

## Day 4 - Code Java Smart

# Palindrome

A **palindrome** is a **word, number, phrase**, or other sequence of symbols that reads the **same backwards as forwards**, such as **madam** or **racecar**, the date **"22/02/2022"** and the sentence: **"A man, a plan, a canal – Panama"**

There are 2 ways to solve this problem :

1. Using the `StringBuilder`'s inbuilt `reverse()` function .  
The **`StringBuilder`** class exists in the **`java.lang`** package and is **automatically imported by default**.
2. Extracting digits from original number and forming its reverse .

*`public StringBuilder reverse()`*

*Causes this character sequence to be replaced by the reverse of the sequence. If there are any surrogate pairs included in the sequence, these are treated as single characters for the reverse operation. Thus, the order of the high-low surrogates is never reversed. Let  $n$  be the character length of this character sequence (not the length in char values) just prior to execution of the reverse method. Then the character at index  $k$  in the new character sequence is equal to the character at index  $n-k-1$  in the old character sequence.*

*Note that the reverse operation may result in producing surrogate pairs that were unpaired low-surrogates and high-surrogates before the operation. For example, reversing `"\uD800\uDC00"` produces `"\uD800\uDC00"` which is a valid surrogate pair*

Returns: a reference to this object.

## Program

1. `reverse()`

```
class Solution {  
    public boolean isPalindrome(int x) {  
        StringBuilder st=new StringBuilder(""+x);
```

```
        return st.reverse().toString().equals(""+x);  
    }  
}
```

```
    2. rev = (rev*10) + d  
class Solution {  
    public boolean isPalindrome(int x) {  
  
        if(x<0) // for negative numbers as -121 not eq. to 121-  
            return false;  
  
        int rev=0,copy=x;  
        while(copy!=0)  
        {  
            int d=copy%10;  
            rev=(rev*10)+d;  
            copy=copy/10;  
        }  
  
        return rev==x;  
    }  
}
```