

# Contracts

Name	Domain	Range	example
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	
;	:	↑	

# Contracts

Name	Domain	Range	example
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	
:	:	↑	

# Lesson 1

Circles Competition

Time: 5 minutes

	<i>Math</i>	<i>Circle of Evaluation</i>	<i>Scheme Code</i>
<i>Round 1</i>	$(1 + 2) - (3 * 7)$		
<i>Round 2</i>	$3 - (1 + 2)$		
<i>Round 3</i>	$3 - (1 + (5 * 6))$		
<i>Round 4</i>	$(1 + (5 * 6)) - 3$		

### Fast Functions!

Fill out the contract for each function, then try to write two examples and the definition by yourself.

; double : Number -> Number  
name domain range

(EXAMPLE ( double 5 ) ( \* 2 5 ) )

(EXAMPLE ( double 7 ) ( \* 2 7 ) )

(define ( double n ) ( \* 2 n ) )

; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name domain range

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )

(define ( \_\_\_\_\_ ) \_\_\_\_\_ )

; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )

(define ( \_\_\_\_\_ ) \_\_\_\_\_ )

; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )

(define ( \_\_\_\_\_ ) \_\_\_\_\_ )



# DESIGN RECIPE: SAMPLE

State the problem:

Make a circle (spot) of size 100, with the provided color

## Contract+Purpose Statement

Every contract has three parts:

; spot100 : String -> Image  
name Domain Range

; Makes a size 100 circle in a color  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( spot100 green ) (circle 100 "solid" green ) )  
the user types... ...which should become  
color

(EXAMPLE ( spot100 "blue" ) (circle 100 "solid" "blue" ) )  
the user types... ...which should become

## Function

Circle the changes in the EXAMPLEs, and name the variables.

Write the code, copying everything that isn't circled, and using names where you find variables!

(define ( spot100 color ) (circle 100 "solid" color ) )  
function name variable names ...and the computer does this

# DESIGN RECIPE: DOUBLE-RADIUS

State the problem:

## Contract+Purpose Statement

Every contract has three parts:

;  
; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range

;  
; \_\_\_\_\_  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
the user types... ...which should become

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
the user types... ...which should become

## Function

Circle the changes in the EXAMPLEs, and name the variables.

Write the code, copying everything that isn't circled, and using names where you find variables!

(define ( \_\_\_\_\_ ) \_\_\_\_\_ )

# DESIGN RECIPE: DOUBLE-WIDTH

State the problem:

## Contract+Purpose Statement

Every contract has three parts:

;  
; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range

;  
; \_\_\_\_\_  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
the user types... ...which should become

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
the user types... ...which should become

## Function

Circle the changes in the EXAMPLEs, and name the variables.

Write the code, copying everything that isn't circled, and using names where you find variables!

(define ( \_\_\_\_\_ ) \_\_\_\_\_ )



# DESIGN RECIPE: PAINT-JOB

State the problem:

## Contract+Purpose Statement

;  
; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range  
;  
; \_\_\_\_\_  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

(EXAMPLE ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

## Function

(define ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

# DESIGN RECIPE: TURBO-CHARGE

State the problem:

## Contract+Purpose Statement

;  
; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range  
;  
; \_\_\_\_\_  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

(EXAMPLE ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

## Function

(define ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)



# DESIGN RECIPE: PIMP

State the problem:

## Contract+Purpose Statement

; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range

; \_\_\_\_\_  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

(EXAMPLE ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

## Function

(define ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

## DEFINE-STRUCT

**Autos:**

; an auto is a \_\_\_\_\_

```
(define-struct auto (_____  
                    _____  
                    _____  
                    _____  
                    _____))
```

; a party is a \_\_\_\_\_

```
(define-struct party (_____  
                    _____  
                    _____))
```

# DESIGN RECIPE: RSVP

State the problem:

## Contract+Purpose Statement

; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range

; \_\_\_\_\_  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( \_\_\_\_\_ )  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

(EXAMPLE ( \_\_\_\_\_ )  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

## Function

(define ( \_\_\_\_\_ )  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

# DESIGN RECIPE: RELOCATE

State the problem:

## Contract+Purpose Statement

; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range

; \_\_\_\_\_  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( \_\_\_\_\_ )  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

(EXAMPLE ( \_\_\_\_\_ )  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

## Function

(define ( \_\_\_\_\_ )  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

## Dissecting a Demo: Ninja World

What changes?

---

---

---

---

## Ninja World:

; a world is a \_\_\_\_\_

(define-struct world (\_\_\_\_\_))

My constructor function is:

1) (How do you make a world?) \_\_\_\_\_

What is its contract? \_\_\_\_\_

My accessor function is:

2) (How do you get the dogX out of the world?)

\_\_\_\_\_

What is its contract? \_\_\_\_\_



# DESIGN RECIPE: UPDATE-WORLD (NINJA WORLD)

State the problem:

## Contract+Purpose Statement

; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range

; \_\_\_\_\_  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( \_\_\_\_\_ )

\_\_\_\_\_ )

(EXAMPLE ( \_\_\_\_\_ )

\_\_\_\_\_ )

## Function

(define ( \_\_\_\_\_ )

\_\_\_\_\_ )



## Review: define-struct

Last week we talked about a function that created new structs. For the structs below, what function would you use for each of the following?

---

; an auto is a String \_\_\_\_\_ Number \_\_\_\_\_ Number  
(define-struct auto (model hp rims color value))

Make an auto? \_\_\_\_\_

Get the model out of an auto? \_\_\_\_\_

Get the hp out of an auto? \_\_\_\_\_

---

; a party is a \_\_\_\_\_ String Number  
(define-struct team (location theme guests))

Make a team? \_\_\_\_\_

Get the city out of the team? \_\_\_\_\_

Get the sport out of the team? \_\_\_\_\_

Get the rank out of the team? \_\_\_\_\_

---

; a world is a Number  
(define-struct world (dogX))

What function would you use to:

Make a world? \_\_\_\_\_

Get the dogX out of the world? \_\_\_\_\_

# GAME DESIGN

*“Start Simple, Get Complex”*

**Draw a rough sketch of your game in action**



**What images will you need for your game?**

Background	

**List everything that has changed, and the datatype you will use to represent it**

<b>Changed (position? score? color? costume?)</b>	<b>Datatype (number? string? image? boolean?)</b>

; a world is a \_\_\_\_\_

(define-struct world ( \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_ ))

**My constructor function is...**

; make-world : \_\_\_\_\_ → World

**My accessor functions are...**

.  
; \_\_\_\_\_  
.  
; \_\_\_\_\_  
.  
; \_\_\_\_\_  
.  
; \_\_\_\_\_  
.  
; \_\_\_\_\_



(0, 480)

**START**

(640,480)



(0, 0)

(640, 0)

At the start of my game, this is where everything is:

Object (top to bottom of stack)	Position (x, y)
Background	

```

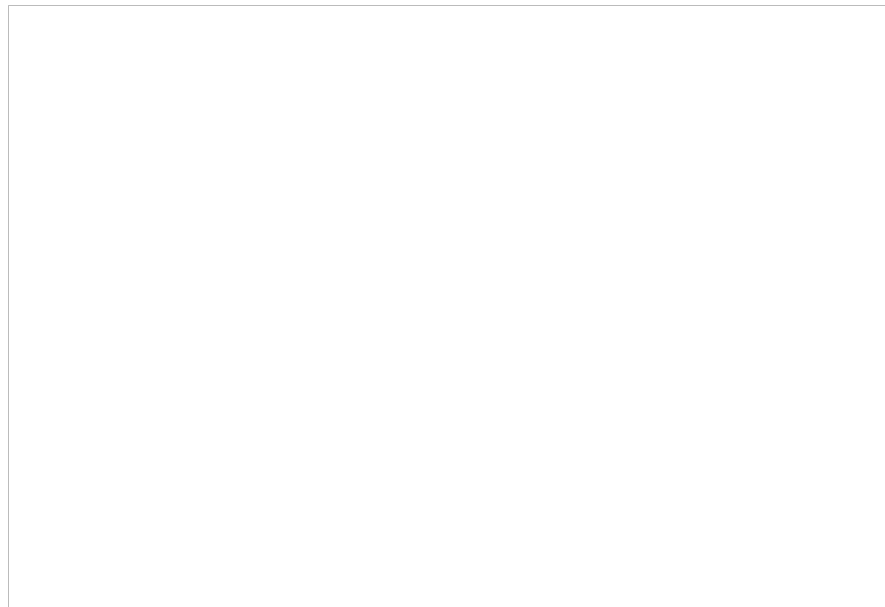
(define START(make-world _____
                        _____
                        _____
                        _____
                        _____))

```

(0, 480)

**NEXT**

(640,480)



(0, 0)

(640, 0)

A split second later, this is where everything is:

Object (top to bottom of stack)	Position (x, y)
Background	

(define NEXT (make-world \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_))



# DRAW-WORLD

## Contract

; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_

## Using put-image

(define ( \_\_\_\_\_ )

(put-image \_\_\_\_\_

\_\_\_\_\_

(put-image \_\_\_\_\_

\_\_\_\_\_

(put-image \_\_\_\_\_

\_\_\_\_\_

(put-image \_\_\_\_\_

\_\_\_\_\_

BACKGROUND \_\_\_\_\_

# DESIGN RECIPE: UPDATE-WORLD

State the problem (What changes?):

## Contract+Purpose Statement

;  
; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range  
;  
; \_\_\_\_\_  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

(EXAMPLE ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

## Function

(define ( \_\_\_\_\_ )

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)



# DESIGN RECIPE

## State the Problem

For each keypress in the Ninja World game, show how (keypress START <key>) should change your world.

## Contract+Purpose Statement

⌈ \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Ranges

## Give Examples

(EXAMPLE (keypress START \_\_\_\_\_))

(make-world \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

(EXAMPLE (keypress START \_\_\_\_\_))

(make-world \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_)

(define ( \_\_\_\_\_ )

(cond

[( \_\_\_\_\_ )

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_ ]

[( \_\_\_\_\_ )

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_ ]))

# DESIGN RECIPE

## State the Problem

For each keypress in your game, show how (keypress START <key>) should change your world.

## Contract+Purpose Statement

⤵ \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Ranges

## Give Examples

(EXAMPLE (keypress START \_\_\_\_\_))

(make-world \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_))

---

(EXAMPLE (keypress START \_\_\_\_\_))

(make-world \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_))

---

```
(EXAMPLE (keypress START _____)
  (make-world _____
    _____
    _____
    _____
    _____))
```

---

```
(define (_____ _____)
  (cond
    [(_____ )
     _____
     _____
     _____
     _____]
    [(_____ )
     _____
     _____
     _____
     _____]
    [(_____ )
     _____
     _____
     _____
     _____]))
```







TEST	RESULT
	(make-world _____ _____ _____ _____ _____) )
	(make-world _____ _____ _____ _____ _____) )
	(make-world _____ _____ _____ _____ _____) )
	(make-world _____ _____ _____ _____ _____) )





## Distance:

The Player is at (4, 2) and the Target is at (0, 5).

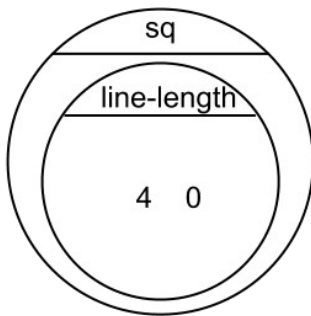
Distance takes in the player-x, player-y, character-x and character-y.

Use the formula below to fill in the EXAMPLE:

$$\sqrt{(\text{line-length } 4 \ 0)^2 + (\text{line-length } 2 \ 5)^2}$$

---

Convert it into a Circle of Evaluation. (We've already gotten you started!)



---

Convert it into Racket code:

(EXAMPLE ( \_\_\_\_\_ ))

( \_\_\_\_\_ )

\_\_\_\_\_ )

## Design Recipe: distance

Write a function `distance`, which takes FOUR inputs:

- `px`: The x-coordinate of the player
- `py`: The y-coordinate of the player
- `cx`: The x-coordinate of another game character
- `cy`: The y-coordinate of another game character

It should return the distance between the two, using the Distance formula:

$$\text{Distance} = ((\text{line-length } px \text{ } cx)^2 + (\text{line-length } py \text{ } cy)^2)$$

### Contract+Purpose Statement

```
; _____ : _____ -> _____  
   name                Domain                Range  
  
; _____  
   What does the function do?
```

### Give Examples

```
(EXAMPLE  
  ( _____ )  
_____)
```

```
(EXAMPLE  
  ( _____ )  
_____)
```

### Function Header

```
(define ( _____ )  
         function name      variable names  
_____)
```



TEST	RESULT
	(make-world _____ _____ _____ _____ _____) _____)
	(make-world _____ _____ _____ _____ _____) _____)
	(make-world _____ _____ _____ _____ _____) _____)
	(make-world _____ _____ _____ _____ _____) _____)





# DESIGN RECIPE

State the problem:

## Contract+Purpose Statement

Every contract has three parts:

;  
; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range

;  
; \_\_\_\_\_  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
the user types... ...which should become

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
the user types... ...which should become

## Function

Circle the changes in the EXAMPLEs, and name the variables.

Write the code, copying everything that isn't circled, and using names where you find variables!

(define ( \_\_\_\_\_ ) \_\_\_\_\_ )

# DESIGN RECIPE

State the problem:

## Contract+Purpose Statement

Every contract has three parts:

; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range

; \_\_\_\_\_  
What does the function do?

## Give Examples

On the computer, write an example of your function in action, using EXAMPLE.

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
the user types... ...which should become

(EXAMPLE ( \_\_\_\_\_ ) \_\_\_\_\_ )  
the user types... ...which should become

## Function

Circle the changes in the EXAMPLEs, and name the variables.

Write the code, copying everything that isn't circled, and using names where you find variables!

(define ( \_\_\_\_\_ ) \_\_\_\_\_ )

TEST	RESULT
	(make-world _____ _____ _____ _____ _____) )
	(make-world _____ _____ _____ _____ _____) )
	(make-world _____ _____ _____ _____ _____) )
	(make-world _____ _____ _____ _____ _____) )