字符串

Python Reference

索引

字符串 (string) 是引号里的文本。每个字符 (character) 都有一个位置, 叫作它的索引 (index)。第一个索引是 `0`, 不是 `1`。

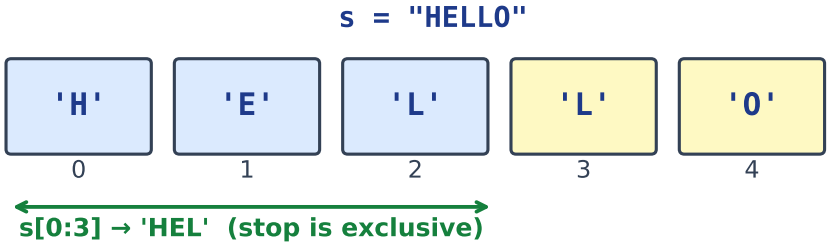
用方括号读取一个字符:

```
word = "Python"
print(word[0])    # P  (the first character)
print(word[2])    # t
print(len(word))  # 6  (how many characters)
```

- 计数从 `0` 开始, 所以最后一个索引是 `len(word) - 1`。
- 负数索引从末尾往回数:`word[-1]` 是最后一个字符。

```
word = "Python"
print(word[-1])   # n
print(word[-2])   # o
```

- 索引太大会产生错误 (error)(一个 `IndexError`)。



Exam tip: slice is half-open [start, stop); omit ends for defaults; step is optional

s[start:stop] 是一个半开区间的字符窗口

切片

切片 (slice) 取字符串的一部分。写成 `word[start:end]`。切片包含 `start`, 但**在** `end` **之前**停止。

```
word = "Python"
print(word[0:3])   # Pyt   (positions 0, 1, 2)
print(word[2:5])   # tho
```

- 省略 `start` 表示从 0 开始; 省略 `end` 表示一直到末尾。

```
word = "Python"
print(word[:3])    # Pyt
print(word[3:])    # hon
```

- 第三个数字是步长 (step)。`word[::-1]` 会反转 (reverse) 字符串。

```
print("Python"[::-1])   # nohtyP
```

字符串方法与长度

方法 (method) 是属于某个值的函数。你用点号来调用它:

```
name = "mei chen"
print(name.upper())     # MEI CHEN
print(name.title())     # Mei Chen
print(len(name))        # 8
```

字符串是不可变的 (immutable): 方法会返回一个**新**字符串, 绝不改变原始 (original) 字符串。

```
name = "mei"
print(name.upper())     # MEI  (the returned value)
print(name)             # mei  (the original is unchanged)
```

常用方法 (每个都返回一个新值):

方法	含义	例子	结果
`.upper()` / `.lower()`	改变大小写	`"Hi".lower()`	`hi`
`.strip()`	去掉两端空格	`" hi ".strip()`	`hi`
`.replace(a, b)`	替换文本	`"cat".replace("c", "b")`	`bat`
`.split(sep)`	拆成列表	`"a,b".split(",")`	`['a', 'b']`

用 + 连接字符串, 这叫作拼接 (concatenation):

```
first = "Mei"
last = "Chen"
print(first + " " + last)   # Mei Chen
```

f-字符串

f-字符串 (f-string) 用各个值来拼出文本。在引号前加 `f`, 再用 `{...}` 把值括起来。

```
name = "Mei"
age = 17
print(f"{name} is {age} years old")   # Mei is 17 years old
```

- 大括号里可以放任何表达式 (expression)。
- `{value:.2f}` 会四舍五入到 2 位小数 (decimal places)。

```
price = 9.5
print(f"Two cost {price * 2}")        # Two cost 19.0
print(f"Pi is about {3.14159:.2f}")   # Pi is about 3.14
```

常见错误

- 字符串不能原地修改:`s[0] = "x"` 会报错。要新建一个字符串。
- 索引从 `0` 开始; 最后一个字符是 `s[-1]`, 而 `s[len(s)]` 会越界。
- 切片 `s[a:b]` 包含 `a`, 但在 `b` **之前**停止。
- 字符串方法返回的是**新**字符串:`s.upper()` 不存下结果就等于没做。