



Java

Ввод и вывод данных

Лекция #2 В

Пустовалова О.Г.
доцент. каф. мат.мод.
ИММИКН ЮФУ

Содержание

- Консольный ввод и вывод
- Класс System, объект out, метод println
- Класс System, объект out, метод printf
- Поток ввода System.in, объект Scanner
- Методы класса Scanner

Консольный ввод и вывод

Консольный вывод данных

Для взаимодействия с консолью в Java применяется класс **System**.

С его помощью можно выполнять консольный ввод и вывод.

Для создания потока вывода в класс System определен объект **out**.

В этом объекте определен метод **println**, который позволяет вывести на консоль некоторое значение.

В метод **println** передается любое значение, как правило, строка, которое надо вывести на консоль.

Консольный вывод данных

```
System.out.println("Hello world!");  
System.out.println("Bye world...");
```

```
int x=5;  
int y=6;  
System.out.println("x=" + x + "; y=" + y);;
```

Форматирование выводимых данных

```
double z = 10000.0 / 3.0;  
System.out.println(z);
```

3333.3333333333335

Строка формата

```
System.out.printf("%8.2f", z);
```

3333,33

Метод **printf ()** позволяет задавать произвольное количество параметров:

```
System.out.printf("Hello, %s. Next year, you'll be %d", name, age);
```

Форматирование выводимых данных

Для составления отформатированной символьной строки без последующего ее вывода можно вызвать статический метод `String.format ()`

```
int age = 21;
```

```
String name = "Tom";
```

```
String message =
```

```
    String.format("Hello, %s. Next year, you'll be %d",  
                name, age+1);
```

```
Hello, Tom. Next year, you'll be 22
```

Символы преобразования для метода printf ()

- d** – десятичное число 159
- x** – шестнадцатеричное целое 3f
- f** – число с фиксированной или плавающей точкой 1.59
- e** – число с плавающей точкой 1.59e+2
- a** – шестнадцатеричное представление числа с плавающей точкой
- s** – символьная строка
- c** – символ
- b** – логическое значение
- h** – хеш-код
- %** – знак процента
- n** – разделитель строк, зависящий от платформы

Флаги для метода printf ()

+ - выводит знак для отрицательных и для положительных чисел

пробел - добавляет пробел для положительных чисел

0 - выводит начальные нули

- - выравнивает поле по левому краю

(- заключает отрицательные числа в скобки

, - задает использование разделителя групп

- (для формата f) всегда отображает десятичную точку

- (для формата x или o) добавляет префикс 0x или 0

Флаги для метода printf ()

\$ – определяет индекс параметра, предназначенного для форматирования.

Например, выражение **%1\$d %1\$x** указывает на то, что первый параметр должен быть сначала выведен в **десятичной**, а затем в **шестнадцатеричной** форме **159 9F**

< – задает форматирование того же самого значения, которое отформатировано предыдущим спецификатором.

Например, выражение **%d %<x** указывает на то, что одно и то же значение должно быть представлено как в **десятичной**, так и в **шестнадцатеричной** форме **159 9F**

Пример

```
String name = "Tom";
```

```
int age = 30;
```

```
float height = 1.7f;
```

```
System.out.printf("Name: %s  
                  Age: %d  
                  Height: %.2f \n",  
                  name, age, height);
```

Чтение данных из консоли

```
import java.util.Scanner;

...
// создание объекта типа Scanner
// связывание этого объекта
// со стандартным потоком ввода System.in
Scanner in = new Scanner(System.in);

System.out.print("What is your name? ");

// чтение строки
String name = in.nextLine();

System.out.println(name);
```

Чтение данных из консоли

```
import java.util.Scanner;
...

// чтение строки
String name = in.nextLine();

// чтение одного слова
String firstName = in.next();

System.out.print("How old are you? ");

// чтение целого
int age = in.nextInt();
```

Ввод паролей

Класс `Scanner` не подходит для ввода паролей с консоли, поскольку такой ввод будет явно виден всякому желающему.

В версии Java SE 6 появился класс **`Console`**, специально предназначенный для этой цели.

Пример 1. Ввод данных

```
// класс Scanner находится в пакете java.util
import java.util.Scanner;

public class Main{

    public static void main(String[] args) {

        //Для создания самого объекта Scanner
        // в его конструктор передается объект System.in.
        Scanner in = new Scanner(System.in);
        System.out.print("Input a number: ");

        //Чтобы получить введенное число, используется метод in.nextInt(),
        // который возвращает введенное с клавиатуры цел. значение.
        int num = in.nextInt();

        System.out.printf("Your number: %d \n", num);
        in.close();
    }
}
```

Пример 2. Ввод данных

```
import java.util.Scanner;

public class Main{

    public static void main(String[] args) {

        Scanner in = new Scanner(System.in);

        System.out.print("Input name: ");
        String name = in.nextLine();

        System.out.print("Input age: ");
        int age = in.nextInt();

        System.out.print("Input height: ");
        float height = in.nextFloat();

        System.out.printf("Name: %s Age: %d Height: %.2f \n",
                           name, age, height);
        in.close();
    }
}
```


Ввод данных. Методы класса Scanner

next(): считывает введенную строку до первого пробела

nextLine(): считывает всю введенную строку

nextInt(): считывает введенное число **int**

nextDouble(): считывает введенное число **double**

nextBoolean(): считывает значение **boolean**

nextByte(): считывает введенное число **byte**

nextFloat(): считывает введенное число **float**

nextShort(): считывает введенное число **short**



Спасибо за внимание!