JAVA SAMPLE PROGRAMS

Premkumar Balasubramanian

Java Basic Programs :

1. **Fibonacci series:**

```
Int n1=0, n2=1, I, count=10;
System,out.println("n1" + "n2");
For(i=2;i<count;i++)
{
N3=n2+n1;
System.out.println(n3);
N1=n2;
N2=n3;
}
```

2.Triangle:

Eg: 1: 1

1 1

1 1 1

1 1 1 1

```
Int a1=1; I, j;
For ( int i=0;i<5;i++)
{
  For(j=0;j<=I;j++)
    {
      System.out.print(a+ " ");
    }
  System.out.println();
}
```

---

3) 1

```
   21
  321
 4321
54321
654321
7654321
87654321
987654321
10987654321
```

```
int lines =10;
             for (int i = 1; i <= lines; i++){
                   System.out.print(" ");
                     for (int j = lines; j > 0; j--)
                         System.out.print (j > i ? " " : j);
                     System.out.println();
                 }
             }
```

---

4) Flyod's Triangle

Eg:

1

2 3

4 5 6

7 8 9 10

```
Int n=10,num=1;
for ( int i = 1 ; i <= n ; i++ )
    {
      for ( int j = 1 ; j <= i ; j++ )
      {
        System.out.print(num+" ");
        num++;
      }
      System.out.println();
        }
```

5) ReverseNumber

```
public class FindReverseNumber
{ public static void main(String[] args)
{
 //number defined
 int number = 1234;
 int reversedNumber = 0;
int temp = 0;
while(number > 0)
{
 //modulus operator used to strip off the last digit
 temp = number%10; //create reversed number
reversedNumber = reversedNumber * 10 + temp; number = number/10;
}
//output System.out.println("Reversed Number is: " + reversedNumber);
}
}
```

6) JavaPalindromeNumberExample

```
public class JavaPalindromeNumberExample {
        public static void main(String[] args) {
                //array of numbers to be checked
                int numbers[] = new int[]{121,13,34,11,22,54};
                //iterate through the numbers
                for(int i=0; i < numbers.length; i++){
                        int number = numbers[i];
                        int reversedNumber  = 0;
                        int temp=0;
                        /*
```

```
                * If the number is equal to it's reversed number, then
                * the given number is a palindrome number.
                * For example, 121 is a palindrome number while 12 is not.
                */
               //reverse the number
               while(number > 0){
                    temp = number % 10;
                    number = number / 10;
                    reversedNumber = reversedNumber * 10 + temp;
               }
               if(numbers[i] == reversedNumber)
                    System.out.println(numbers[i] + " is a palindrome number");
               else
                    System.out.println(numbers[i] + " is not a palindrome number");
          }
     }
}
/*
Output of Java Palindrome Number Example would be
121 is a palindrome number
13 is not a palindrome number
34 is not a palindrome number
11 is a palindrome number
22 is a palindrome number
54 is not a palindrome number
*/
```

7)Invert Pyramid Triangle

```
public class JavaPyramid2 {


	public static void main(String[] args) {
		for(int i=5; i>0 ;i--){
			for(int j=0; j < i; j++){
				System.out.print("*");
			}
			//generate a new line
			System.out.println("");
		}
	}
}
```

Output of the **example** would be

```
*****
****
***
**
*
```

---

8) GeneratePrimeNumbersExample

```
public class GeneratePrimeNumbersExample {
	public static void main(String[] args) {
		//define limit
		int limit = 100;
		System.out.println("Prime numbers between 1 and " + limit);
		//loop through the numbers one by one
		for(int i=1; i < 100; i++){
```

```
            boolean isPrime = true;
            //check to see if the number is prime
            for(int j=2; j < i ; j++){
                if(i % j == 0){
                    isPrime = false;
                    break;
                }
            }
            // print the number
            if(isPrime)
                System.out.print(i + " ");
        }
    }
}
/*
Output of Prime Numbers example would be
Prime numbers between 1 and 100
1 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
```

---

9. Program to read from file line by line:

bufferedReader class provides readLine method to be used to read the file line by line.

```
public static void readFile() throws FileNotFoundException {
FileReader fr = new FileReader("C:\\Users\\...\\Desktop\\unused.txt");
BufferedReader br = new BufferedReader(fr);
StringBuffer str = new StringBuffer();
try {
while (br.readLine()!= null){
str.append(br.readLine());
}
} catch (IOException e) {
e.printStackTrace();
}
System.out.println(str);
}
```
