

Our Solution(s)		Run Code
Solution 1	Solution 2	
<pre>1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved. 2 3 class Program { 4 // O(n^3) time O(1) space 5 public static String longestPalindromicSubstring(String str) { 6 String longest = ""; 7 for (int i = 0; i < str.length(); i++) { 8 for (int j = i; j < str.length(); j++) { 9 String substring = str.substring(i, j + 1); 10 if (substring.length() > longest.length() && isPalindromicSubstr 11 longest = substring; 12 } 13 } 14 } 15 return longest; 16 } 17 18 public static boolean isPalindrome(String str) { 19 int leftIdx = 0; 20 int rightIdx = str.length() - 1; 21 while (leftIdx < rightIdx) { 22 if (str.charAt(leftIdx) != str.charAt(rightIdx)) { 23 return false; 24 } 25 leftIdx++; 26 rightIdx--; 27 } 28 return true; 29 } 30 } 31</pre>		

Your Solutions			Run Code
Solution 1	Solution 2	Solution 3	
<pre>1 class Program { 2 public static String longestPalindromicSubstring(String str) { 3 // Write your code here. 4 return ""; 5 } 6 } 7</pre>			

```
1 #Test
2 #Test
3 #Test
4 #Test
5 #Test
6 #Test
7 #Test
8 #Test
9 #Test
10 #Test
11 #Test
12 #Test
13 #Test
14 #Test
15 #Test
16 #Test
17 #Test
18 #Test
19 #Test
20 #Test
21 #Test
22 #Test
23 #Test
24 #Test
25 #Test
26 #Test
27 #Test
28 #Test
29 #Test
30 #Test
31 #Test
32 #Test
33 #Test
34 #Test
35 #Test
36 #Test
37 #Test
38 #Test
39 #Test
40 #Test
41 #Test
42 #Test
43 #Test
44 #Test
45 #Test
46 #Test
47 #Test
48 #Test
49 #Test
50 #Test
51 #Test
52 #Test
53 #Test
54 #Test
55 #Test
56 #Test
57 #Test
58 #Test
59 #Test
60 #Test
61 #Test
62 #Test
63 #Test
64 #Test
65 #Test
66 #Test
67 #Test
68 #Test
69 #Test
70 #Test
71 #Test
72 #Test
73 #Test
74 #Test
75 #Test
76 #Test
77 #Test
78 #Test
79 #Test
80 #Test
81 #Test
82 #Test
83 #Test
84 #Test
85 #Test
86 #Test
87 #Test
88 #Test
89 #Test
90 #Test
91 #Test
92 #Test
93 #Test
94 #Test
95 #Test
96 #Test
97 #Test
98 #Test
99 #Test
100 #Test
```

Run or submit code when you're ready.