

Задания из пятой Главы учебника

Упражнение 1. Цель: Написать программу на языке Java, которая рассчитывает длину окружности и площадь круга на основе заданного радиуса. Программа должна принимать радиус в виде аргумента командной строки и выводить результаты на экран.

Pi.java

```
public class Pi {
    public static void main(String[] args) {
        if (args.length == 0) {
            System.out.println("Укажите радиус в аргументах командной строки.");
            return; // Завершаем выполнение программы, если не указан радиус.
        }

        float radius = Float.parseFloat(args[0]); // Преобразуем строку аргумента в число
        с плавающей точкой.
        float shortPi = (float) Math.PI; // Приводим число Пи к типу float.
        float circ = 2 * shortPi * radius; // Для длины окружности используется формула 2
        * π * r
        float area = shortPi * radius * radius; // Вычисляем площадь круга.

        System.out.println("Если число Пи рассчитано в диапазоне от " + Math.PI + " до " +
        shortPi + "...");
        System.out.println("Окружность с радиусом " + radius + " см");
        System.out.println("имеет длину " + circ + " см и площадь " + area + " кв.см");
    }
}
```

Результат

```
PS C:\Users\abzal\Documents\q-java\6.1> javac Pi.java
PS C:\Users\abzal\Documents\q-java\6.1> java Pi 5
Если число Пи рассчитано в диапазоне от 3.141592653589793 до 3.1415927...
Окружность с радиусом 5.0 см
имеет длину 31.415928 см и площадь 78.53982 кв.см
```

Упражнение 2. Цель: Создание программы для вычисления степени числа и его квадратного корня.

Array.java

```
public class Power {
    public static void main(String[] args) {
        if (args.length == 0) {
            System.out.println("Укажите число в аргументах командной строки.");
            return; // Завершаем выполнение программы, если не указано число.
        }

        int num = Integer.parseInt(args[0]);
        int square = (int) Math.pow(num, 2);
        int cube = (int) Math.pow(num, 3);
        double sqrt = Math.sqrt(num); // Для квадратного корня используем тип double.

        System.out.println(num + " в квадрате равно " + square);
        System.out.println(num + " в кубе равно " + cube);
        System.out.println("Квадратный корень из " + num + " равен " + sqrt);
    }
}
```

Результат

```
PS C:\Users\abzal\Documents\q-java\6.1> javac Power.java
```

```
PS C:\Users\abzal\Documents\q-java\6.1> java Power 9
```

```
9 в квадрате равно 81
```

```
9 в кубе равно 729
```

```
Квадратный корень из 9 равен 3.0
```

Упражнение 3. Цель: Создание программы, которая выполняет округление числа в разных направлениях.

Round.java

```
public class Round {
    public static void main ( String[] args ) {
        float num = 7.25f ;
        System.out.println( num+" округленное равно "+Math.round( num ) );
        System.out.println( num+" округленное вниз равно " +Math.floor( num ) );
        System.out.println( num+" округленное вверх равно " + Math.ceil( num ) ) ;
    }
}
```

Результат

```
7.25 округленное равно 7
```

```
7.25 округленное вниз равно 7.0
```

```
7.25 округленное вверх равно 8.0
```

Упражнение 4. Цель: Создание программы, которая сравнивает два числа и выводит наибольшее и наименьшее из них.

Compare.java

```
public class Compare {
    public static void main ( String[] args ) {
        float num1 = 24.75f ;
        int num2 = 25 ;
        System.out.println( "Наибольшее: " + Math.max( num1, num2 ) ) ;
        System.out.println( "Наименьшее: " + Math.min( num1, num2 ) ) ;
    }
}
```

Результат

```
Наибольшее: 25.0
```

```
Наименьшее: 24.75
```

Упражнение 5. Цель: Создание программы, которая генерирует случайное число, умножает его на 10, и округляет результат до ближайшего целого числа. Программа должна выводить сгенерированное случайное число, результат умножения на 10 и округленное случайное целое число.

Random.java

```
public class Random {
    public static void main ( String[] args ) {
        float random = (float) Math.random() ;
        System.out.println( "Случайное число: " + random ) ;
        float multiplied = random * 10 ;
        System.out.println( "Умноженное на 10: " + multiplied ) ;
        int randomInt = (int) Math.ceil( multiplied ) ;
        System.out.println( "Случайное целое: " + randomInt ) ;
    }
}
```

```
}  
}
```

Результат

Случайное число: 0.9145456
Умноженное на 10: 9.145456
Случайное целое: 10

Упражнение 6. Цель: Создание программы, которая генерирует случайные числа для лотерейных билетов.

Lottery.java

```
public class Lottery {  
    public static void main ( String[] args ) {  
        int[] nums = new int[50] ;  
        for( int i = 1 ; i < 50 ; i++ ) { nums[i] = i ; }  
        for( int i = 1 ; i < 50 ; i++ )  
        {  
            int r = (int) Math.ceil( Math.random() * 49 ) ;  
            int temp = nums[i] ;  
            nums[i] = nums[r] ;  
            nums[r] = temp ;  
        }  
        for ( int i = 1 ; i < 7 ; i++ )  
        {  
            System.out.print( Integer.toString( nums[i]) + " " );  
        }  
    }  
}
```

Результат

```
PS C:\Users\abzal\Documents\q-java\6.1> javac Lottery.java  
PS C:\Users\abzal\Documents\q-java\6.1> java Lottery  
7 38 32 37 31 15  
PS C:\Users\abzal\Documents\q-java\6.1> java Lottery  
30 42 37 27 8 31  
PS C:\Users\abzal\Documents\q-java\6.1> java Lottery  
24 12 27 28 45 30
```

Упражнение 7. Цель: Создание программы, которая объединяет две строки и определяет их общую длину.

StringLength.java

```
public class StringLength {  
    public static void main ( String[] args) {  
        String lang = "Java" ;  
        String series = " in easy steps" ;  
        String title = lang.concat( series ) ;  
        System.out.print( "\"" + title + "\" содержит " ) ;  
        System.out.println( title.length() + " символов" ) ;  
    }  
}
```

Результат

"Java in easy steps" содержит 18 символов

Упражнение 8. Цель: Создание программы, которая выполняет проверку введенного пароля.

StringComparison.java

```
public class StringComparison {
    public static void main(String[] args) {
        if (args.length == 0) {
            System.out.println("Требуется ввод пароля.");
            return;
        }

        String password = "bingo";
        String userInput = args[0].toLowerCase();

        if (userInput.equals(password)) {
            System.out.println("Пароль подтвержден.");
        } else {
            System.out.println("Неверный пароль.");
        }
    }
}
```

* Код отличается от представленной в учебнике и был изменен чтобы избежать ошибки: `Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 0 out of bounds for length 0`
 at StringComparison.main(StringComparison.java:8)

Результат

```
PS C:\Users\abzal\Documents\q-java\6.1> javac StringComparison.java
PS C:\Users\abzal\Documents\q-java\6.1> java StringComparison BINGO
Пароль подтвержден.
PS C:\Users\abzal\Documents\q-java\6.1> java StringComparison bimbo
Неверный пароль.
PS C:\Users\abzal\Documents\q-java\6.1> java StringComparison
Требуется ввод пароля.
```

Упражнение 9. Цель: Создание программы для поиска и анализа строк в массиве.

StringSearch.java

```
public class StringSearch {
    public static void main(String[] args) {
        String[] books = { "Java in easy steps", "XML in easy steps",
            "HTML in easy steps", "CSS in easy steps",
            "Gone With the Wind", "Drop the Defense" };
        int counter1 = 0, counter2 = 0, counter3 = 0;
        for (int i = 0; i < books.length; i++) {
            System.out.print(books[i].substring(0, 4) + " | ");
            if (books[i].endsWith("in easy steps"))
                counter1++;
            if (books[i].startsWith("Java"))
                counter2++;
            if (books[i].indexOf("easy") == -1)
                counter3++;
        }
        System.out.println("\nНайдено " + counter1 + " названий из этой серии");
        System.out.println("Найдено " + counter2 + " названий с Java");
        System.out.println("Найдено " + counter3 + " других названий");
    }
}
```

}

Результат

Java | XML | HTML | CSS | Gone | Drop |

Найдено 4 названий из этой серии

Найдено 1 названий с Java

Найдено 2 других названий

Упражнение 10. Цель: Напишите программу на Java, которая начинает с пустой строки, затем заполняет ее значением "Боррокудо" при условии, что она пуста, выполняет обрезку пробелов и выводит длину, выводит первую и последнюю буквы строки, и заменяет все буквы "о" на "а" в строке.

CharacterSwap.java

```
public class CharacterSwap {
    public static void main(String[] args) {
        String txt = "";
        if (txt.isEmpty())
            txt = "        Боррокудо ";
        System.out.println("Строка: " + txt);
        System.out.println("Длина первоначальной строки: " + txt.length());
        txt = txt.trim();
        System.out.println("Строка: " + txt);
        System.out.println("Длина строки: " + txt.length());
        char initial = txt.charAt(0);
        System.out.println("Первая буква: " + initial);
        initial = txt.charAt((txt.length() - 1));
        System.out.println("Последняя буква: " + initial);
        txt = txt.replace('o', 'a');
        System.out.println("Строка: " + txt);
    }
}
```

Результат

```
Строка:          Боррокудо
Длина первоначальной строки: 19
Строка: Боррокудо
Длина строки: 9
Первая буква: Б
Последняя буква: о
Строка: Барракуда
```

Заключение

- В документации по языку Java можно найти информацию о методах и свойствах для каждого класса.
- Базовые классы языка Java содержатся в пакете **java.lang**.
- Класс **Math** предоставляет константы **Math.PI** и **Math.E**.
- Метод **Math.pow()** осуществляет возведение в указанную степень, а метод **Math.sqrt()** возвращает квадратный корень указанного числа.
- Числа можно округлять до целых значений, используя методы **Math.round()**, **Math.floor()** и **Math.ceil()**.
- При помощи методов **Math.max()** и **Math.min()** можно сравнивать числа.
- Метод **Math.random()** возвращает случайное число двойной точности в диапазоне от 0,0 до 0,9999999999999999.
- Строка представляет собой 0 или более символов, заключенных в знаки кавычек. • Подобно свойству массива **length** метод **length()** возвращает размер строки.
- Метод **concat()** используется для добавления одной строки к другой .
- Метод **equals()** возвращает значение **true** только в том случае, когда две строки содержат идентичные символы в одинаковом порядке.
- Регистр символа строки можно изменять, используя методы **toUpperCase()** и **toLowerCase()**.
- Используя методы **startsWith()** и **endsWith()**, можно сравнивать строки.
- При помощи методов **indexOf()** и **substring()** можно производить поиск подстроки в указанной строке.
- Метод **isEmpty()** возвращает значение **true** только в том случае, когда строка не содержит ни одного символа.
- Для работы с символами строки используются методы **trim()**, **charAt()** и **replace()**.