Chapter 3:

Formal Parameter: A variable that appears inside parentheses in the header of a method that is used to generalize the method's behavior

Actual Parameter: A specific value or expression that appears inside parentheses in a method call

Method Signature: The name of a method, along with its number and type of parameters

Method Overloading: The ability to define two or more different methods with the same name but different method signatures

Return: To send a value out as the result of a method that can be used in an expression in your program

Useful Static Methods in the Math Class

Method	Description	Example
`abs`	absolute value	`Math.abs(-308)` returns `308`
`ceil`	ceiling (rounds upward)	`Math.ceil(2.13)` returns `3.0`
`cos`	cosine (radians)	`Math.cos(Math.PI)` returns `-1.0`
`exp`	exponent base *e*	`Math.exp(1)` returns `2.7182818284590455`
`floor`	floor (rounds downward)	`Math.floor(2.93)` returns `2.0`
`log`	logarithm base *e*	`Math.log(Math.E)` returns `1.0`
`log10`	logarithm base 10	`Math.log10(1000)` returns `3.0`
`max`	maximum of two values	`Math.max(45, 207)` returns `207`
`min`	minimum of two values	`Math.min(3.8, 2.75)` returns `2.75`
`pow`	power (general exponentiation)	`Math.pow(3, 4)` returns `81.0`
`random`	random value	`Math.random()` returns a random `double` value *k* such that $0.0 \le k < 1.0$
`round`	round real number to nearest integer	`Math.round(2.718)` returns `3`
`sin`	sine (radians)	`Math.sin(0)` returns `0.0`
`sqrt`	square root	`Math.sqrt(2)` returns `1.4142135623730951`
`toDegrees`	converts from radians to degrees	`Math.toDegrees(Math.PI)` returns `180.0`
`toRadians`	converts from degrees to radians	`Math.toRadians(270.0)` returns `4.71238898038469`

Object: A programming entity that contains state (data) and behavior (methods)

Class: A category or type of object

String Objects: String objects are one of the most useful and most commonly used types of objects in Java

There are a lot of special rules that apply only to strings:

- One special property of String objects is that there are literals that represent them
- You can declare variables of type String and use the assignment statement to give values to these variables
    - String s = "hello there";
- The type String is capitalized (as are the names of all object types in Java), unlike the primitive types such as double and int

Useful Methods of String Objects

Method	Description	Example (assuming s is "hello")
`charAt(index)`	character at a specific index	`s.charAt(1)` returns `'e'`
`endsWith(text)`	whether or not the string ends with some text	`s.endsWith("llo")` returns `true`
`indexOf(text)`	index of a particular character or `String` (–1 if not present)	`s.indexOf("o")` returns `4`
`length()`	number of characters in the string	`s.length()` returns `5`
`replace(s1, s2)`	replace all occurrences of one substring with another	`s.replace("l", "y")` returns `"heyyyyo"`
`startsWith(text)`	whether or not the string starts with some text	`s.startsWith("hi")` returns `false`
`substring(start, stop)`	characters from start index to just before stop index	`s.substring(1, 3)` returns `"el"`
`toLowerCase()`	a new string with all lowercase letters	`s.toLowerCase()` returns `"hello"`
`toUpperCase()`	a new string with all uppercase letters	`s.toUpperCase()` returns `"HELLO"`

Index: An integer used to specify a location in a sequence of values

Exception: A runtime error that prevents a program from continuing its normal execution

Immutable Object: An object whose value cannot be changed

Console Input: Responses typed by the user when an interactive program pauses for input

Constructor (Construct): A method that creates and initializes an object. Objects in Java programs must be constructed before they can be used

Token: A single element of input (e.g., one word, one number).

Whitespace Spaces: tab characters, and new line characters.

Package: A collection of related Java classes.

Import Declaration: A request to access a specific Java package.

(Page 297).