(InterruptedException e){
            System.out.println("Прерывание вспомогательного потока!");}
        System.out.println("Завершение вспомогательного потока!");}
    }
    class ExtendsThreadDemo{
        public static void main(String[] args) {
            // TODO Auto-generated method stub
            new NewThread();
            try{
                for(int i=5;i>0;i--){
                    System.out.println("Главный поток: "+i*100);
                    Thread.sleep(1000);}
                }catch(InterruptedException e){
                    System.out.println("Прерывание главного потока!");}
                System.out.println("Завершение главного потока!");}
        }
    }
}
```

Conclusions (ВЫВОДЫ):

A thread in Java is an object of the class Thread. This case represents creating a thread by extending the Thread class.

In the NewThread class, which inherits the Thread class, the constructor and the run () method are defined. In the constructor, the super command ("New thread") invokes the constructor of the class Thread with an argument — the name of the thread being created. Displaying information about the stream is carried out by the command: System.out.println ("Auxiliary stream:" + this)

In the NewThread class constructor, using the t = new Thread (this, "New thread") field, the t field is assigned a value to the created Thread class object. The object is created on the basis of the object of the class NewThread, whose field is the object variable t (the first argument of the constructor is the keyword this), and the name of the stream is given by the second argument of the constructor — in this case, the

string "New stream". The `System.out.println ("Auxiliary Stream:" + i)` command displays information about the created stream. All these actions are described in the `NewThread` class constructor, that is, they are performed when creating an object of this class. When executing the program, two processes (threads) are superimposed on each other: the main stream of the message is output at one second interval, and the child thread of the message is displayed at 0.5 second interval. Since the number of messages displayed in each of the message flows is the same, and the interval between the messages of the main stream is larger than the interval between the messages of the child stream, the first one ends with the child stream, and its messages appear "heap". As a reference to the stream object, the `this` link to the created object is used. Starts a thread by calling the object's `start ()` method.

References:

- 'MultiThread' project files
- Learning materials in LMS