

Number Conversion

You said:



- 1] Convert Binary to Decimal
- 2] Convert Binary to Octal
- 3] Convert Decimal to Binary Number
- 4] Convert Decimal to Octal
- 5] Convert Octal to Binary
- 6] Convert Octal to Decimal
- 7] Convert digits/numbers to words

give code of each in java from scratch and also give another optimized code also for individual and also give little explanation with example

here i want one code that do not use any inbuilt function it will do pure mathematical computations and another will use in build method also give mathematical way how this conversion happen

ChatGPT said:



Memory updated

Here are the Java programs for each conversion with both pure mathematical computation and optimized code using built-in methods, along with explanations of how the conversions work mathematically.

1. Convert Binary to Decimal

Pure Mathematical Computation

java

```
public class BinaryToDecimal {
    public static int binaryToDecimal(int binary) {
        int decimal = 0;
        int power = 0;

        while (binary > 0) {
            int lastDigit = binary % 10;
            decimal += lastDigit * Math.pow(2, power);
            binary = binary / 10;
            power++;
        }
        return decimal;
    }

    public static void main(String[] args) {
        int binary = 1010; // Example: Binary 1010
        System.out.println("Decimal: " + binaryToDecimal(binary));
    }
}
```

```
}
}
```

Explanation:

- Binary numbers use base 2, where each digit represents a power of 2.
- For binary `1010`, we calculate: $1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0 = 8 + 0 + 2 + 0 = 10$

Optimized Code (Using `Integer.parseInt()`)

```
java
```

```
public class BinaryToDecimalOptimized {
    public static void main(String[] args) {
        String binary = "1010"; // Example: Binary 1010
        int decimal = Integer.parseInt(binary, 2);
        System.out.println("Decimal: " + decimal);
    }
}
```

2. Convert Binary to Octal

Pure Mathematical Computation

```
java
```

```
public class BinaryToOctal {
    public static int binaryToDecimal(int binary) {
        int decimal = 0, power = 0;
        while (binary > 0) {
            int lastDigit = binary % 10;
            decimal += lastDigit * Math.pow(2, power);
            binary /= 10;
            power++;
        }
        return decimal;
    }

    public static int decimalToOctal(int decimal) {
        int octal = 0, power = 1;
        while (decimal > 0) {
            octal += (decimal % 8) * power;
            decimal /= 8;
            power *= 10;
        }
        return octal;
    }

    public static void main(String[] args) {
        int binary = 1010; // Example: Binary 1010
        int decimal = binaryToDecimal(binary);
        System.out.println("Octal: " + decimalToOctal(decimal));
    }
}
```

Optimized Code (Using `Integer.parseInt()` and `Integer.toOctalString()`)

```
java
```

```
public class BinaryToOctalOptimized {
    public static void main(String[] args) {
        String binary = "1010"; // Example: Binary 1010
        int decimal = Integer.parseInt(binary, 2);
        String octal = Integer.toOctalString(decimal);
        System.out.println("Octal: " + octal);
    }
}
```

3. Convert Decimal to Binary

Pure Mathematical Computation

```
java
```

```
public class DecimalToBinary {
    public static int decimalToBinary(int decimal) {
        int binary = 0, power = 1;
        while (decimal > 0) {
            binary += (decimal % 2) * power;
            decimal /= 2;
            power *= 10;
        }
        return binary;
    }

    public static void main(String[] args) {
        int decimal = 10; // Example: Decimal 10
        System.out.println("Binary: " + decimalToBinary(decimal));
    }
}
```

Optimized Code (Using `Integer.toBinaryString()`)

```
java
```

```
public class DecimalToBinaryOptimized {
    public static void main(String[] args) {
        int decimal = 10; // Example: Decimal 10
        String binary = Integer.toBinaryString(decimal);
        System.out.println("Binary: " + binary);
    }
}
```

4. Convert Decimal to Octal

Pure Mathematical Computation

```
java
```

```
public class DecimalToOctal {
    public static int decimalToOctal(int decimal) {
        int octal = 0, power = 1;

```

```

        while (decimal > 0) {
            octal += (decimal % 8) * power;
            decimal /= 8;
            power *= 10;
        }
        return octal;
    }

    public static void main(String[] args) {
        int decimal = 10; // Example: Decimal 10
        System.out.println("Octal: " + decimalToOctal(decimal));
    }
}

```

Optimized Code (Using `Integer.toOctalString()`)

java

```

public class DecimalToOctalOptimized {
    public static void main(String[] args) {
        int decimal = 10; // Example: Decimal 10
        String octal = Integer.toOctalString(decimal);
        System.out.println("Octal: " + octal);
    }
}

```

5. Convert Octal to Binary

Pure Mathematical Computation

java

```

public class OctalToBinary {
    public static int octalToDecimal(int octal) {
        int decimal = 0, power = 0;
        while (octal > 0) {
            decimal += (octal % 10) * Math.pow(8, power);
            octal /= 10;
            power++;
        }
        return decimal;
    }

    public static int decimalToBinary(int decimal) {
        int binary = 0, power = 1;
        while (decimal > 0) {
            binary += (decimal % 2) * power;
            decimal /= 2;
            power *= 10;
        }
        return binary;
    }

    public static void main(String[] args) {
        int octal = 12; // Example: Octal 12
        int decimal = octalToDecimal(octal);
        System.out.println("Binary: " + decimalToBinary(decimal));
    }
}

```

Optimized Code (Using `Integer.parseInt()` and `Integer.toBinaryString()`)

```
java
```

```
public class OctalToBinaryOptimized {
    public static void main(String[] args) {
        String octal = "12"; // Example: Octal 12
        int decimal = Integer.parseInt(octal, 8);
        String binary = Integer.toBinaryString(decimal);
        System.out.println("Binary: " + binary);
    }
}
```

6. Convert Octal to Decimal

Pure Mathematical Computation

```
java
```

```
public class OctalToDecimal {
    public static int octalToDecimal(int octal) {
        int decimal = 0, power = 0;
        while (octal > 0) {
            decimal += (octal % 10) * Math.pow(8, power);
            octal /= 10;
            power++;
        }
        return decimal;
    }

    public static void main(String[] args) {
        int octal = 12; // Example: Octal 12
        System.out.println("Decimal: " + octalToDecimal(octal));
    }
}
```

Optimized Code (Using `Integer.parseInt()`)

```
java
```

```
public class OctalToDecimalOptimized {
    public static void main(String[] args) {
        String octal = "12"; // Example: Octal 12
        int decimal = Integer.parseInt(octal, 8);
        System.out.println("Decimal: " + decimal);
    }
}
```

7. Convert Digits/Numbers to Words

Pure Mathematical Computation

```
java
```

```
public class NumberToWords {
    static String[] ones = { "Zero", "One", "Two", "Three", "Four", "Five", "Six",
        "Seven", "Eight", "Nine" };
}
```

```

public static void numberToWords(int number) {
    if (number == 0) {
        System.out.print("Zero");
        return;
    }
    printWords(number);
}

private static void printWords(int number) {
    if (number == 0) return;
    printWords(number / 10); // Recursion to handle digits
    System.out.print(ones[number % 10] + " ");
}

public static void main(String[] args) {
    int number = 123; // Example: Number 123
    numberToWords(number);
}
}

```

Optimized Code (Using Arrays for Words)

java

```

public class NumberToWordsOptimized {
    static String[] words = { "Zero", "One", "Two", "Three", "Four", "Five", "Six",
    "Seven", "Eight", "Nine" };

    public static void main(String[] args) {
        int number = 123; // Example: Number 123
        System.out.println(convertToWords(number));
    }

    public static String convertToWords(int number) {
        if (number == 0) return words[0];
        StringBuilder result = new StringBuilder();
        while (number > 0) {
            int digit = number % 10;
            result.insert(0, words[digit] + " ");
            number /= 10;
        }
        return result.toString().trim();
    }
}

```

This covers all the conversions you requested with both pure mathematical methods and optimized versions using built-in functions.

ChatGPT can make mistakes. Check important info.