

字符串操作

Ref Pseudocode

LENGTH、UCASE 与 LCASE

0478 只公布四个字符串 (string) 例程, 没有别的。其中三个在这里。

LENGTH(<identifier>) 返回字符串含有多少个字符 (character)—— 空格也算。

```
OUTPUT LENGTH("Happy Days")
OUTPUT LENGTH("")
```

UCASE(<identifier>) 返回大写形式, LCASE(<identifier>) 返回小写形式。两者都不改变原值 (original)—— 各自交回一个新值。

```
DECLARE Name : STRING
Name ← "Happy"
OUTPUT UCASE(Name)
OUTPUT LCASE(Name)
OUTPUT Name
```

最后那一行仍然输出 Happy: 要保留改动, 必须把结果赋回去。

```
DECLARE Name : STRING
Name ← "Happy"
Name ← UCASE(Name)
OUTPUT Name
```

比较时不必顾虑大小写

两边都套 UCASE, 是接受任意大小写输入的标准做法:

```
DECLARE Answer : STRING
Answer ← "yes"
IF UCASE(Answer) = "YES"
    THEN
        OUTPUT "Accepted"
ENDIF
```

常见错误

- 以为 UCASE(Name) 会改变 Name。请把结果赋回去。
- LENGTH 数的是字符, 不是单词。LENGTH("Happy Days") 是 10。

SUBSTRING

SUBSTRING(<identifier>, <start>, <length>) 从字符串中取出一段 —— 从位置 start 开始, 取 length 个字符。

```
OUTPUT SUBSTRING("Happy Days", 1, 5)
OUTPUT SUBSTRING("Happy Days", 7, 4)
```

△ 计数从 1 开始, 不是 0。"Happy Days" 的第一个字符在位置 1。

取一个字符, 以及取剩下的部分

```
DECLARE Word : STRING
Word ← "Computer"
OUTPUT "First letter: ", SUBSTRING(Word, 1, 1)
OUTPUT "Last letter: ", SUBSTRING(Word, LENGTH(Word), 1)
OUTPUT "Without the first: ", SUBSTRING(Word, 2, LENGTH(Word) - 1)
```

把 LENGTH 套进 SUBSTRING, 正是后两行对任意长度的词都成立的原因 —— 写死 8 的话, 词一变就错。

逐字符读取字符串

```
DECLARE Word : STRING
DECLARE Index : INTEGER
Word ← "Cat"
FOR Index ← 1 TO LENGTH(Word)
    OUTPUT SUBSTRING(Word, Index, 1)
NEXT Index
```

常见错误

- 从 0 开始。位置 1 才是第一个字符。
- 把第三个参数当成结束位置。它是一个个数 —— SUBSTRING(w, 2, 3) 从第二个字符起取三个。
- 要求的字符数超过字符串本身的长度。