[What happens When System.out.println(null)?](http://www.instanceofjava.com/2014/12/what-happens-when-systemoutprintlnnull.html)

//System.out.println(char[] arr);//error

System.*out*.println((String)null);//null

//System.out.println((char[])null);//null Pointer exception

System.*out*.println((Object) null);

* Compilation Error -This is because you can pass an Object or a String or char[]. Since null can fit in both, the compiler doesn't know which method to use, leading to compile error.

Method Overloading:  
    1.public void prinltln(String str) { }  
    2.public void prinltln(char[] ch){ }  
    3.public void prinltln(Object ch){

 It seems the call System.out.print(null) is ambiguous to compiler because print(null) here will find the two best specific matches i.e. print(String) and print(char[]) . So compiler is unable to determine which method to call here .

Compilation Error:  
   System.out.println(null)

1. public class StaticDemo{

4. static {
5. i=10;
6. }
8. static   int i;
10. public static void main(String[] args) {
12. System.out.println("i= "+i);
13. }
14. }

10

1. public class StaticDemo{
3. static {
4. i=10;
5. }
7. static   int i;
9. public static void main(String[] args) {
10. StaticDemo obj= new StaticDemo();
11. obj.i=20;
13. System.out.println("i= "+StaticDemo.i);
14. }

20

1. int a=0,b=0;
3. for(int i=0;i<5;i++){
4. if(++a>2||++b>2){
5. a++;
6. }
7. }
9. System.out.println("a= "+a+" b="+b);

A=8 ,b=2

1. byte[] byteArray = new byte[] {78,73, 67,69};
2. String value = new String(byteArray);
3. System.out.println(value);
5. }
7. public static void main(String[] args) {
8. ByteArrayToString.convertByteArrayToString();
9. }

NICE

### Convert String to Byte Array:

1. package com.instanceofjava;
2. public class StringTOByteArray{
3. /\*
4. \* This example shows how to convert a String object to byte array.
5. \*/
6. public static void main(String[] args) {
8. String data = "Instance of java";
10. byte[] byteData = data.getBytes();
12. System.out.println(byteData);
14. }
16. }
17. [B@3b26456a