KODNEST ASSIGNEMENT

In Java, the bitwise complement operator (~) can be used to perform a bitwise complement operation on an integer value. It flips the bits of the operand, turning 0s into 1s and 1s into 0s. Here's an example:

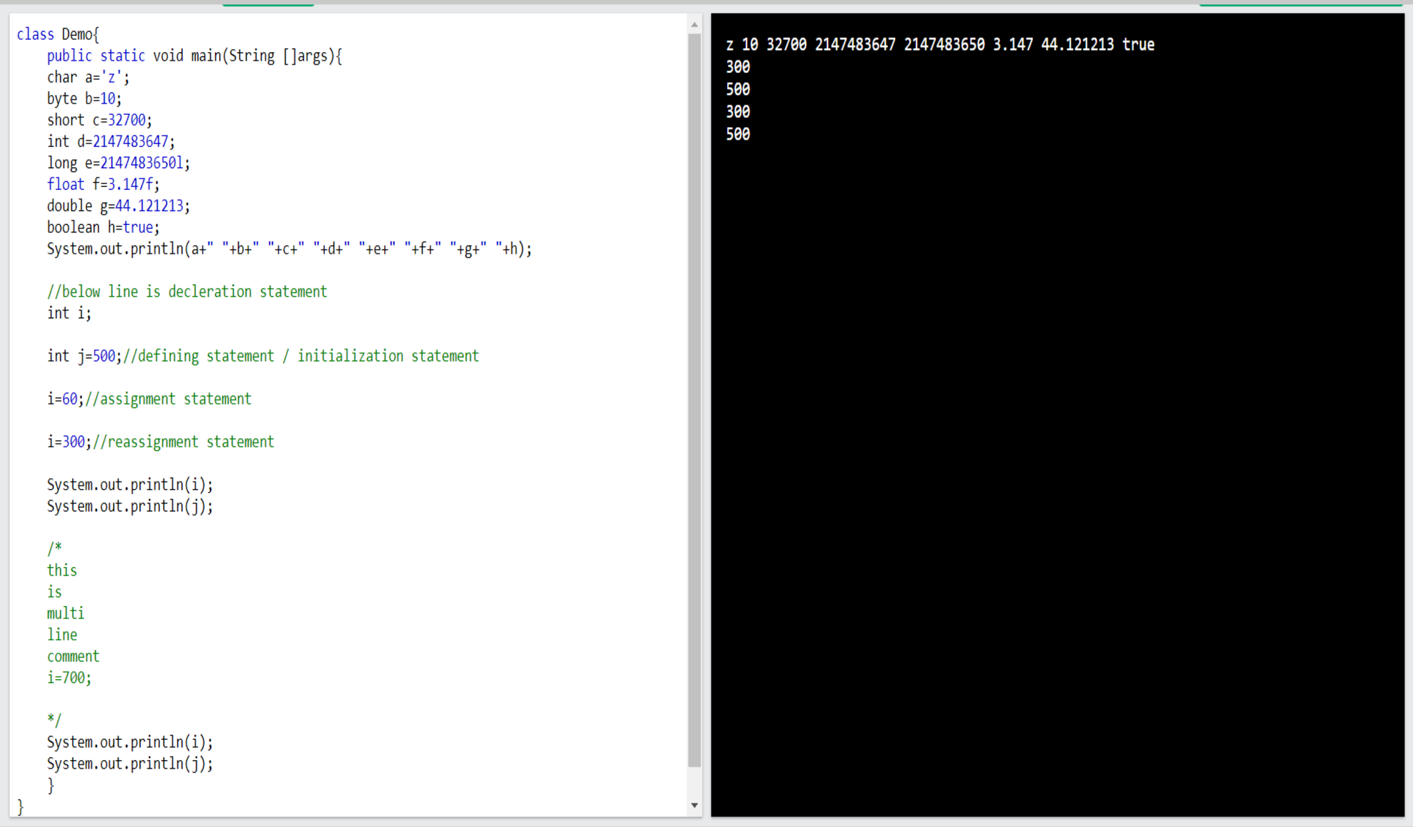
```
int num = 42; // Binary: 00101010
int complement = ~num; // Binary complement: 11010101
System.out.println(complement); // Output: -43
```

In Java, the logical complement operator (!) can be used to perform a logical complement operation on a boolean value. It negates the value of a boolean, flipping true to false and false to true. Here's an example:

```
boolean flag = true;
boolean complement = !flag;
System.out.println(complement); // Output: false
```

- Minimum value of float: **1.4E-45**
- Maximum value of float: **3.4028235E38**

- Minimum value of double: **−1.7976931348623157×10308**
- Maximum value of double: **1.7976931348623157×10308**



```
class Demo{
    public static void main(String []args){
    char a='z';
    byte b=10;
    short c=32700;
    int d=2147483647;
    long e=2147483650l;
    float f=3.147f;
    double g=44.121213;
    boolean h=true;
    System.out.println(a+" "+b+" "+c+" "+d+" "+e+" "+f+" "+g+" "+h);

    //below line is decleration statement
    int i;

    int j=500;//defining statement / initialization statement

    i=60;//assignment statement

    i=300;//reassignment statement

    System.out.println(i);
    System.out.println(j);

    /*
    this
    is
    multi
    line
    comment
    i=700;

    */
    System.out.println(i);
    System.out.println(j);
    }
}
```

```
z 10 32700 2147483647 2147483650 3.147 44.121213 true
300
500
300
500
```

https://drive.google.com/file/d/1GJnTTpFg1sxTyqoiddUaoyKmAaX9FYUc/view?usp=drive_link