**Why Convert Integers to Strings for Octal and Hexadecimal Representations?**

In Java, we typically convert integers to strings for octal and hexadecimal representations because:

1. **Direct Integer Representation Limitations:**
   1. Integer data types in Java are stored in binary format.
   2. While we can perform arithmetic operations directly on these binary representations, they are not directly human-readable in octal or hexadecimal forms.
   3. To display or process these numbers in octal or hexadecimal, we need to convert them to a textual format.
2. **String Manipulation and Formatting:**
   1. Strings provide flexibility for various manipulations, including:
      * Formatting: Adding prefixes (like "0x" for hexadecimal), suffixes, or delimiters.
      * Concatenation: Combining with other strings for output or further processing.
      * Parsing: Converting back to numerical values if needed.
      * Regular expressions: Applying pattern matching and extraction.
3. **Compatibility with I/O Operations:**
   1. Most I/O operations, such as printing to the console or writing to files, work with text-based data.
   2. Converting numbers to strings allows us to integrate them seamlessly with text-based output and input.