Python Program to Check if a String is Palindrome or Not

Last Updated : 06 Sep, 2024

Given a string, write a python function to check if it is palindrome or not. A string is said to be a palindrome if the reverse of the string is the same as the string. For example, “radar” is a palindrome, but “radix” is not a palindrome.

Examples:

```
Output : Yes
Input : geeks
Output : No
```

Python Program to Check if a String is Palindrome or Not Using Native Approach

Here we will find reverse of the string and then Check if reverse and original are same or not.

Python

```python
# function which return reverse of a string

def isPalindrome(s):
    return s == s[::-1]


# Driver code
s = "malayalam"
ans = isPalindrome(s)

if ans:
    print("Yes")
else:
```

Output

```
Yes
```

Time complexity: O(n)
Auxiliary Space: O(1)

Check if a String is Palindrome or Not Using Iterative Method

Run a loop from starting to length/2 and check the first character to the last character of the string and second to second last one and so on .... If any character mismatches, the string wouldn't be a palindrome.

Below is the implementation of above approach:

Python

```
# function to check string is
```

```
    for i in range(0, int(len(str)/2)):
        if str[i] != str[len(str)-i-1]:
            return False
    return True

# main function
s = "malayalam"
ans = isPalindrome(s)

if (ans):
    print("Yes")
else:
    print("No")
```

Output

```
Yes
```

Time complexity: O(n)
Auxiliary Space: O(1)

Check String is Palindrome or Not Using the inbuilt function to reverse a string

In this method, the predefined function **' '.join(reversed(string))** is used to reverse string.

Below is the implementation of the above approach:

Python

```
# function to check string is
# palindrome or not
def isPalindrome(s):

    # Using predefined function to
    # reverse to string print(s)
    rev = ''.join(reversed(s))

    # Checking if both string are
    # equal or not
    if (s == rev):
        return True
    return False

# main function
s = "malayalam"
```

```
else:
    print("No")
```

Output

```
Yes
```

Time complexity: O(n)
Auxiliary Space: O(n)

Check String is Palindrome using one extra variable

In this method, the user takes a character of string one by one and store it in an empty variable. After storing all the characters user will compare both the string and check whether it is palindrome or not.

Python

```python
x = "malayalam"

w = ""
for i in x:
    w = i + w

if (x == w):
    print("Yes")
else:
    print("No")
```

Output

```
Yes
```

Time complexity: O(n)
Auxiliary Space: O(n)

Check String is Palindrome using flag

In this method, the user compares each character from starting and ending in a for loop and if the character does not match then it will change the status of the flag. Then it will check the status of the flag and accordingly and print whether it is a palindrome or not.

Python

```python
# Python program to check
# if a string is palindrome
# or not
st = 'malayalam'
j = -1
flag = 0
for i in st:
    if i != st[j]:
        flag = 1
        break
    j = j - 1
if flag == 1:
    print("NO")
else:
    print("Yes")
```

```
Yes
```

Time complexity: O(n)
Auxiliary Space: O(1)

Check String is Palindrome using recursion

This method compares the first and the last element of the string and gives the rest of the substring to a recursive call to itself.

Python

```python
# Recursive function to check if a
# string is palindrome
def isPalindrome(s):

    # to change it the string is similar case
    s = s.lower()
    # length of s
    l = len(s)

    # if length is less than 2
    if l < 2:
        return True

    # If s[0] and s[l-1] are equal
    elif s[0] == s[l - 1]:

        # Call is palindrome form substring(1,l-1)
        return isPalindrome(s[1: l - 1])

    else:
        return False

# Driver Code
s = "MalaYaLam"
ans = isPalindrome(s)

if ans:
    print("Yes")

else:
    print("No")
```

Output

```
Yes
```

Python Program to Check if a String is Palindrome or Not – FAQs

What is a strong number in Python?

> *A strong number (or Krishnamurthy number) is a number whose sum of the factorial of its digits equals the original number. For example, 145 is a strong number because 1!+4!+5!=1+24+120=145.*

What are any four palindrome words?

> 1. *"racecar"*
> 2. *"level"*
> 3. *"radar"*
> 4. *"civic"*

How do you identify a palindrome function?

> *A function correctly checks for palindromes if it returns `True` for strings that read the same forward and backward and `False` otherwise. Test it with known palindromes (e.g., "madam") and non-palindromes (e.g., "hello").*

Can I use slicing or reversing techniques to check for palindromes?

> *Yes, you can use slicing (`s == s[::-1]`) or reversing techniques (`s == ''.join(reversed(s))`) to check if a string is a palindrome. Both methods compare the string with its reverse to determine if it reads the same forward and backward.*

How to write a Python program to check whether a string is palindrome or not using a stack?

```
def is_palindrome(s):
    stack = []
    
    # Push all characters to the stack
    for char in s:
        stack.append(char)
    
    # Pop characters from the stack and compare with the
original string
    for char in s:
        if char != stack.pop():
            return False
    
    return True

# Example usage
string = "racecar"
print(f"{string} is a palindrome: {is_palindrome(string)}")  #
Output: True

string = "hello"
print(f"{string} is a palindrome: {is_palindrome(string)}")  #
Output: False
```

GeeksforGeeks 119

Next Article

Article Tags : Python Python Programs Python string-programs

Practice Tags : python

Bootstrap
Web Design

Django

Computer Science

Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design
Engineering Maths
Software Development
Software Testing

DevOps

Git
Linux
AWS
Docker
Kubernetes
Azure
GCP
DevOps Roadmap

System Design

High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD
System Design Bootcamp
Interview Questions

Inteview Preparation

Competitive Programming
Top DS or Algo for CP
Company-Wise Recruitment Process
Company-Wise Preparation
Aptitude Preparation
Puzzles

School Subjects

Mathematics
Physics
Chemistry
Biology
Social Science
English Grammar
Commerce
World GK

GeeksforGeeks Videos

DSA
Python
Java
C++
Web Development
Data Science
CS Subjects