

# Shell 字符串

2022年10月21日 17:24

## 1. 单引号

```
str='this is a string'
```

- 单引号字符串的限制：
  - 单引号里的任何字符都会原样输出，单引号字符串中的变量是无效的；
  - 单引号字符串中不能出现单独一个的单引号（对单引号使用转义符后也不行），但可成对出现，作为字符串拼接使用。

## 2. 双引号

```
your_name='runoob' str="Hello, I know you are \"${your_name}\"! \n" echo -e $str #输出结果为: #Hello, I know you are "runoob"!
```

- 双引号的优点：
  - 双引号里可以有变量
  - 双引号里可以出现转义字符

## 3. 拼接字符串

```
your_name="runoob" #使用双引号拼接 greeting="hello, "${your_name}"! " greeting_1="hello, ${your_name}!" echo $greeting $greeting_1 #使用单引号拼接 greeting_2='hello, '${your_name}'!' greeting_3='hello, ${your_name}'!' echo $greeting_2 $greeting_3 #输出结果为: #hello, runoob ! hello, runoob ! #hello, runoob ! hello, ${your_name} !
```

## 4. 获取字符串长度

```
string="abcd" echo ${#string} #输出 4
```

## 5. 提取子字符串

以下实例从字符串第 2 个字符开始截取 4 个字符：

```
string="runoob is a great site" echo ${string:1:4} #输出 unoo
```

## 6. 查找子字符串

查找字符 i 或 o 的位置(哪个字母先出现就计算哪个:)

```
string="runoob is a great site" echo `expr index "${string}" io` #输出 4 注意： 以上脚本中 ` 是反引号，而不是单引号 '，不要看错了哦。
```