Math Functions, Characters and Strings

COP2250: Java Programming

Kevin Pyatt, Ph.D.

State College of Florida
Pyatt Labs

2026

Today's Objectives

- Use common Math class methods
- Understand the `char` data type
- Compare and test characters
- Work with String methods
- Convert between characters and strings

The Math Class

Java's Math class provides common mathematical functions.

```java
// No import needed - java.lang.Math is automatic

double x = Math.pow(2, 3);   // 8.0 (2^3)
double y = Math.sqrt(16);    // 4.0
double z = Math.abs(-5.5);   // 5.5
int max = Math.max(10, 20);  // 20
int min = Math.min(10, 20);  // 10
double r = Math.random();    // 0.0 to 0.999...
```

All methods are static — call with `Math.methodName()`

More Math Methods

Method	Description	Example
`Math.pow(a, b)`	a^b	`Math.pow(2, 3)` → 8.0
`Math.sqrt(x)`	$\sqrt{x}$	`Math.sqrt(25)` → 5.0
`Math.abs(x)`	$\|x\|$	`Math.abs(-7)` → 7
`Math.max(a, b)`	larger of a, b	`Math.max(3, 9)` → 9
`Math.min(a, b)`	smaller of a, b	`Math.min(3, 9)` → 3
`Math.round(x)`	nearest integer	`Math.round(3.7)` → 4
`Math.ceil(x)`	round up	`Math.ceil(3.1)` → 4.0
`Math.floor(x)`	round down	`Math.floor(3.9)` → 3.0

The char Data Type

A `char` holds a single character.

```
char letter = ’A’;
char digit = ’5’;
char symbol = ’@’;
char space = ’ ’;
```

Key points:

- Use single quotes: `’A’` not `"A"`
- Each char has a numeric Unicode value
- `’A’` = 65, `’a’` = 97, `’0’` = 48

Characters are Numbers

You can do math with characters!

```
char letter = ’A’;
System.out.println((int) letter); // 65

char next = (char) (letter + 1);
System.out.println(next);       // B

// Check if uppercase: A=65, Z=90
char ch = ’M’;
if (ch >= ’A’ && ch <= ’Z’) {
    System.out.println("Uppercase!");
}
```

ASCII/Unicode ranges:

- ’A’ to ’Z’: 65–90
- ’a’ to ’z’: 97–122
- ’0’ to ’9’: 48–57

The Character Class

Helper methods for testing characters:

```
char ch = 'A';

Character.isLetter(ch)    // true
Character.isDigit(ch)     // false
Character.isUpperCase(ch) // true
Character.isLowerCase(ch) // false
Character.toUpperCase('a') // 'A'
Character.toLowerCase('A') // 'a'
```

These are perfect for validating input!

Reading a Character from Input

Scanner doesn't have `nextChar()` — use this pattern:

```
Scanner input = new Scanner(System.in);

System.out.print("Enter a letter: ");
String s = input.nextLine();
char ch = s.charAt(0); // Get first character

System.out.println("You entered: " + ch);
```

Pattern:

1. Read the whole line as a String
2. Extract first character with `.charAt(0)`

Example: Check for Vowel

```
char ch = ’E’;
char lower = Character.toLowerCase(ch);

if (lower == ’a’ || lower == ’e’ || lower == ’i’
    || lower == ’o’ || lower == ’u’) {
    System.out.println(ch + " is a vowel");
} else {
    System.out.println(ch + " is a consonant");
}
```

Why convert to lowercase first?

- Only need to check 5 vowels instead of 10
- Works for both 'A' and 'a'

String Basics

A String is a sequence of characters.

```
String greeting = "Hello";
String name = "World";
String message = greeting + " " + name; // Concatenation

System.out.println(message);     // Hello World
System.out.println(message.length()); // 11
```

Key points:

- Use double quotes: `"Hello"` not `’Hello’`
- Strings are **objects**, not primitives
- `+` concatenates strings

Common String Methods

```
String s = "Hello World";

s.length()          // 11
s.charAt(0)         // 'H'
s.charAt(6)         // 'W'
s.toUpperCase()     // "HELLO WORLD"
s.toLowerCase()     // "hello world"
s.substring(0, 5)   // "Hello"
s.substring(6)      // "World"
s.indexOf("o")      // 4 (first occurrence)
s.contains("World") // true
```

Comparing Strings

NEVER use == to compare strings!

```
String s1 = "Hello";
String s2 = "Hello";

// WRONG - compares memory addresses
if (s1 == s2) { ... }

// CORRECT - compares actual content
if (s1.equals(s2)) { ... }

// Case-insensitive comparison
if (s1.equalsIgnoreCase("HELLO")) { ... }
```

Rule: Always use `.equals()` for strings!

char vs String

	char	**String**
Quotes	Single: 'A'	Double: "A"
Type	Primitive	Object
Length	Always 1	0 or more
Compare	== works	Use .equals()

```
char c = 'A';            // Single character
String s = "A";          // String with one character
String t = "ABC";        // String with three characters

char first = t.charAt(0); // Extract char from String
```

Summary

- **Math class:** `pow`, `sqrt`, `abs`, `max`, `min`, `random`
- **char:** Single character in single quotes
- **Character class:** `isLetter`, `isDigit`, `isUpperCase`, `toLowerCase`
- **Reading char:** `input.nextLine().charAt(0)`
- **String:** Sequence of characters in double quotes
- **String methods:** `length`, `charAt`, `substring`, `equals`
- **NEVER** use == to compare strings!

Lab 3: Character Practice

Complete `CharacterPractice.java`:

1. Read a character from user
2. Check if it's a letter using `Character.isLetter()`
3. Check if it's uppercase or lowercase
4. Convert case and display

This prepares you for Assignment 4!

Assignment 4: Vowel or Consonant

Build a program that:

1. Prompts user to enter a letter
2. Checks if it’s a vowel (a, e, i, o, u)
3. Checks if it’s a consonant
4. Displays “invalid input” for non-letters

Starter code and reference sheet in the repo!

Questions?

Lab 3: `CharacterPractice.java`

Assignment 4: Vowel or Consonant