1415102

3451

2261617

713188

9192012

1121



**1. Prime Number:**

package imp.level1;

public class PrimeExample{

public static void main(String args[]){

int i,m=0,flag=0;

int n=3;//it is the number to be checked

m=n/2;

if(n==0||n==1){

System.out.println(n+" is not prime number");

}else{

for(i=2;i<=m;i++){

if(n%i==0){

System.out.println(n+" is not prime number");

flag=1;

break;

}

}

if(flag==0) { System.out.println(n+" is prime number"); }

}//end of else

}

}

**2. Factorial:**

package imp.level1;

public class Factorial {

public static void main(String[] args) {

int num = 5, factorial = 1;

for (int i = 1; i <= num; ++i) {

factorial \*= i;

}

System.out.println("Factorial of " + num + " is: " + factorial);

}

}

**3. Fibonacci:**

package imp.level1;

public class Fibonacci {

public static void main(String[] args) {

//initializing the constants

int n = 100, t1 = 0, t2 = 1;

System.out.print("Upto " + n + ": ");

//while loop to calculate fibonacci series upto n numbers

while (t1 <= n) {

System.out.print(t1 + " + ");

int sum = t1 + t2;

t1 = t2;

t2 = sum;

}

}

public int fibonacci1(int number) {

if (number == 1 || number == 2) {

return 1;

}

return fibonacci1(number - 1) + fibonacci1(number - 2);

}

public static int fibonacci2(int number) {

if (number == 1 || number == 2) {

return 1;

}

int fibo1 = 1, fibo2 = 1, fibonacci = 1;

for (int i = 3; i <= number; i++) {

fibonacci = fibo1 + fibo2;

fibo1 = fibo2;

fibo2 = fibonacci;

}

return fibonacci;

}

}

**4. PowerOfNumber:**

package imp.level1;

import java.util.Scanner;

public class PowerOfNumber {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.print("Enter base: ");

int base = input.nextInt();

System.out.print("Enter exponent: ");

int exponent = input.nextInt();

long result = 1;

while (exponent != 0) {

result \*= base;

--exponent;

}

System.out.println("Result: " + result);

input.close();

}

}

**5. ReverseNumber :**

package imp.level1;

public class ReverseNumber

{

public static void main(String[] args)

{

int number = 987654, reverse = 0;

while(number != 0)

{

int remainder = number % 10;

reverse = reverse \* 10 + remainder;

number = number/10;

}

System.out.println("The reverse of the given number is: " + reverse);

}

}

**6. Palindrome Integer:**

package imp.level1;

public class PalindromeInteger{

public static void main(String args[]){

int r,sum=0,temp;

int n=454;//It is the number variable to be checked for palindrome

temp=n;

while(n>0){

r=n%10; //getting remainder

sum=(sum\*10)+r;

n=n/10;

}

if(temp==sum)

System.out.println("palindrome number ");

else

System.out.println("not palindrome");

}

}

**7. SumOfDigits :**

package imp.level1;

import java.util.Scanner;

public class SumOfDigits

{

public static void main(String args[])

{

int number, digit, sum = 0;

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number: ");

number = sc.nextInt();

while(number > 0)

{

//finds the last digit of the given number

digit = number % 10;

//adds last digit to the variable sum

sum = sum + digit;

//removes the last digit from the number

number = number / 10;

}

//prints the result

System.out.println("Sum of Digits: "+sum);

}

}

**8. Swap Numbers:**

package imp.level1;

import java.util.Scanner;

class SwapNumbers {

//a=a+b;b=a-b;a=a-b;

/\*a=10 b=5

a=a+b=15

b=a-b=15-5=10

a=a-b=15-10=5\*/

public static void main(String args[]) {

int z, y, temp;

System.out.println("Enter z and y");

Scanner sct = new Scanner(System.in); // User inputs two numbers

z = sct.nextInt(); // User inputs two numbers

y = sct.nextInt();

System.out.println("Before Swapping\nz = " + z + "\ny = " + y);

temp = z; // Swapping is done

z = y;

y = temp;

System.out.println("After Swapping\nz = " + z + "\ny = " + y);

}

}

**9.Largest:**

package imp.level1;

import java.util.Scanner;

class Largest {

public static void main(String args[]) {

int a, b, c;

System.out.println("Enter three integers ");

Scanner sct = new Scanner(System.in);

a = sct.nextInt();

b = sct.nextInt(); // User Input

c = sct.nextInt();

if (a > b && a > c) // Condition check for Largest number

System.out.println("1st number is largest.");

else if (b > a && b > c)

System.out.println("2nd number is largest.");

else if (c > a && c > b)

System.out.println("3rd number is largest.");

else

System.out.println("Number are not distinct.");

}

}

**10. Armstrong Number:**

package imp.level1;

import java.util.Scanner;

import java.lang.Math;

public class ArmstsrongNumber {

//function to check if the number is Armstrong or not

static boolean isArmstrong(int n) {

int temp, digits = 0, last = 0, sum = 0;

//assigning n into a temp variable

temp = n;

//loop execute until the condition becomes false

while (temp > 0) {

temp = temp / 10;

digits++;

}

temp = n;

while (temp > 0) {

//determines the last digit from the number

last = temp % 10;

//calculates the power of a number up to digit times and add the resultant to the sum variable

sum += (Math.pow(last, digits));

//removes the last digit

temp = temp / 10;

}

//compares the sum with n

if (n == sum)

//returns if sum and n are equal

return true;

//returns false if sum and n are not equal

else

return false;

}

//driver code

public static void main(String args[]) {

int num;

Scanner sc = new Scanner(System.in);

System.out.print("Enter the limit: ");

//reads the limit from the user

num = sc.nextInt();

System.out.println("Armstrong Number up to " + num + " are: ");

for (int i = 0; i <= num; i++)

//function calling

if (isArmstrong(i))

//prints the armstrong numbers

System.out.print(i + ", ");

}

}

**11. Largest And Smallest In Array:**

package imp.level1;

import java.util.Arrays;

public class LargestAndSmallestInArray {

public static void largestAndSmallest(int[] numbers) {

int largest = Integer.MIN\_VALUE;

int smallest = Integer.MAX\_VALUE;

for (int number : numbers) {

if (number > largest) {

largest = number;

}

if (number < smallest) {

smallest = number;

}

}

System.out.println("Given integer array : " + Arrays.toString(numbers));

System.out.println("Largest number in array is : " + largest);

System.out.println("Smallest number in array is : " + smallest);

}

public static void secondLargest(int[] numbers) {

int largest = Integer.MIN\_VALUE;

int scondLarge=Integer.MIN\_VALUE;

for (int number : numbers) {

if (number > largest) {

scondLarge=largest;

largest = number;

} else if (scondLarge < number) {

scondLarge = number;

}

}

System.out.println("Largest number in array is : " + largest);

System.out.println("Second Largest number in array is : " + scondLarge);

}

public static void secondMin(int[] numbers) {

int min = Integer.MAX\_VALUE;

int secondMin= Integer.MAX\_VALUE;

for (int number : numbers) {

if (number < min) {

secondMin=min;

min = number;

} else if (secondMin > number) {

secondMin = number;

}

}

System.out.println("Min number in array is : " + min);

System.out.println("Second Min number in array is : " + secondMin);

}

public static void main(String[] args) {

largestAndSmallest(new int[] { 1 ,2, 5, 7, 3 ,9 });

// largestAndSmallest(new int[] { 10, Integer.MIN\_VALUE, -2 });

// largestAndSmallest(new int[] { Integer.MAX\_VALUE, 40, Integer.MAX\_VALUE });

// largestAndSmallest(new int[] { 1, -1, 0 });

secondLargest(new int[] { 1 ,2 ,5 ,7 ,3, 9});

secondMin(new int[] { 1 ,2 ,5, 7, 3, 9});

}

}

**12. Reverse Array:**

package imp.level1;

import java.util.Arrays;

public class ReverseArray {

public static void main(String[] args) {

int[] array = { 1, 2, 3, 4, 5 };

reverseArray(array);

System.out.println("Reversed array: " + Arrays.toString(array));

}

public static void reverseArray(int[] array) {

int start = 0;

int end = array.length - 1;

while (start < end) {

int temp = array[start];

array[start] = array[end];

array[end] = temp;

start++;

end--;

}

}

}

**13. Repeat Element:**

package imp.level1;

class RepeatElement {

void printRepeatingElements(int arr[], int size) {

int count[] = new int[size];

int i;

System.out.print("Repeated elements are : ");

for (i = 0; i < size; i++) {

if (count[arr[i]] == 1)

System.out.print(arr[i] + " ");

count[arr[i]]++;

}

System.out.println("");

}

void printRepeatingElementsCounts(int arr[], int size) {

int count[] = new int[size];

int i;

System.out.println("Repeated counts are : ");

for (i = 0; i < size; i++) {

count[arr[i]]++;

}

for (i = 0; i < size; i++) {

if (count[arr[i]] > 1) {

System.out.println(arr[i] + " : " + count[arr[i]]);

count[arr[i]] = 0;

}

}

}

public static void main(String[] args) {

RepeatElement repeat = new RepeatElement();

int arr[] = { 4, 2, 4, 5,4, 2, 1 };

int arr\_size = arr.length;

repeat.printRepeatingElements(arr, arr\_size);

repeat.printRepeatingElementsCounts(arr, arr\_size);

}

}

**14.Reverse String:**

package imp.level1;

import java.util.\*;

class ReverseString {

public static void main(String args[]) {

String original, reverse = "";

Scanner sct = new Scanner(System.in);

System.out.println("Enter string to reverse");

original = sct.nextLine(); // String input from user

int length = original.length();

for (int i = length - 1; i >= 0; i--) // Reversing the String

reverse = reverse + original.charAt(i);

System.out.println("Reverse of entered string is: " + reverse);

}

}

15. **Reverse Words String:**

package imp.level1;

import java.io.\*;

import java.util.\*;

import java.lang.\*;

class ReverseWordsString {

static String wordReverse(String str) {

int i = str.length() - 1;

int start, end = i + 1;

String result = "";

while (i >= 0) {

if (str.charAt(i) == ' ') {

start = i + 1;

while (start != end)

result += str.charAt(start++);

result += ' ';

end = i;

}

i--;

}

System.out.println(result);

start = 0;

while (start != end)

result += str.charAt(start++);

return result;

}

// Driver code

public static void main(String[] args) {

String str = "I AM A GEEK";

System.out.print(wordReverse(str));

}

}

**16. string reverse:**

package imp.level1;

public class stringreverse {

public static void main(String[] args) {

// TODO Auto-generated method stub

String str = "Welcome To Edureka";

String[] strArray = str.split(" ");

for (String temp: strArray){

System.out.println(temp);

}

for(int i=0; i<3; i++){ char[] s1 = strArray[i].toCharArray(); for (int j = s1.length-1; j>=0; j--)

{System.out.print(s1[j]);}

System.out.print(" ");

}

}

}

**17. String Rev:**

package imp.level1;

import java.util.Scanner;

public class StringRev {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String str = scanner.nextLine();

StringBuilder s = new StringBuilder(str);

for(int i=0,j=s.length()-1;i<j;i++,j--) {

char charAt = s.charAt(j);

s.setCharAt(j, s.charAt(i));

s.setCharAt(i, charAt);

}

System.out.println(s);

String stringRev="";

for(int i=0;i<s.length();i++)

{

stringRev=stringRev.concat(s.charAt(s.length()-i-1)+"");

}

System.out.println(stringRev);

}

}

**18. Reverse Words:**

package imp.level1;

public class ReverseWords {

public static void main(String[] args) {

String str = "Hello World";

String[] words = str.split(" ");

StringBuilder reversed = new StringBuilder();

for (int i = words.length - 1; i >= 0; i--) {

reversed.append(words[i]).append(" ");

}

System.out.println("Reversed Words: " + reversed.toString().trim());

}

}

**19. Anagram String:**

package imp.level1;

import java.util.Arrays;

public class AnagramString {

public static int NO\_OF\_CHARS = 256;

static boolean areAnagram(char str1[], char str2[]) {

int count[] = new int[NO\_OF\_CHARS];

// Arrays.fill(count, 0);

if (str1.length != str2.length)

return false;

int i;

for (i = 0; i < str1.length && i < str2.length; i++) {

count[str1[i]]++;

count[str2[i]]--;

}

for (i = 0; i < NO\_OF\_CHARS; i++)

if (count[i] > 0)

return false;

return true;

}

public static void main(String[] args) {

char str1[] = ("geeksforgeeks").toCharArray();

char str2[] = ("forgeeksgeeks").toCharArray();

System.out.println(areAnagram(str1, str2));

}

}

**20. Words Reverse:**

package imp.level1;

public class WordsReverse {

public static void main(String args[]) {

/\*String s=new String("abcd xyz");

char[] charArray = s.toCharArray();

int n=charArray.length;

for (int i=0;i<charArray.length/2; i++) {

char c = charArray[n-i-1];

charArray[n-i-1]=charArray[i];

charArray[i]=c;

}

System.out.println(String.valueOf(charArray));

\*/

//indexOf,charAt,setCharAt,length

StringBuilder s=new StringBuilder("abcd xyz");

System.out.println(s);

int index=0;

int n=s.length();

for (int i=0;i<s.length(); i++) {

char c = s.charAt(i);

if(c==' '||i==n-1) {

if(i==n-1)

i=i+1;

/\*for (int j=index,k=0;j<(i+index)/2; j++,k++) {

char c1 = s.charAt(i-k-1);

s.setCharAt(i-k-1, s.charAt(j));

s.setCharAt(j,c1);

}\*/

for (int j=index,k=i-1;j<k; j++,k--) {

char c1 = s.charAt(j);

s.setCharAt(j, s.charAt(k));

s.setCharAt(k,c1);

}

index=i+1;

}

}

System.out.println(s);

stringReveerse(s);

System.out.println(s);

}

private static void stringReveerse(StringBuilder s) {

int n=s.length();

for (int i=0;i<s.length()/2; i++) {

char c = s.charAt(n-i-1);

s.setCharAt(n-i-1, s.charAt(i));

s.setCharAt(i,c);

}

}

}

**21. Palindrome String:**

package imp.level1;

import java.util.Scanner;

public class PalindromeString {

static void checkPalindrome(String input) {

//Assuming result to be true

boolean res = true;

int length = input.length();

//dividing the length of the string by 2 and comparing it.

for(int i=0; i<= length/2; i++) {

if(input.charAt(i) != input.charAt(length-i-1)) {

res = false;

break;

}

}

System.out.println(input + " is palindrome = "+res);

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter your Statement: ");

String str = sc.nextLine();

//function call

checkPalindrome(str);

}

}

**Pattern:**

package imp.pattern;

import java.util.Scanner;

public class InvertedHalfPyramidNumbers {

public static void main(String[] args) {

int i, j, rows;

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of rows: ");

rows = scanner.nextInt();

for (i = rows; i >= 1; --i) {

for (j = 1; j <= i; ++j) {

System.out.print(j + " ");

}

System.out.println();

}

scanner.close();

}

}

***Input:****n = 5****Output:*** *\*\*\*\*\*  
\*\*\*\*  
\*\*\*  
\*\*  
\**

package imp.pattern;

import java.util.Scanner;

public class HalfPyramidAlphabets {

public static void main(String[] args) {

int i, j;

char input, alphabet = 'A';

Scanner scanner = new Scanner(System.in);

System.out.print("Enter an uppercase character you want to print in the last row: ");

input = scanner.next().charAt(0);

for (i = 1; i <= (input - 'A' + 1); ++i) {

for (j = 1; j <= i; ++j) {

System.out.print(alphabet + " ");

}

++alphabet;

System.out.println();

}

scanner.close();

}

}

A

B B

C C C

D D D D

E E E E E

F F F F F F

package imp.pattern;

import java.util.Scanner;

public class InvertedHalfPyramidStars {

public static void main(String[] args) {

int i, j, rows;

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of rows: ");

rows = scanner.nextInt();

for (i = rows; i >= 1; --i) {

for (j = 1; j <= i; ++j) {

System.out.print("\* ");

}

System.out.println();

}

scanner.close();

}

}

***Input:****5****Output:*** *\*\*\*\*\*  
\*\*\*\*  
\*\*\*  
\*\*  
\**

package imp.pattern;

import java.util.Scanner;

public class FullPyramidStars {

public static void main(String[] args) {

int rows, k = 0;

System.out.print("Enter the number of rows: ");

rows = STDIN\_SCANNER.nextInt();

for(int i = 1; i <= rows; ++i, k = 0) {

for(int space = 1; space <= rows - i; ++space) {

System.out.print(" ");

}

while(k != 2 \* i - 1) {

System.out.print("\* ");

++k;

}

System.out.println();

}

}

public final static Scanner STDIN\_SCANNER = new Scanner(System.in);

}

\*

\* \* \*

\* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \*

package imp.pattern;

import java.util.Scanner;

public class FloydTriangle {

public static void main(String[] args) {

int rows, number = 1;

System.out.print("Enter the number of rows: ");

rows = STDIN\_SCANNER.nextInt();

for(int i = 1; i <= rows; i++) {

for(int j = 1; j <= i; ++j) {

System.out.print(number + " ");

++number;

}

System.out.println();

}

}

public final static Scanner STDIN\_SCANNER = new Scanner(System.in);

}

1

2 3

4 5 6

7 8 9 10

package imp.pattern;

import java.util.Scanner;

public class FullPyramidNumbers {

public static void main(String[] args) {

int rows, k = 0, count = 0, count1 = 0;

System.out.print("Enter the number of rows: ");

rows = STDIN\_SCANNER.nextInt();

for(int i = 1; i <= rows; ++i) {

for(int space = 1; space <= rows - i; ++space) {

System.out.print(" ");

++count;

}

while(k != 2 \* i - 1) {

if(count <= rows - 1) {

System.out.print((i + k) + " ");

++count;

} else {

++count1;

System.out.print((i + k - 2 \* count1) + " ");

}

++k;

}

count1 = count = k = 0;

System.out.println();

}

}

public final static Scanner STDIN\_SCANNER = new Scanner(System.in);

}

1

2 3 2

3 4 5 4 3

4 5 6 7 6 5 4

5 6 7 8 9 8 7 6 5

package imp.pattern;

import java.util.Scanner;

public class HalfPyramidStars {

public static void main(String[] args) {

int i, j, rows;

Scanner scanner = new Scanner(System.in);

System.out.print("Enter number of rows: ");

rows = scanner.nextInt();

for (i = 1; i <= rows; ++i) {

for (j = 1; j <= i; ++j) {

System.out.print("\* ");

}

System.out.println();

}

scanner.close();

}

}

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

package imp.pattern;

import java.util.Scanner;

public class HalfPyramidNumbers {

public static void main(String[] args) {

int i, j, rows;

Scanner scanner = new Scanner(System.in);

System.out.print("Enter number of rows: ");

rows = scanner.nextInt();

for (i = 1; i <= rows; ++i) {

for (j = 1; j <= i; ++j) {

System.out.print(j + " ");

}

System.out.println();

}

scanner.close();

}

}

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

package imp.pattern;

public class PrintPattern {

public static void main(String[] args) {

int rows = 5;

for (int i = 1; i <= rows; ++i) {

for (int j = 1; j <= i; ++j) {

System.out.print("\* ");

}

System.out.println();

}

}

}

package imp.pattern;

import java.util.Scanner;

public class PascalTriangle {

public static void main(String[] args) {

int rows, coef = 1;

System.out.print("Enter the number of rows: ");

rows = STDIN\_SCANNER.nextInt();

for(int i = 0; i < rows; i++) {

for(int space = 1; space <= rows - i; space++) {

System.out.print(" ");

}

for(int j = 0; j <= i; j++) {

if(j == 0 || i == 0) {

coef = 1;

} else {

coef = coef \* (i - j + 1) / j;

}

System.out.printf("%4d", coef);

}

System.out.println();

}

}

public final static Scanner STDIN\_SCANNER = new Scanner(System.in);

}

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

package imp.pattern;

import java.util.Scanner;

public class InvertedFullPyramidStars {

public static void main(String[] args) {

int rows;

System.out.print("Enter the number of rows: ");

rows = STDIN\_SCANNER.nextInt();

for(int i = rows; i >= 1; --i) {

for(int space = 0; space < rows - i; ++space) {

System.out.print(" ");

}

for(int j = i; j <= 2 \* i - 1; ++j) {

System.out.print("\* ");

}

for(int j = 0; j < i - 1; ++j) {

System.out.print("\* ");

}

System.out.println();

}

}

public final static Scanner STDIN\_SCANNER = new Scanner(System.in);

}

\* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \*

\* \* \*

\*

package imp.pattern;

import java.util.Scanner;

public class PatternA {

// Java program to print alphabet A pattern

void display(int n) {

// Outer for loop for number of lines

for (int i = 0; i <= n; i++) {

// Inner for loop for logic execution

for (int j = 0; j <= n / 2; j++) {

// prints two column lines

if ((j == 0 || j == n / 2) && i != 0 ||

// print first line of alphabet

i == 0 && j != n / 2 ||

// prints middle line

i == n / 2)

System.out.print("\*");

else

System.out.print(" ");

}

System.out.println();

}

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

PatternA a = new PatternA();

a.display(7);

}

}