**Reverse String by using Stack.**

This is one of the interview question. This question demostrate that does the candidate knows about the Stack's push and pop operations and How the stack works.

String "abc" would be be pushed onto Stack like this.

| c |

| b |

| a |

Then we will pop() elements from stack. Hence string will be reversed.

Below is the class that implements the method to reverse the String using Stack<E>.

**package** stackimpl;

**import** java.util.Stack;

**public** **class** StringReverseByStack {

**public** **static** String reverseByStack(String str){

/\*\*

\* If input string is null then throw NullPointerException.

\* \*/

**if** (str == **null**) {

**throw** **new** NullPointerException("Input Should not be null");

}

/\*\*

\* Push characters in Stack.

\* \*/

Stack<Character> stack=**new** Stack<Character>();

**for**(**char** ch:str.toCharArray()){

stack.push(ch);

}

StringBuilder sb=**new** StringBuilder();

/\*\*

\* As Stack is Last In First Out last character will be popped first.

\* Do it till Stack is empty.

\* \*/

**while**(!stack.isEmpty()){

sb.append(stack.pop());

}

//return the reversed String

**return** sb.toString();

}

}

Below are the designed test cases for the above written method.

Test cases are for null string, empty string, single length string and string length > 2.

**package** stackimpl;

**import** junit.framework.Assert;

**import** org.junit.Test;

**import** strings.StringReverse;

**public** **class** StringReverseByStackTest {

**final** String SINGLE\_LENGTH = "a";

**final** String STR1 = "abc";

**final** String NULL = **null**;

**final** String EMPTY\_STRING = "";

@Test(expected = NullPointerException.**class**)

**public** **void** testReverseByStackNull() {

StringReverse.*reverseByStack*(NULL);

}

@Test

**public** **void** testReverseByStackEmptyString() {

Assert.*assertEquals*(EMPTY\_STRING,

StringReverse.*reverseByStack*(EMPTY\_STRING));

}

@Test

**public** **void** testReverseByStackSingleLength() {

Assert.*assertEquals*("a", StringReverse.*reverseByStack*(SINGLE\_LENGTH));

}

@Test

**public** **void** testReverseByStack() {

Assert.*assertEquals*("cba", StringReverse.*reverseByStack*(STR1));

}

}

