

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
УРАЛЬСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
имени первого Президента России Б. Н. Ельцина

Институт радиоэлектроники и информационных технологий-РТФ

Лабораторная работа №1
Тема: «Знакомство с языком Java»

Выполнил:
Нарсеев А.В.

Екатеринбург, 2025

Цель работы

Получить представление о написании программ на языке программирования Java с использованием командной строки, интегрированной среды разработки, а также решение задач на сайте timus

Ход выполнения

1.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk-  
Hello world!
```

2.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk-  
num: 100  
num * 2 = 200
```

3.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk-  
S < D  
S = D  
S > D
```

4.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openj  
Input a number:  
123  
Your number 123
```

5.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk-  
Input name:  
Ivan  
Input age:  
11  
Input height:  
123  
Name: Ivan  
Age: 11  
Height: 123.0
```

6.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk  
Введите фамилию:  
Familia  
Введите имя:  
Imea  
Введите отчество:  
Otchestvo  
Hello Familia, Imea, Otchestvo
```

7.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjd  
Введите имя:  
Imea  
Введите возраст:  
11  
Имя: Imea, возраст: 11
```

8.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk
```

Введите название текущего дня недели:

Пятница

Введите название месяца:

Февраль

Введите дату (номер дня в месяце):

20

Сегодня: Пятница, 20, Февраль

9.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk
```

Введите название месяца:

Февраль

Введите количество дней в этом месяце:

28

Февраль содержит 28 дней.

10.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk
```

Введите год рождения:

1981

Ваш возраст: 45

11.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk
```

Введите имя:

Имеа

Введите год рождения:

1981

Имеа, ваш возраст: 45

12.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk
Введите возраст:
45
Год рождения: 1981
```

13.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk
Введите первое число:
10
Введите второе число:
20
Сумма: 30
```

14.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk
Введите число:
10
9, 10, 11, 900
```

15.

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk
Введите первое число:
10
Введите второе число:
5
Сумма: 15
Разность: 5
```

Ответы на вопросы

1.

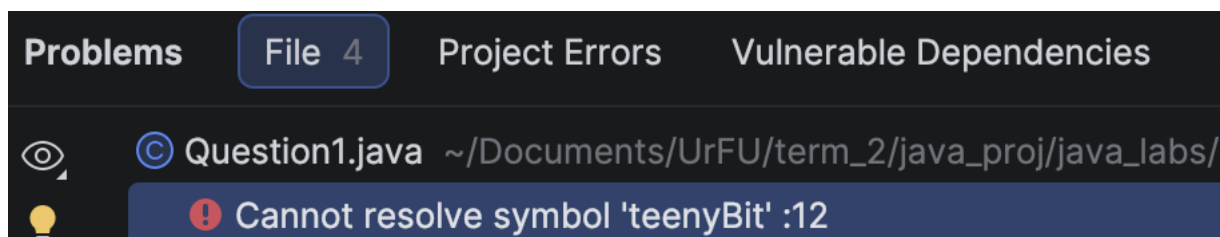
Будет ли исполняться программа без ошибок. Если без ошибок, то почему все работает. Если есть ошибка, то указать:

```
class SomeScopeExample {  
    public void eatIfHungry(boolean hungry) {  
        if (hungry) {  
            int bitesOfCheese = 1;  
            {  
                var teenyBit = true;  
                System.out.println(bitesOfCheese);  
            }  
        }  
        System.out.println(teenyBit);  
    }  
}
```

Ответ:

Нет, будет ошибка компиляции

Переменная `teenyBit` объявлена внутри внутреннего блока `{ var teenyBit = true; ... }` и имеет область видимости только внутри этого блока. Строка `System.out.println(teenyBit);` находится вне этого блока, поэтому компилятор выдаст ошибку



2.

Даны два куска кода. Определить какой из кусков будет работать без ошибок, а какой даст ошибку. В куске, который дает ошибку, объяснить почему:

```
short bird = 'd';  
char mammal = (short)83;  
char mammal = 83;
```

```
short reptile = 65535;  
char fish = (short)-1;  
char fish = -1;
```

Ответ:

Второй вернет ошибку

short reptile = 65535; // Ошибка: 65535 > Short.MAX_VALUE (32767), не помещается в short

char fish = -1; // Ошибка: литерал -1 имеет тип int; неявное приведение int к char для отрицательных значений не допускается

3.

Почему код ниже не будет компилироваться и как это исправить?

```
long max = 3123456789;
```

Ответ:

Не компилируется, потому что целочисленные литералы по умолчанию имеют тип int.

Число 3123456789 больше Integer.MAX_VALUE (2147483647)

Нужно указать, что литерал типа long

```
long max = 3123456789L;
```

4.

Что из нижеперечисленного будет компилироваться, а что нет. И почему:

```
int value = null;
```

```
List<Integer> heights = new ArrayList<>();  
heights.add(null);
```

```
String s = null;
```

Ответ:

```
int value = null;
```

Не компилируется

int нельзя присвоить null. Только ссылочные типы могут быть null

```
List<Integer> heights = new ArrayList<>();
```

```
heights.add(null);
```

Компилируется и выполняется

ArrayList допускает хранение null среди элементов

```
String s = null;
```

Компилируется

String ссылочный тип, переменная может ссылаться на null

5.

Что из нижеперечисленного **верно**? (Выберите все, что применимо)

A. Классовая переменная типа "boolean" имеет значение по умолчанию 0

B. Переменная класса типа boolean по умолчанию имеет значение false. C

Переменная класса типа boolean по дефолту равна null.

D. Переменная класса типа long по умолчанию имеет значение null

- E. Переменная класса типа long по умолчанию имеет значение 0L
- F. Классовая переменная типа long имеет значение по умолчанию 0
- G. Ничего из вышеперечисленного

Ответ:

B, E, F

6.

Что из нижеперечисленного **верно**? (Выберите все, что применимо).

- A. Локальная переменная типа boolean по умолчанию равна null.
- B. Локальная переменная типа float по умолчанию имеет значение 0.0f.
- C. Локальная переменная типа double по умолчанию равна 0.
- D. Локальная переменная типа Object имеет значение по умолчанию null.
- E. Локальная переменная типа boolean по умолчанию имеет значение false.
- F. Локальная переменная типа float по умолчанию равна 0.0.
- G. Ничего из вышеперечисленного

Ответ:

G

7.

Что из нижеперечисленного **верно**? (Выберите все, что применимо).

- A. Переменная экземпляра типа float по умолчанию равна 0.
- B. Переменная экземпляра типа char по умолчанию равна null.
- C. Переменная экземпляра типа double имеет значение по умолчанию 0.0.
- D. Переменная экземпляра типа int по умолчанию равна null.
- E. Переменная экземпляра типа String имеет значение по умолчанию null.
- F. Переменная экземпляра типа String по умолчанию принимает значение пустой строки "".
- G. Ничего из вышеперечисленного

Ответ:

A, C, E

8.

Какие из следующих утверждений о фрагменте кода являются **верны**? (Выберите все, что применимо.)

```
1. short numPets = 5L;  
2. int numGrains = 2.0;  
3. String name = "Scruffy";  
4. int d = numPets.length();  
5. int e = numGrains.length;  
6. int f = name.length();
```

1. Строка 1 генерирует ошибку компиляции.
2. Строка 2 генерирует ошибку компиляции.
3. Строка 3 генерирует ошибку компиляции.
4. Строка 4 генерирует ошибку компиляции.
5. Строка 5 генерирует ошибку компиляции.
6. Строка 6 генерирует ошибку компиляции.

Ответ:

1. ошибка компиляции (long в short без приведения)
2. ошибка компиляции (double в int без приведения)
3. компилируется
4. ошибка компиляции (у short нет метода length())
5. ошибка компиляции (у int нет поля length).
6. компилируется

9.

Какой из следующих наименований переменных **корректный**?

A. _

B. _helloWorld\$

C. true

D. java.lang

E. Public

F. 1980_s

G. _Q2_

Ответ:

A, B, E, G

10.

Написать в строках с System.out.println() что выведется в консоль: true или false

```
String firstString = "new";  
String secondString = "new";  
  
System.out.println(firstString == secondString);  
System.out.println(firstString.equals(secondString));  
  
String thirdString = new String("new");  
  
System.out.println(thirdString.equals(secondString));  
System.out.println(thirdString == secondString);  
System.out.println(thirdString.intern() == firstString);  
System.out.println(thirdString.equals(secondString));
```

Ответ:

```
/Users/narseev/Library/Java/JavaVirtualMachines/openjdk
true
true
true
false
true
true
```

11.

В следующем задании:

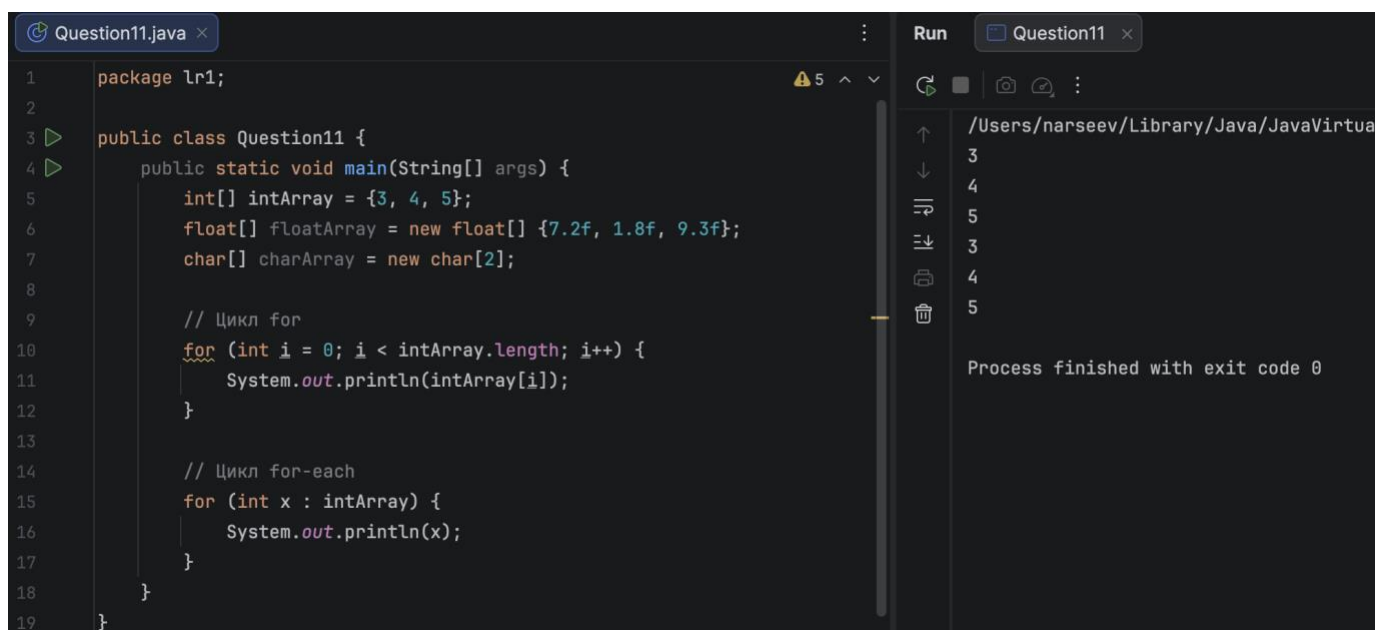
1. Определить есть ли ошибки в коде. Если есть, то где и почему. Исправить при наличии
2. Написать цикл, который будет итерироваться по массиву (по поводу цикла: **for** и **forEach**)

```
int[] intArray = {3, 4, 5};
float[] floatArray = float[] {7.2f, 1.8f, 9.3};
char[] charArray = new char[2];
```

Ответ:

Ошибки:

`float[] floatArray = float[] {7.2f, 1.8f, 9.3};` //неверный синтаксис. Должно быть либо `{7.2f, 1.8f, 9.3f}`, либо `new float[] { ... }` и `9.3` должно быть `9.3f` (литерал типа `float`).



The screenshot shows an IDE with a Java file named `Question11.java`. The code defines a class `Question11` with a `main` method. It initializes three arrays: `intArray` with values `{3, 4, 5}`, `floatArray` with values `{7.2f, 1.8f, 9.3f}`, and `charArray` with size `2`. It then prints the elements of `intArray` using a `for` loop and the elements of `intArray` using a `for-each` loop. The output window on the right shows the execution results: `3`, `4`, `5`, `3`, `4`, `5`. The process finished with exit code `0`.

```
package lr1;

public class Question11 {
    public static void main(String[] args) {
        int[] intArray = {3, 4, 5};
        float[] floatArray = new float[] {7.2f, 1.8f, 9.3f};
        char[] charArray = new char[2];

        // Цикл for
        for (int i = 0; i < intArray.length; i++) {
            System.out.println(intArray[i]);
        }

        // Цикл for-each
        for (int x : intArray) {
            System.out.println(x);
        }
    }
}
```

Run Question11

/Users/narseev/Library/Java/JavaVirtual

3
4
5
3
4
5

Process finished with exit code 0

12.

(Доп задание) Будет ли объект **one** доступен для удаления с помощью GC в результате исполнения программы

```
public class Scope {  
    public static void main(String[] args) {  
        String one, two;  
        one = new String("a");  
        two = new String("b");  
        one = two;  
        String three = one;  
        one = null;  
    }  
}
```

Ответ:

Да, так как имеется в виду объект String "a", на который изначально ссылалась переменная one

После one = two переменная one ссылается на тот же объект, что и two

Переменная three ссылается на тот же объект (three = one)

После one = null переменная one перестаёт ссылаться на что-либо

На строку "a" больше нет ссылок, поэтому строка "a" становится доступной для сборщика мусора и может быть удалена

GitHub

Ссылка: https://github.com/gh-u14/java_labs