

Python终端魔法：使用ANSI转义码打印彩色文本的简单方法

📅 作者：微信公众号：【架构师老卢】

📅 12-19 18:8

~ 351



概述: 在Python中, 通过使用ANSI转义码可以在终端中实现彩色文本打印。通过定义颜色代码, 我们可以轻松实现不同颜色的文本输出, 提升终端显示效果。然而, 注意ANSI转义码在某些终端上可能不被完全支持, 也可考虑使用第三方库如`colorama`以获得更复杂的彩色文本控制。

在Python中, 可以使用ANSI转义码实现在终端中打印彩色文本。以下是一种简单的方法:

1. 使用ANSI转义码

```
1 # 导入 sys 模块
2 import sys
3
4 # 定义颜色代码
5 class Colors:
6     RED = '\033[91m'
7     GREEN = '\033[92m'
8     YELLOW = '\033[93m'
9     BLUE = '\033[94m'
10    RESET = '\033[0m'
11
12 # 打印彩色文本
13 def print_colored_text(text, color):
14     print(f"{color}{text}{Colors.RESET}")
15
16 # 示例
17 if __name__ == "__main__":
18     print_colored_text("红色文本", Colors.RED)
19     print_colored_text("绿色文本", Colors.GREEN)
20     print_colored_text("黄色文本", Colors.YELLOW)
21     print_colored_text("蓝色文本", Colors.BLUE)
```

这个示例中, 定义了一个 `Colors` 类, 其中包含了ANSI转义码表示的不同颜色。然后, 通过 `print_colored_text` 函数将文本和颜色代码组合起来, 实现在终端中打印彩色文本。

2. 注意事项

- ANSI转义码的支持因终端而异, 一些终端可能不支持。
- 在Windows的cmd中, ANSI转义码默认是被禁用的, 可以通过额外设置来启用。

这种方法是在控制台中实现简单彩色文本的一种途径。在更复杂的情况下, 可以考虑使用第三方库, 如`colorama`等。