YazLang Commands
YazLang Commands

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

\[
\text{COMMAND } arg_1 \ arg_2 \ldots \ arg_n
\]

**EX:**  \text{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:
  - \( \text{arg}_1 \) - the original temperature value (integer)
  - \( \text{arg}_2 \) - 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:
  - \( F = 1.8 \times C + 32 \)  \( \text{C} \to \text{F} \text{ formula} \)
  - \( C = (F - 32) / 1.8 \)  \( \text{F} \to \text{C} \text{ formula} \)

- Output: the converted temperature value (truncated to an integer) and converted unit (uppercase)
  - **EX: 48 F**
CONVERT Examples

**CONVERT 9 C**

1. Use C → F formula
2. Calculate:
   \[ F = 1.8 \times C + 32 \]
   \[ = 1.8 \times 9 + 32 \]
   \[ = 48.2 \]

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:
  - \( \text{arg}_1 \) - the first number to \textit{start} the range at
  - \( \text{arg}_2 \) - the first number to \textit{end} the range at
  - \( \text{arg}_3 \) - the amount to increment by

- \textbf{EX: RANGE 0 5 1}

- Output: the sequence of numbers starting from \( \text{arg}_1 \) (inclusive) and incrementing by \( \text{arg}_3 \) up until the value of \( \text{arg}_2 \) is reached/surpassed (exclusive)
  - \textbf{EX: 0 1 2 3 4}

- Note: only valid ranges are printed out
  - \textbf{ RANGE 2 0 1} has no output (since \( \text{arg}_1 \geq \text{arg}_2 \) 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
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
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
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

Output: -9 -6 -3 0 3 6

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!
REPEAT

- Followed by an even # of arguments:
  - \( \text{arg}_{\text{odd}} \) - the string to be printed
  - \( \text{arg}_{\text{even}} \) - the # of times to print the specified string (always \( \geq 0 \))
    - \textbf{EX:} REPEAT "a_" 5 "B" 0 "C" 2

- Output: each string argument repeated the number of times indicated by the following integer argument
  - \textbf{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!"
Example: pg 1

Output: a a a a a a
Example: pg 2

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

2. Output “B” 0 times

Output: a 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 a CC
Example: pg 4

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

Output: a a a a a CCD

2. Output “D” 1 time, with appropriate String formatting
File I/O Example

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"_zy" 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
```