YazLang Commands

YazLang Commands

Each YazLang command (found in a .yzy file) follows this pattern:

$$\mathtt{COMMAND}\ \ \mathtt{arg}_1\ \ \mathtt{arg}_2\ \ \dots\ \ \mathtt{arg}_n$$

EX: `RANGE 0 5 1`

Here there are 3 arguments

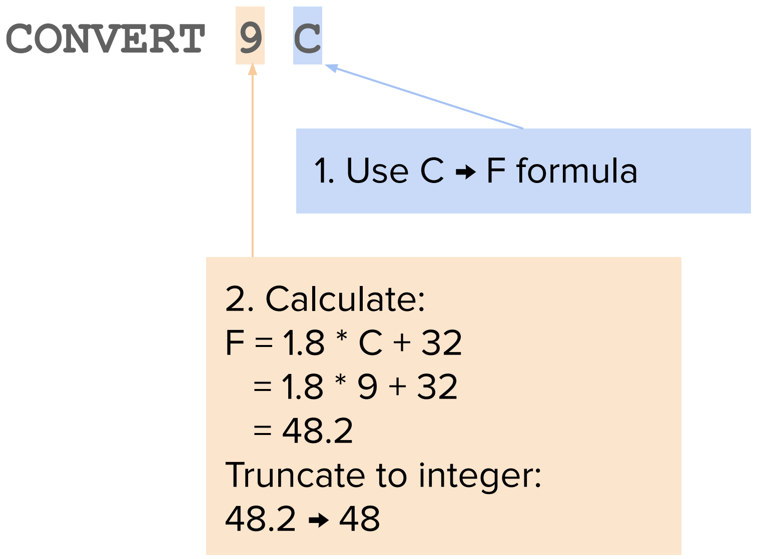
There are 3 commands that you should implement for YazLang:

- **CONVERT**
- **RANGE**
- **REPEAT**

CONVERT

- Followed by 2 arguments:
  - $\texttt{arg}_1$ - the original temperature value (integer)
  - $\texttt{arg}_2$ - the original temperature unit (C or F, case-insensitive)
  - EX: **`CONVERT 9 c`**
- Converts given temperature from Celsius to Fahrenheit, or vice versa, using the formulas:
  - $F = 1.8 * C + 32$ C → F formula
  - $C = (F - 32) / 1.8$ F → C formula
- Output: the converted temperature value (truncated to an integer) and converted unit (uppercase)
  - EX: **`48F`**

`CONVERT` Examples

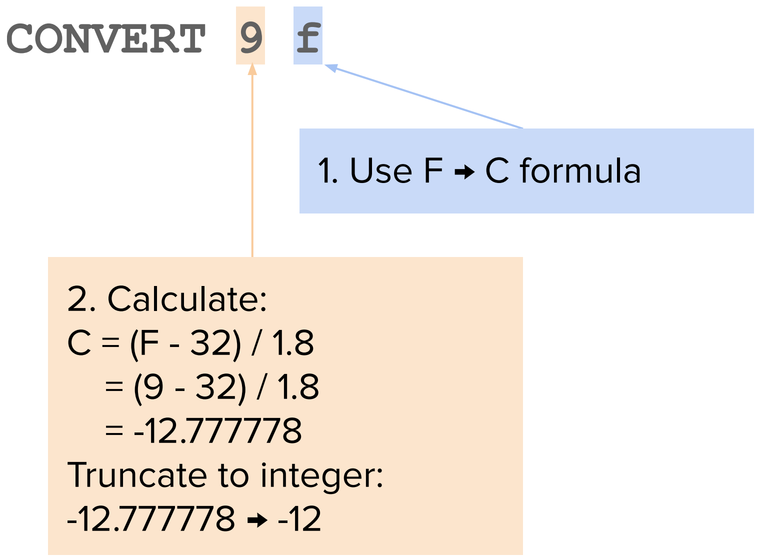


`CONVERT 9 C`

1. Use C → F formula

2. Calculate:
F = 1.8 * C + 32
= 1.8 * 9 + 32
= 48.2
Truncate to integer:
48.2 → 48

Output: `48F`



`CONVERT 9 f`

1. Use F → C formula

2. Calculate:
C = (F - 32) / 1.8
= (9 - 32) / 1.8
= -12.777778
Truncate to integer:
-12.777778 → -12

Output: `-12C`

RANGE

- Followed by 3 arguments:
  - $\texttt{arg}_1$ - the first number to *start* the range at
  - $\texttt{arg}_2$ - the first number to *end* the range at
  - $\texttt{arg}_3$ - the amount to increment by
  - EX: **`RANGE 0 5 1`**
- Output: the sequence of numbers starting from $\texttt{arg}_1$ (inclusive) and incrementing by $\texttt{arg}_3$ up until the value of $\texttt{arg}_2$ is reached/surpassed (exclusive)
  - EX: **`0 1 2 3 4`**
- Note: only valid ranges are printed out
  - **`RANGE 2 0 1`** has no output (since $\texttt{arg}_1$ >= $\texttt{arg}_2$ here)

`RANGE` Example: pg 1

```
RANGE -9 9 3
```

1. Print out the starting value **-9**

Output: `-9`

`RANGE` Example: pg 2

```
RANGE -9 9 3
```

2. Keep on incrementing the “current” value by **3**, until we have reached or surpassed the value **9**

Output: `-9  -6`

`RANGE` Example: pg 3

```
RANGE -9 9 3
```

2. Keep on incrementing the “current” value by **3**, until we have reached or surpassed the value **9**

Output: `-9 -6 -3`

RANGE Example: pg 4

```
RANGE -9 9 3
```

2. Keep on incrementing the “current” value by **3**, until we have reached or surpassed the value **9**

Output: `-9 -6 -3 0`

`RANGE` Example: pg 5

```
RANGE -9 9 3
```

2. Keep on incrementing the “current” value by **3**, until we have reached or surpassed the value **9**

Output: `-9 -6 -3 0 3`

RANGE Example: pg 6

```
RANGE -9 9 3
```

2. Keep on incrementing the “current” value by **3**, until we have reached or surpassed the value **9**

Output: `-9 -6 -3 0 3 6`

RANGE Example: pg 7

```
RANGE -9 9 3
```

3. Notice now that we have incremented up to the value of `arg`$_2$ (**9**). Remember that we don't want to print out `arg`$_2$ or any value greater than it, so this is our final output!

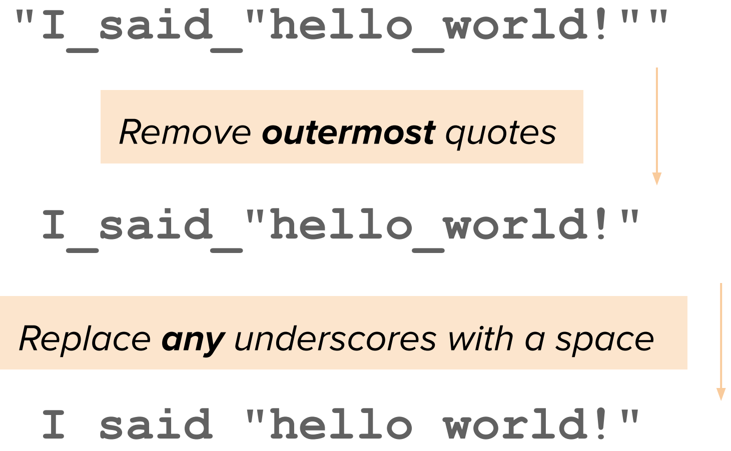
Output: `-9 -6 -3 0 3 6`

REPEAT

- Followed by an even # of arguments:
  - $\mathtt{arg}_{\mathtt{odd}}$ - the string to be printed
  - $\mathtt{arg}_{\mathtt{even}}$ - the # of times to print the specified string (always ≥0)
  - EX: **`REPEAT "a_" 5 "B" 0 "C" 2`**
- Output: each string argument repeated the number of times indicated by the following integer argument
  - EX: **`a a a a a CC`**
- Note: string arguments should have outermost quotation marks removed and any underscores replaced with a space (see next slide)

String Formatting

EX: **REPEAT "I_said_"hello_world!"" 1**



"I_said_"hello_world!""

*Remove **outermost** quotes*

I_said_"hello_world!"

*Replace **any** underscores with a space*

I said "hello world!"

Output: **I said "hello world!"**

REPEAT Example: pg 1

```
REPEAT "a_" 5 "B" 0 "C" 2 "D" 1
```

1. Output “a_” 5 times, with appropriate String formatting

Output: `a a a a a`

REPEAT Example: pg 2

```
REPEAT "a_" 5 "B" 0 "C" 2 "D" 1
```

2. Output “B” 0 times

Output: **a a a a a**

REPEAT Example: pg 3

```
REPEAT "a_" 5 "B" 0 "C" 2 "D" 1
```

2. Output “C” 2 times, with appropriate String formatting

Output: **a a a a a CC**

REPEAT Example: pg 4

```
REPEAT "a_" 5 "B" 0 "C" 2 "D" 1
```

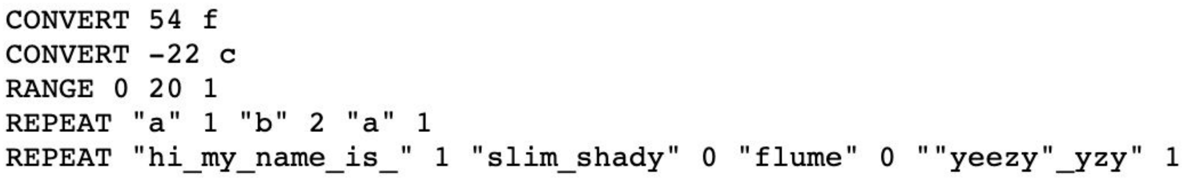
2. Output “D” 1 time, with appropriate String formatting

Output: **`a a a a a CCD`**

File I/O Example

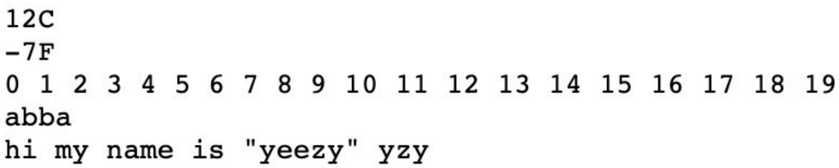
Try each of these commands out by hand and see if you get the same output as the one below!

Sample Input File: yaz.yzy



```
CONVERT 54 f
CONVERT -22 c
RANGE 0 20 1
REPEAT "a" 1 "b" 2 "a" 1
REPEAT "hi_my_name_is_" 1 "slim_shady" 0 "flume" 0 ""yeezy"_yzy" 1
```

Expected Output File: yaz-out.txt



```
12C
-7F
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
abba
hi my name is "yeezy" yzy
```