

## Harshad number

```
import java.io.*;
import java.util.*;

public class Solution {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int originalNumber = n;
        int sumOfDigits = 0;

        while (n > 0) {
            sumOfDigits += n % 10;
            n /= 10;
        }

        if (originalNumber % sumOfDigits == 0) {
            System.out.println("Harshad Number");
        } else {
            System.out.println("Not Harshad Number");
        }
        sc.close();
    }
}
```

## Abundant number

```
import java.io.*;
import java.util.*;

public class Solution {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int n = scanner.nextInt();
        scanner.close();

        int sumOfProperDivisors = 0;
        for (int i = 1; i < n; i++) {
            if (n % i == 0) {
                sumOfProperDivisors += i;
            }
        }
    }
}
```

```

    }
}

if (sumOfProperDivisors > n) {
    System.out.println("Abundant Number");
} else {
    System.out.println("Not Abundant Number");
}
}
}

```

## Sum of digits 10

```

import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int n = scanner.nextInt();
        scanner.close();

        if (n < 100) { // Check if N is more than 2 digits (i.e., >= 100)
            System.out.println("Invalid Input");
        } else {
            int sum = 0;
            int tempN = n; // Use a temporary variable to avoid modifying the original 'n'
            while (tempN > 0) {
                int digit = tempN % 10;
                sum += digit;
                tempN /= 10;
            }
            System.out.println("Sum of digit is " + sum);
        }
    }
}

```

## Fibonacci series

```

import java.util.Scanner;

public class FibonacciSum {

```

```

// Method to calculate nth Fibonacci number
public static int fibonacci(int n) {
    if (n == 1) return 0; // 1st Fibonacci = 0
    if (n == 2) return 1; // 2nd Fibonacci = 1
    int a = 0, b = 1, c = 0;
    for (int i = 3; i <= n; i++) {
        c = a + b;
        a = b;
        b = c;
    }
    return b;
}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    // Reading two inputs
    int start = sc.nextInt();
    int end = sc.nextInt();

    // Check valid range
    if (start < 1 || end > 20 || start > end) {
        System.out.println("Invalid Input");
    } else {
        double sum = 0;
        for (int i = start; i <= end; i++) {
            sum += fibonacci(i);
        }
        System.out.println("The Sum of Fibonacci value is " + sum);
    }

    sc.close();
}
}

```

## Multiplication number 79

```

import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
    }
}

```

```

        if (n >= 1 && n <= 9) {
            for (int i = 1; i <= n; i++) {
                System.out.println(n + " x " + i + " = " + (n * i));
            }
        } else {
            System.out.println("Invalid Input");
        }
    }
}

```

## Sum of even number 1

```

import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int m = sc.nextInt();
        sc.close();

        if (n <= 0 || n >= 30 || m <= 0 || m >= 30 || n > m) {
            System.out.println("Invalid Input");
        } else {
            int sumOfEvens = 0;
            for (int i = n; i <= m; i++) {
                if (i % 2 == 0) {
                    sumOfEvens += i;
                }
            }
            System.out.println(sumOfEvens);
        }
    }
}

```

## Armstrong number or not 13

```

import java.util.Scanner;

public class ArmstrongCheck {
    public static void main(String[] args) {

```

```

Scanner sc = new Scanner(System.in);

// Read input
int num = sc.nextInt();
int original = num;
int digits = String.valueOf(num).length(); // Count digits

int sum = 0;
int temp = num;

// Calculate Armstrong sum
while (temp > 0) {
    int digit = temp % 10;
    sum += Math.pow(digit, digits);
    temp /= 10;
}

// Print result
if (sum == original) {
    System.out.println("Yes");
} else {
    System.out.println("No");
}

sc.close();
}
}

```

## Swap 2 digit number

```

import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String num = sc.next();

        if (num.length() < 2) {
            System.out.println("STDOUT");
        } else {
            // Take only the first two digits
            char first = num.charAt(0);
            char second = num.charAt(1);

```

```

        // Swap and print as number
        System.out.println("" + second + first);
    }
}

```

## Reverse a number using loop 1

```

import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        long n = sc.nextLong();

        // Check constraints
        if (n <= 0 || n > 1000000000) {
            System.out.println("Invalid Input");
        } else {
            long rev = 0;
            long temp = n;

            // Loop to reverse
            while (temp > 0) {
                long digit = temp % 10; // get last digit
                rev = rev * 10 + digit; // build reverse number
                temp = temp / 10;       // remove last digit
            }

            System.out.println(rev);
        }
    }
}

```

## Collatz sequence 7

```

import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter a number");
    }
}

```

```

long n = sc.nextLong();

int count = 0;

System.out.println(n); // print the first number

while (n != 1) {
    if (n % 2 == 0) {
        n = n / 2;
    } else {
        n = 3 * n + 1;
    }
    System.out.println(n);
    count++;
}

System.out.println("count:" + count);
}
}

```

## Count digits in an integer 1

```

import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        long num = sc.nextLong();

        // Check constraint: 1 <= num <= 10000000
        if (num >= 1 && num <= 10000000) {
            int count = 0;
            long temp = num;

            while (temp > 0) {
                temp /= 10;
                count++;
            }

            System.out.println("The count of the given integer is: " + count);
        } else {
            System.out.println("Enter a Valid Input");
        }
    }
}

```

```
}  
}
```

## Print a pattern 4

```
import java.util.Scanner;  
  
public class Main {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        int n = sc.nextInt();  
  
        if (n < 0 || n > 9) {  
            System.out.println("Invalid Input");  
        } else {  
            char ch = 'A';  
            for (int i = 1; i <= n; i++) {  
                for (int j = 1; j <= i; j++) {  
                    System.out.print(ch + " ");  
                    ch++;  
                }  
                System.out.println();  
            }  
        }  
    }  
}
```

## Alphabet diamond 1

```
import java.util.Scanner;  
  
public class Main {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        int n = sc.nextInt();  
  
        // Upper part (1 to n)  
        for (int i = 1; i <= n; i++) {  
            // Print leading spaces  
            for (int s = 1; s <= n - i; s++) {  
                System.out.print(" ");  
            }  
            // Print letters
```



```

        for (int j = 1; j <= i; j++) {
            System.out.print((char)('A' + j - 1) + " ");
        }
        System.out.println();
    }

    // Lower part (n-1 to 1)
    for (int i = n - 1; i >= 1; i--) {
        // Print leading spaces
        for (int s = 1; s <= n - i; s++) {
            System.out.print(" ");
        }
        // Print letters
        for (int j = 1; j <= i; j++) {
            System.out.print((char)('A' + j - 1) + " ");
        }
        System.out.println();
    }
}
}

```

## Hollow square pattern 4

```

import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();

        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= n; j++) {
                // Print * if border, else space
                if (i == 1 || i == n || j == 1 || j == n) {
                    System.out.print("*");
                } else {
                    System.out.print(" ");
                }
            }
            System.out.println();
        }
    }
}

```

# Alphabet right triangle

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();

        for (int i = 1; i <= n; i++) {
            char ch = (char) ('A' + i - 1); // starting character for row
            for (int j = 1; j <= i; j++) {
                System.out.print(ch + " ");
                ch++;
            }
            System.out.println();
        }
    }
}
```