

python 字符串前加f、r、b、u 分别代表什么?

在Python中，字符串前面加上特定的前缀有不同的含义：

版权

1. 前缀 **f** 表示格式化字符串 (Formatted String)：使用f-string可以在字符串中嵌入表达式，并将其值格式化为字符串。在f-string中，用花括号 **{}** 括起的表达式会被替换为相应的值。例如：

```
1 | name = "Alice"
2 | age = 25
3 | message = f"My name is {name} and I'm {age} years old."
4 | print(message)
```

输出：

```
1 | My name is Alice and I'm 25 years old.
```

登录后复制

2. 前缀 **r** 表示原始字符串 (Raw String)：原始字符串中的转义字符不会被转义，而是按照字面意义来解释。常见用途是在正则表达式中使用，以避免双重转义。例如：

```
1 | path = r"C:\Program Files\Python"
2 | print(path)
```

输出：

```
1 | C:\Program Files\Python
```

3. 前缀 **b** 表示字节字符串 (Bytes String)：字节字符串是以字节（二进制）形式表示的字符串。在字节字符串中，每个字符都用一个字节来表示。字节字符串通常用于处理二进制数据，例如读取图像或文件。例如：

```
1 | data = b'\x48\x65\x6c\x6c\x6f' # 表示字符串 "Hello" 的字节表示
2 | print(data.decode())
```

输出：

```
1 | Hello
```

4. 前缀 **u** 表示Unicode字符串：在Python 2中，它表示Unicode字符串，而在Python 3中，所有的字符串都是Unicode字符串，所以 **u** 前缀在Python 3中没有特殊含义，仍然可以使用。例如：

```
1 | unicode_string = u"Hello, 世界"
2 | print(unicode_string)
```

输出：

```
1 | Hello, 世界
```

需要注意的是，**f**、**r**、**b**、**u** 前缀可以组合使用，例如 **fr"..."** 表示既是**格式化字符串** 又是**原始字符串**。