Standard I/O

Standard I/O

The class `System` provides basic machinery for Java's input/output (I/O).

- `System.out` is the standard output stream, and most often used for output.
- `System.in` is the standard input stream, but it is cumbersome to work with since it is just a sequence of characters—difficult to extract meaningful information.

Formatting output

`System.out.print()`

- `System.out.print()` is the simplest print method. It prints a string with a single (or no) argument.
- `System.out.println()` prints a string with a single (or no) argument, ending with a line break.

System.out.printf()

`System.out.printf()` prints a formatted string with additional arguments. In particular,

```
System.out.printf("...", <arguments>);
```

formats and inserts the arguments within the string and then print.

System.out.printf()

To format an integer, use `%d` in a string.

Example.

```
int age = 21;
System.out.printf("I am %d years old.", age);
```

Output.

```
I am 21 years old.
```

System.out.printf()

To format an integer, right-aligned for ***n*** characters, use %***n***d in a string.

Example.

```
int age = 21;
System.out.printf("I am %6d years old.", age);
```

Output.

```
I am     21 years old.
```

System.out.printf()

To format multiple integers, use `%d` for each integer in a string.

Example.

```
int a = 5;
int b = 10;
System.out.printf("I am %d ft. %d in. tall.",
                  a, b);
```

Output.

```
I am 5 ft. and 10 in. tall.
```

System.out.printf()

To format a real number, use `%.f` in a string.

Example.

```
static final double PI = Math.PI;
System.out.printf("The value of pi is %f.", PI);
```

Output.

```
The value of pi is 3.141592653589793.
```

System.out.printf()

To format a real number, rounded to ***m*** decimal places, use `%.`***m***`f` in a string.

Example.

```
static final double PI = Math.PI;
System.out.printf("The value of pi is
                   approximately %.2f.", PI);
```

Output.

```
The value of pi is approximately 3.14.
```

System.out.printf()

To format a real number, right-aligned for *n* characters, rounded to *m* decimal places, use %*n*.*m*f in a string.

Example.

```
static final double PI = Math.PI;
System.out.printf("The value of pi is
                   approximately %6.2f.", PI);
```

Output.

```
The value of pi is approximately   3.14.
```

Scanner

The class `Scanner` facilitates extracting meaningful information from `System.in`. It extracts information separated by *blank spaces*.

- `next()` — extracts the next word as a single `String` object.
- `nextInt()` — extracts the next integer as an `int` value.
- `nextDouble()` — extracts the next real number as a `double` value.

Scanner

Example.

```
Scanner s = new Scanner(System.in);
System.out.print(”Enter an integer: ”);
int i = s.nextInt();
System.out.print(”You entered: ” + i);
```

Scanner

Example.

```
Scanner s = new Scanner(System.in);
System.out.print(”Enter a real number: ”);
double d = s.nextDouble();
System.out.print(”You entered: ” + d);
```

Scanner

Example.

```
Scanner s = new Scanner(System.in);
System.out.print(”Enter your name: ”);
String w = s.next();
System.out.print(”You entered: ” + w);
```