

Задание 1

Выводим список команд с описанием используя команды `javac` и `java`

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.6456]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\veon\Documents\oop\first>javac
Usage: javac [options] [source files]
where possible options include:
  @<filename>          Read options and filenames from file
  -Akey[=value]         Options to pass to annotation processors
  --add-modules <module>[,<module>]*
                        Root modules to resolve in addition to the initial modules,
                        or all modules on the module path if <module> is ALL-MODULE-PATH.
  --boot-class-path <path>  -bootclasspath <path>
                        Override location of bootstrap class files
  --class-path <path>  -classpath <path>  -cp <path>
                        Specify where to find user class files and annotation processors
  -d <directory>         Specify where to place generated class files
  -deprecation           Output source locations where deprecated APIs are used
  --enable-preview       Enable preview language features.
                        To be used in conjunction with either -source or --release.
  -encoding <encoding>   Specify character encoding used by source files
  -endorseddirs <dirs>   Override location of endorsed standards paths
  -extdirs <dirs>        Override location of installed extensions
  -g                    Generate all debugging info
  -g:lines,vars,source   Generate only some debugging info
  -g:none               Generate no debugging info
  -h <directory>         Specify where to place generated native header files
  --help, -help, -?     Print this help message
  --help-extra, -X      Print help on extra options
  -implicit:[none,class]
                        Specify whether to generate class files for implicitly referenced files
  -J<flag>              Pass <flag> directly to the runtime system
  --limit-modules <module>[,<module>]*
                        Limit the universe of observable modules
  --module <module>[,<module>]*, -m <module>[,<module>]*
                        Compile only the specified module(s), check timestamps
  --module-path <path>  -p <path>
                        Specify where to find application modules
  --module-source-path <module-source-path>
                        Specify where to find input source files for multiple modules
  --module-version <version>
                        Specify version of modules that are being compiled
  -nowarn               Generate only mandatory warnings
  -parameters          Generate metadata for reflection on method parameters
  -proc:[none,only,full]
                        Control whether annotation processing and/or compilation is done.
  -processor <class>[,<class>[,<class>]...]
                        Names of the annotation processors to run;
                        bypasses default discovery process
  --processor-module-path <path>
                        Specify a module path where to find annotation processors
  --processor-path <path>  -processorpath <path>
                        Specify where to find annotation processors
  -profile <profile>     Check that API used is available in the specified profile.
                        This option is deprecated and may be removed in a future release.
  --release <release>   Compile for the specified Java SE release.
                        Supported releases:
                        8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25
  -s <directory>         Specify where to place generated source files
  --source <release>  -source <release>
                        Provide source compatibility with the specified Java SE release.
                        Supported releases:
                        8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25
  --source-path <path>  -sourcepath <path>
                        Specify where to find input source files
  --system <jdk>[none]  Override location of system modules
  --target <release>  -target <release>
                        Generate class files suitable for the specified Java SE release.
                        Supported releases:
                        8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25
  --upgrade-module-path <path>
                        Override location of upgradeable modules
  -verbose              Output messages about what the compiler is doing
  --version, -version   Version information
  -Werror               Terminate compilation if warnings occur

C:\Users\veon\Documents\oop\first>java
Usage: java [java options...] <application> [<application arguments...>]
where <application> is one of:
  <mainclass>          to execute the main method of a compiled main class
  -jar <jarfile>.-jar   to execute the main class of a JAR archive
  -m <module>[/<mainclass>]
                        to execute the main class of a module
  <sourcefile>.-java   to compile and execute a source-file program

where key java options include:
  --class-path <class path>
                        where <class path> is a list of directories and JAR archives to search for class files, separated by ";"
  --module-path <module path>
                        where <module path> is a list of directories and JAR archives to search for modules, separated by ";"
  --version
                        to print product version to the error stream and exit

for additional help on usage:      java --help
for an interactive Java environment: jshell

C:\Users\veon\Documents\oop\first>
```

```
C:\Windows\System32\cmd.exe
Control whether annotation processing and/or compilation is done.
-processor <class>[,<class>[,<class>]...]
Names of the annotation processors to run;
bypasses default discovery process
--processor-module-path <path>
Specify a module path where to find annotation processors
--processor-path <path>  -processorpath <path>
Specify where to find annotation processors
-profile <profile>
Check that API used is available in the specified profile.
This option is deprecated and may be removed in a future release.
--release <release>
Compile for the specified Java SE release.
Supported releases:
8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25
-s <directory>
Specify where to place generated source files
--source <release>  -source <release>
Provide source compatibility with the specified Java SE release.
Supported releases:
8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25
--source-path <path>  -sourcepath <path>
Specify where to find input source files
--system <jdk>[none]  Override location of system modules
--target <release>  -target <release>
Generate class files suitable for the specified Java SE release.
Supported releases:
8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25
--upgrade-module-path <path>
Override location of upgradeable modules
-verbose
Output messages about what the compiler is doing
--version, -version
Version information
-Werror
Terminate compilation if warnings occur

C:\Users\veon\Documents\oop\first>java
Usage: java [java options...] <application> [<application arguments...>]
where <application> is one of:
  <mainclass>          to execute the main method of a compiled main class
  -jar <jarfile>.-jar   to execute the main class of a JAR archive
  -m <module>[/<mainclass>]
                        to execute the main class of a module
  <sourcefile>.-java   to compile and execute a source-file program

where key java options include:
  --class-path <class path>
                        where <class path> is a list of directories and JAR archives to search for class files, separated by ";"
  --module-path <module path>
                        where <module path> is a list of directories and JAR archives to search for modules, separated by ";"
  --version
                        to print product version to the error stream and exit

for additional help on usage:      java --help
for an interactive Java environment: jshell

C:\Users\veon\Documents\oop\first>
```

Задание 2

По заданию создаю файл `MyFisrtProgram.java`, в котором создаю пустой класс `MyFisrtClass`. При попытке откомпилировать код (команда `javac MyFirstProgram.java`) получаю ошибку, так как метод `main` не найден

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.6456]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\xeon\Documents\oop\first>mkdir Task2

C:\Users\xeon\Documents\oop\first>cd Task2

C:\Users\xeon\Documents\oop\first\Task2>javac MyFirstProgram.java

C:\Users\xeon\Documents\oop\first\Task2>java MyFirstClass
Error: Main method not found in class MyFirstClass, please define the main method as:
  public static void main(String[] args)
or a JavaFX application class must extend javafx.application.Application

C:\Users\xeon\Documents\oop\first\Task2>
```

Чтобы исправить ошибку был добавлен метод main который выводит текст в консоль

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.6456]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\xeon\Documents\oop\first>mkdir Task2

C:\Users\xeon\Documents\oop\first>cd Task2

C:\Users\xeon\Documents\oop\first\Task2>javac MyFirstProgram.java

C:\Users\xeon\Documents\oop\first\Task2>java MyFirstClass
Error: Main method not found in class MyFirstClass, please define the main method as:
  public static void main(String[] args)
or a JavaFX application class must extend javafx.application.Application

C:\Users\xeon\Documents\oop\first\Task2>javac MyFirstProgram.java
MyFirstProgram.java:3: error: illegal character: '\u201c'
    System.out.println("Hello world!!!");
                        ^
MyFirstProgram.java:3: error: ';' expected
    System.out.println(?Hello world!!!?);
                        ^
MyFirstProgram.java:3: error: illegal character: '\u201d'
    System.out.println(?Hello world!!!?);
                        ^
3 errors

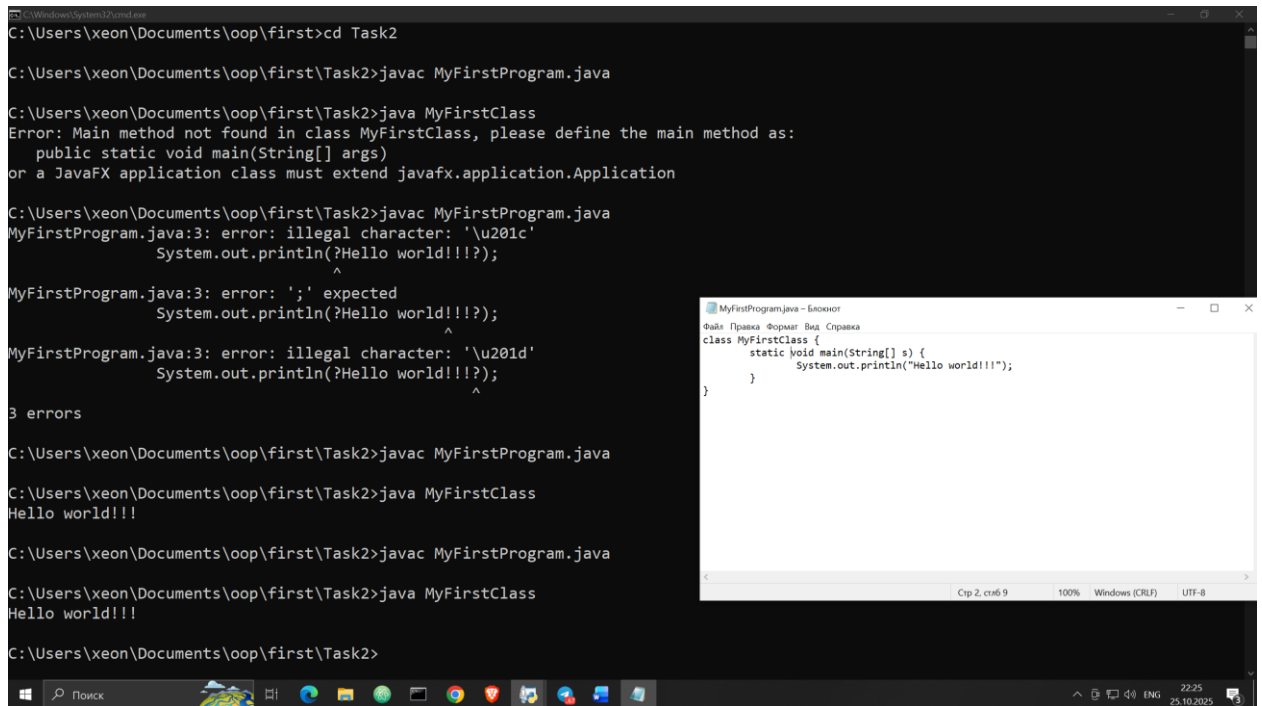
C:\Users\xeon\Documents\oop\first\Task2>javac MyFirstProgram.java

C:\Users\xeon\Documents\oop\first\Task2>java MyFirstClass
Hello world!!!

C:\Users\xeon\Documents\oop\first\Task2>
```

```
MyFirstProgram.java - Source
Файл Правка Формат Вид Справка
class MyFirstClass {
    void main(String[] s) {
        System.out.println("Hello world!!!");
    }
}
```

По заданию сделал метод статическим, снова откомпилировал и запустил программу



```
C:\Users\xeon\Documents\oop\first>cd Task2
C:\Users\xeon\Documents\oop\first\Task2>javac MyFirstProgram.java
C:\Users\xeon\Documents\oop\first\Task2>java MyFirstClass
Error: Main method not found in class MyFirstClass, please define the main method as:
    public static void main(String[] args)
or a JavaFX application class must extend javafx.application.Application

C:\Users\xeon\Documents\oop\first\Task2>javac MyFirstProgram.java
MyFirstProgram.java:3: error: illegal character: '\u201c'
    System.out.println(?Hello world!!!?);
                        ^
MyFirstProgram.java:3: error: ';' expected
    System.out.println(?Hello world!!!?);
                        ^
MyFirstProgram.java:3: error: illegal character: '\u201d'
    System.out.println(?Hello world!!!?);
                        ^
3 errors

C:\Users\xeon\Documents\oop\first\Task2>javac MyFirstProgram.java
C:\Users\xeon\Documents\oop\first\Task2>java MyFirstClass
Hello world!!!

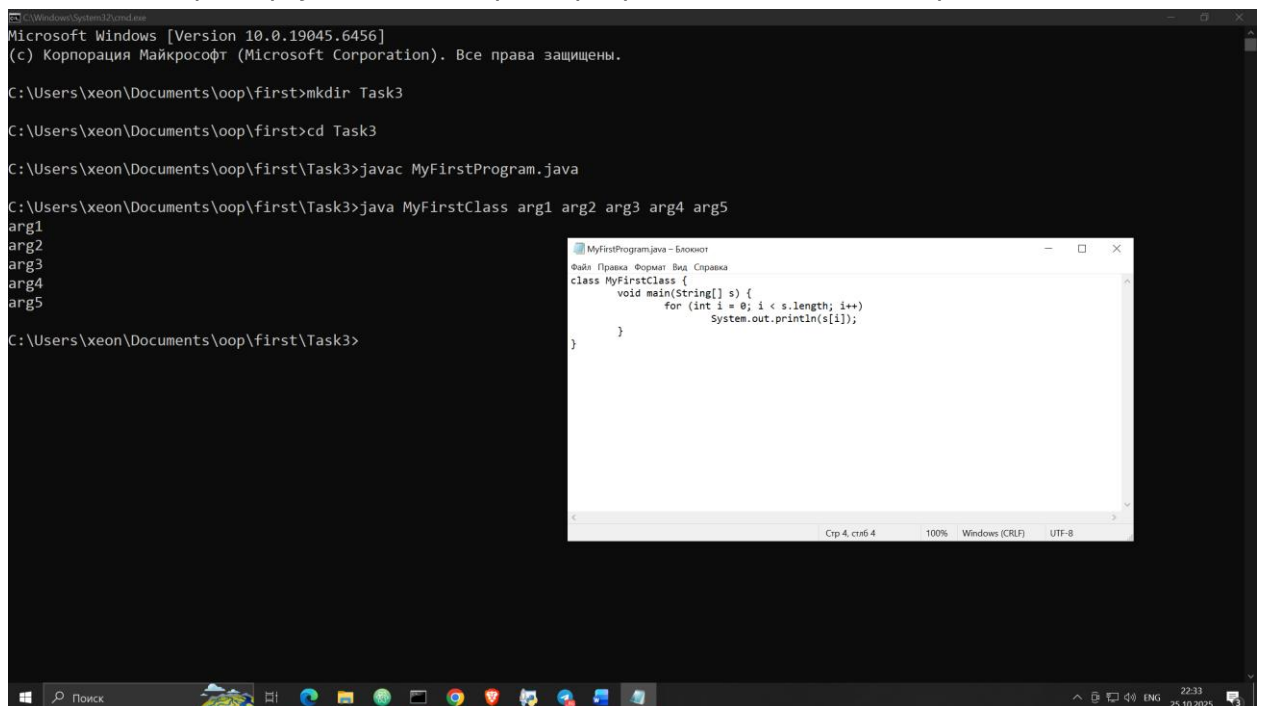
C:\Users\xeon\Documents\oop\first\Task2>javac MyFirstProgram.java
C:\Users\xeon\Documents\oop\first\Task2>java MyFirstClass
Hello world!!!

C:\Users\xeon\Documents\oop\first\Task2>
```

```
MyFirstProgram.java - Блокнот
Файл Правка Формат Вид Справка
class MyFirstClass {
    static void main(String[] s) {
        System.out.println("Hello world!!!");
    }
}
```

Задание 3

Согласно заданию заменил текст метода main. После компиляции, при запуске нужно было ввести ряд аргументов, которые программа выводила построчно в консоли



```
Microsoft Windows [Version 10.0.19045.6456]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\xeon\Documents\oop\first>mkdir Task3
C:\Users\xeon\Documents\oop\first>cd Task3
C:\Users\xeon\Documents\oop\first\Task3>javac MyFirstProgram.java
C:\Users\xeon\Documents\oop\first\Task3>java MyFirstClass arg1 arg2 arg3 arg4 arg5
arg1
arg2
arg3
arg4
arg5

C:\Users\xeon\Documents\oop\first\Task3>
```

```
MyFirstProgram.java - Блокнот
Файл Правка Формат Вид Справка
class MyFirstClass {
    void main(String[] s) {
        for (int i = 0; i < s.length; i++)
            System.out.println(s[i]);
    }
}
```

Задание 4

Согласно заданию был добавлен второй класс MySecondClass, который реализовывал следующую функциональность

- имеет два приватных поля типа int;
- методы для получения и модификации их значений;
- конструктор, создающий объект и инициализирующий значения полей;
- метод с возвращаемым типом int, реализующий над этими числами действие умножение

Код метода MyFirstClass.main() при этом был заменен на код из задания

The screenshot displays a Windows desktop environment. On the left, a Command Prompt window shows the following commands and output:

```
C:\Users\xeon\Documents\oop\first\Task4>javac MyFirstProgram.java
C:\Users\xeon\Documents\oop\first\Task4>java MyFirstClass
1 2 3 4 5 6 7 8
2 4 6 8 10 12 14 16
3 6 9 12 15 18 21 24
4 8 12 16 20 24 28 32
5 10 15 20 25 30 35 40
6 12 18 24 30 36 42 48
7 14 21 28 35 42 49 56
8 16 24 32 40 48 56 64
```

On the right, an IDE window titled 'MyFirstProgram.java - Блокнот' shows the following Java code:

```
class MyFirstClass {
    void main(String[] s) {
        for (int i = 0; i < s.length; i++)
            System.out.println(s[i]);
        int i, j;
        MySecondClass o = new MySecondClass(0, 0);
        for (i = 1; i <= 8; i++) {
            for (j = 1; j <= 8; j++) {
                o.setFirstNumber(i);
                o.setSecondNumber(j);
                System.out.print(o.multiply());
                System.out.print(" ");
            }
            System.out.println();
        }
    }
}

class MySecondClass {
    private int firstNumber;
    private int secondNumber;

    public MySecondClass(int first, int second) {
        firstNumber = first;
        secondNumber = second;
    }

    public void setFirstNumber(int value) {
        firstNumber = value;
    }

    public void setSecondNumber(int value) {
        secondNumber = value;
    }

    public int getFirstNumber() {
        return firstNumber;
    }

    public int getSecondNumber() {
        return secondNumber;
    }

    public int multiply() {
        return firstNumber * secondNumber;
    }
}
```

Задание 5

Были удалены все байт коды классов и создана директория myfirstrpackage, где внутри был создан файл MyFirstPackage.java. Произведена попытка откомпилировать файл MyFirstPackage.java

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.6456]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\xeon\Documents\oop\first\Task5>javac myfirstpackage\MyFirstPackage.java

C:\Users\xeon\Documents\oop\first\Task5>javac MyFirstProgram.java
MyFirstProgram.java:6: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
    ^
  symbol:   class MySecondClass
  location: class MyFirstClass
MyFirstProgram.java:6: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
    ^
  symbol:   class MySecondClass
  location: class MyFirstClass
2 errors

C:\Users\xeon\Documents\oop\first\Task5>
```

Добавлено в начало исходного кода MyFirstProgram import myfirstpackage.*;

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.6456]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\xeon\Documents\oop\first\Task5>javac myfirstpackage\MyFirstPackage.java

C:\Users\xeon\Documents\oop\first\Task5>javac MyFirstProgram.java
MyFirstProgram.java:6: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
    ^
  symbol:   class MySecondClass
  location: class MyFirstClass
MyFirstProgram.java:6: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
    ^
  symbol:   class MySecondClass
  location: class MyFirstClass
2 errors

C:\Users\xeon\Documents\oop\first\Task5>javac MyFirstProgram.java
MyFirstProgram.java:8: error: cannot access MySecondClass
    MySecondClass o = new MySecondClass(0, 0);
    ^
  bad class file: .\myfirstpackage\MySecondClass.class
    class file contains wrong class: MySecondClass
    Please remove or make sure it appears in the correct subdirectory of the classpath.
MyFirstProgram.java:8: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
    ^
  symbol:   class MySecondClass
  location: class MyFirstClass
2 errors
```

```
MyFirstProgram.java - Блокнот
Файл Правка Формат Вид Справка
import myfirstpackage.*;

class MyFirstClass {
    void main(String[] s) {
        for (int i = 0; i < s.length; i++)
            System.out.println(s[i]);
        int i, j;
        MySecondClass o = new MySecondClass(0, 0);
        for (i = 1; i <= 8; i++) {
            for (j = 1; j <= 8; j++) {
                o.setFirstNumber(i);
                o.setSecondNumber(j);
                System.out.print(o.multiply());
                System.out.print(" ");
            }
            System.out.println();
        }
    }
}
```

В MyFirstPackage.java был объявлен пакет используя package myfirstpackage

```
Microsoft Windows [Version 10.0.19045.6456]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\xeon\Documents\oop\first\Task5>javac myfirstpackage/MyFirstPackage.java

C:\Users\xeon\Documents\oop\first\Task5>javac MyFirstProgram.java
MyFirstProgram.java:6: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
                        ^
  symbol:   class MySecondClass
  location: class MyFirstClass
MyFirstProgram.java:6: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
                        ^
  symbol:   class MySecondClass
  location: class MyFirstClass
2 errors

C:\Users\xeon\Documents\oop\first\Task5>javac MyFirstProgram.java
MyFirstProgram.java:8: error: cannot access MySecondClass
    MySecondClass o = new MySecondClass(0, 0);
                        ^
  bad class file: .\myfirstpackage\MySecondClass.class
    class file contains wrong class: MySecondClass
    Please remove or make sure it appears in the correct subdirectory of the classpath
MyFirstProgram.java:8: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
                        ^
  symbol:   class MySecondClass
  location: class MyFirstClass
2 errors

C:\Users\xeon\Documents\oop\first\Task5>
C:\Users\xeon\Documents\oop\first\Task5>javac myfirstpackage/MyFirstPackage.java

C:\Users\xeon\Documents\oop\first\Task5>javac MyFirstProgram.java
MyFirstProgram.java:8: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
                        ^
  symbol:   class MySecondClass
  location: class MyFirstClass
MyFirstProgram.java:8: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
                        ^
  symbol:   class MySecondClass
  location: class MyFirstClass
2 errors

C:\Users\xeon\Documents\oop\first\Task5>
```

```
class MyFirstClass {
    void main(String[] s) {
        for (int i = 0; i < s.length; i++)
            System.out.println(s[i]);
        int i, j;
        MySecondClass o = new MySecondClass(0, 0);
        for (i = 1; i <= 8; i++) {
            for (j = 1; j <= 8; j++) {
                o.setFirstNumber(i);
                o.setSecondNumber(j);
                System.out.print(o.multiply());
                System.out.print(" ");
            }
            System.out.println();
        }
    }
}
```

```
package myfirstpackage;

class MySecondClass {
    private int firstNumber;
    private int secondNumber;

    public MySecondClass(int first, int second) {
        firstNumber = first;
        secondNumber = second;
    }

    public void setFirstNumber(int value) {
        firstNumber = value;
    }

    public void setSecondNumber(int value) {
        secondNumber = value;
    }

    public int getFirstNumber() {
        return firstNumber;
    }

    public int getSecondNumber() {
        return secondNumber;
    }

    public int multiply() {
        return firstNumber * secondNumber;
    }
}
```

Были добавлены модификаторы public static к методу мэйн, появилась ошибка что имя файла не совпадало с именем класса

```
symbol:   class MySecondClass
location: class MyFirstClass
MyFirstProgram.java:8: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
                        ^
2 errors

C:\Users\xeon\Documents\oop\first\Task5>javac MyFirstProgram.java
MyFirstProgram.java:8: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
                        ^
  symbol:   class MySecondClass
  location: class MyFirstClass
MyFirstProgram.java:8: error: cannot find symbol
    MySecondClass o = new MySecondClass(0, 0);
                        ^
  symbol:   class MySecondClass
  location: class MyFirstClass
2 errors

C:\Users\xeon\Documents\oop\first\Task5>
C:\Users\xeon\Documents\oop\first\Task5>javac myfirstpackage/MyFirstPackage.java
myfirstpackage/MyFirstPackage.java:3: error: class MySecondClass is public, should be
static
public class MySecondClass {
^
1 error

C:\Users\xeon\Documents\oop\first\Task5>
```

```
import myfirstpackage.*;

class MyFirstClass {
    public static void main(String[] s) {
        for (int i = 0; i < s.length; i++)
            System.out.println(s[i]);
        int i, j;
        MySecondClass o = new MySecondClass(0, 0);
        for (i = 1; i <= 8; i++) {
            for (j = 1; j <= 8; j++) {
                o.setFirstNumber(i);
                o.setSecondNumber(j);
                System.out.print(o.multiply());
                System.out.print(" ");
            }
            System.out.println();
        }
    }
}
```

```
package myfirstpackage;

public class MySecondClass {
    private int firstNumber;
    private int secondNumber;

    public MySecondClass(int first, int second) {
        firstNumber = first;
        secondNumber = second;
    }

    public void setFirstNumber(int value) {
        firstNumber = value;
    }

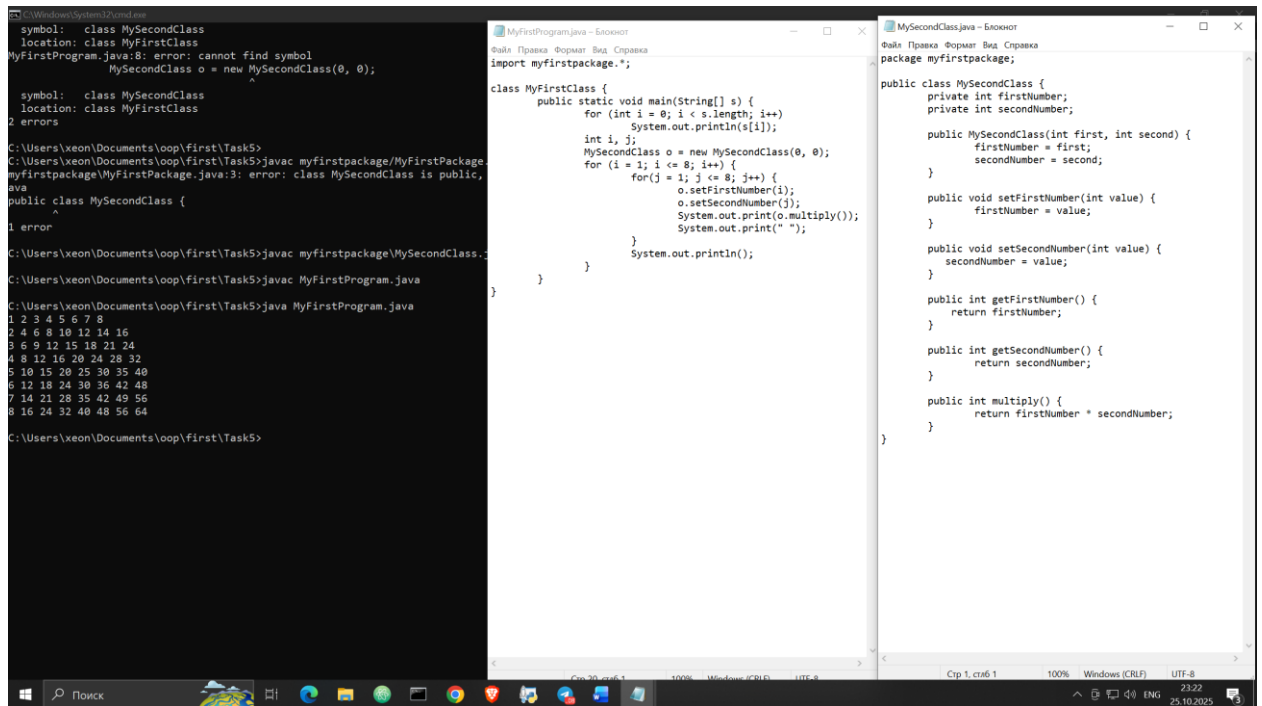
    public void setSecondNumber(int value) {
        secondNumber = value;
    }

    public int getFirstNumber() {
        return firstNumber;
    }

    public int getSecondNumber() {
        return secondNumber;
    }

    public int multiply() {
        return firstNumber * secondNumber;
    }
}
```

После переименования файла всё заработало



Задание 6

Была выполнена команда для запуска программы jar, так же был создан файл manifest.mf, в файл вставил шаблон из задания с указанием моей фамилии. Был создан myfirst.jar архив включающий полученные ранее файлы классов и созданный манифест-файл. Была создана директория MyJar и myfirst.jar был перенесён туда и успешно запущен.

