

The Design Recipe

Directions: Getting a gym membership costs \$150, and then there's a \$45/month fee after that. Write a function `globo-gym` that takes in a number of months and produces the cost of a membership for that many months.

Contract and Purpose Statement

Every contract has three parts...

The diagram illustrates the components of a function signature. It consists of four horizontal lines. The first line is labeled 'function name' on the left, followed by a vertical colon ':'. The second line is labeled 'domain' in the center. The third line is labeled 'range' on the right, preceded by a right-pointing arrow '->'. The fourth line is labeled 'what does the function do?' in the center. The first three lines are grouped by a vertical brace on the left, and the entire group is enclosed in a large right curly bracket '}' on the right.

Examples □

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

(EXAMPLE ())

<u>function name</u>	<u>input(s)</u>	<u>what the function produces</u>
----------------------	-----------------	-----------------------------------

(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: The cost of a ride is a starting price of \$2.50, plus \$1.50/mile. Write a function `rideshare`, that takes in a number of miles and produces the cost of that ride.

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 *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)
)
```