# **Week 1: Core Programming**

Welcome to the repository for Week 1 of our Java learning journey. This week focused on building a strong foundation in the fundamental concepts of Java programming.

## **Table of Contents**

* Overview
* Topics Covered
  + Day 1: Programming Elements
  + Day 2: Control Flows
  + Day 3: Arrays
  + Day 4: Methods
  + Day 5: Strings
* Prerequisites
* How to Use This Repository

## **Overview**

The first week was dedicated to understanding the core syntax and essential building blocks of the Java programming language. We covered how to write simple Java programs, handle different types of data, control the flow of execution, and work with fundamental data structures like arrays and strings, as well as modularizing code with methods.

## **Topics Covered**

### **Day 1: Programming Elements**

* Comments
* Basic Data types
* Variables
* Variable naming
* Arithmetic operator
* Operator Precedence
* Type conversion
* User input

### **Day 2: Control Flows**

* Control flows
* Boolean Data types
* Boolean Expressions
* Logical operators
* If, else, else if statements
* Loops(for & while)
* Break & Continue statements
* Switch statements

### **Day 3: Arrays**

* What is Array?
* Create an Array
* Access an array
* Accessing array using loop
* Modifying an array
* Length of an array
* Two Dimensional Array
* Accessing, finding size of Multidimensional array
* System error() message & exit() method

### **Day 4: Methods**

* Introduction to Methods
* User defined methods
* Using methods multiple times
* Dynamic methods
* Recursive methods
* Static methods
* Standard Library methods
* Java Math class(java.lang.Math)

### **Day 5: Strings**

* Introduction to Strings
* Creating Strings
* Use of “\” Escape sequence
* Take String input
* String Arrays
* String as method parameters
* String Class Built-in Methods
* ASCII Character Code
* Exceptions
* Exceptions Hierarchy
* Types of Exception
* Unchecked Exceptions
* Checked Exceptions

## **Prerequisites**

* A computer with a supported operating system (Windows, macOS, Linux).
* Java Development Kit (JDK) installed.
* A code editor or Integrated Development Environment (IDE) like VS Code, IntelliJ IDEA, or Eclipse.

## **How to Use This Repository**

This repository is structured to reflect the topics covered each day. You can find code examples, notes, or exercises within the respective daily folders (if available). Feel free to explore the code, run it, and modify it to deepen your understanding.