

# The Design Recipe

**Directions :** Write a function `split-tab` that takes in a cost and the number of people sharing the bill and splits the cost equally.

## Contract and Purpose Statement

Every contract has three parts...

;  
\_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
*function name domain range*  
;  
\_\_\_\_\_ *what does the function do?*

## Examples

Write some examples, then circle and label what changes...

( EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
*function name input(s) what the function produces*  
( EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
*function name input(s) what the function produces*

## Definition

Write the definition, giving variable names to all your input values...

(define ( \_\_\_\_\_ )  
\_\_\_\_\_ *function name variable(s)*  
\_\_\_\_\_ )  
*what the function does with those variable(s)*

**Directions :** Write a function `num-cube` that takes in a number and returns the cube of that number.

## Contract and Purpose Statement

Every contract has three parts...

;  
\_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
*function name domain range*  
;  
\_\_\_\_\_ *what does the function do?*

## Examples

Write some examples, then circle and label what changes...

( EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
*function name input(s) what the function produces*  
( EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
*function name input(s) what the function produces*

## Definition

Write the definition, giving variable names to all your input values...

(define ( \_\_\_\_\_ )  
\_\_\_\_\_ *function name variable(s)*  
\_\_\_\_\_ )  
*what the function does with those variable(s)*