Java

<u>Display Result on The Console:</u>
 In C-Language : stdio.h --> printf( )
 In C++ : iostream.h -> Ostream(class) --> cout(object)

In Java
 println( ) | print( ) | printf( )

Packages <-- Classes <-- methods[Functions
     Eg: java.io <-- PrintStream <-- println()

println( )
   - Predefined Method from PrintStream class
      From java.io package
   - Used to display result on screen
         println( ) --> printf( )+"\n"
   -It is an Overloaded method

println( ) | println(int) | println(long)
println(float) | println(double) | println(char)
println(char[ ])| println(String)| println(boolean)
println(Object)

java.io.PrintStream
    public void println(xxx); //signature
         Rep: System.out.println(xxx);

Will it possible to print more than one value using System.out.println( ) ?

Yes, Possible Using an Operator +

In Java an operator + is polymorphic behavior natured operator,ie. Operator is only one but
behavior the operator is getting changing based environment
*Eg: if we pass two integer as an operands [10+20] then it will perform addition and produce the
sum*

        Eg: System.out.println(10+20); //10+20 → 30

If we pass two string as an operands ["Sai"+"Baba"] then it will concatenate [join] and produce the concatenated output

    Eg: System.out.println("Sai"+"Baba"); //SaiBaba

System.out.println(10+" "+20); //10   20
  //any thing + string -> concatenate

Example Prg:
//PrintlnDemo.java
class PrintlnDemo
{
  public static void main(String args[ ])
  {
    System.out.println("welcome"); //println(String)
    System.out.println('s'); //println(char)
    System.out.println(10); //println(int)
    System.out.println(10+20); //30
    System.out.println("Sai"+"Baba"); //SaiBaba
    System.out.println(10+"   "+20);
  }
}

print( )
- Print( ) will display the result on the screen in the same line

    java.io.PrintStream
      public void print(xxx); //overloaded
        Rep: System.out.print(xxx);

print (int) | print (long)
printlnfloat) | print (double) | print(char)
print (char[ ])| print(String)| print(boolean)
print(Object)

System.out.println( ); //valid

Java

System.out.print(); //Error

Example For print():
//PrintDemo.java
class PrintDemo
{
  public static void main(String args[ ])
  {
    System.out.print('s'); //print(char)
    System.out.print("Shashi"); //print(String)
    System.out.print(10+20); //print(int) int+int 10+20
    System.out.print("ba"+"ba"); //print(String) str+str -> str

    System.out.println( ); //println( )
    System.out.print(); //print() Error
  }
}

printf( )
- It will work as same as printf( ) in c-Language
  java.io.PrintStream
    public void printf(String,Object,….);
      Rep: System.out.printf(String,Object,…);

```
//PrintfDemo.java
class PrintfDemo{
  public static void main(String args[]){
    int x=10;
    String s="SssiT";
    System.out.printf("\n x val is : %d",x);
    System.out.printf("\n s val is : %s",s);
  }
}
```

Note: Based on your application requirements we can also use esc sequence char [\n | \t | \a | \b | \r] in any output function [println( ) | print( ) | printf( )] , But we can use format specifiers [%d | %f | %s] in printf() only