Name: _____

Bootstrap:2

www.bootstrapworld.org

Class: _____

Lesson 1

	Racket Code	Pyret Code
Numbers	(define AGE 14) (define A-NUMBER 0.6) (define SPEED -90)	AGE = 14 A-NUMBER = 0.6 SPEED = -90 Two of your own:
Strings	(define CLASS “Bootstrap”) (define PHRASE "Coding is fun!") (define A-STRING "2500")	CLASS = “Bootstrap” PHRASE = “Coding is fun!” A-STRING = “2500” Two of your own:

	(define SHAPE
	  (triangle 40 "outline" "red"))
	
	(define OUTLINE
	  (star 80 "solid" "green"))
	
	(define SQUARE
	  (rectangle 50 50 "solid" "blue"))
		SHAPE =
		  triangle(40, "outline", "red")
		
		OUTLINE =
		  star(80, “solid”, “green”)
		
		SQUARE =
		  rectangle(50, 50, “solid”, “blue”)
		
		One of your own:

Booleans	(define BOOL true) (define BOOL2 false)	BOOL = true One of your own:

Functions	; double : Number -> Number
	; Given a number, multiply by
	; 2 to double it
	
	(EXAMPLE (double 5) (* 2 5)
	(EXAMPLE (double 7) (* 2 7))
	
	(define (double n) (* 2 n))
		# double : Number -> Number
		# Given a number, multiply by
		# 2 to double it
		
		examples:
		  double(5) is 2 * 5
		  double(7) is 2 * 7
		end
		
		fun double(n):
		  2 * n
		end

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

examples:
   double(5) is 2 * 5
   double(7) is 2 * 7
end

fun double(n):
    2 * n
end
```

n

n

```
# _____ : _____ -> _____
  name    domain   range

examples:
   _____(_____) is _____
   _____(_____) is _____
end

fun _____(_____):

    _____

end
```

Fast Functions!

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

```
# _____:_____ -> _____
  name  domain  range

examples:

  _____(_____) is _____

  _____(_____) is _____

end

fun _____(_____):

  _____

end
```

```
# _____:_____ -> _____
  name  domain  range

examples:

  _____(_____) is _____

  _____(_____) is _____

end

fun _____(_____):

  _____

end
```

Fast Functions!

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

```
# _____:_____ -> _____
  name  domain  range

examples:

  _____(_____) is _____

  _____(_____) is _____

end

fun _____(_____):

  _____

end
```

```
# _____:_____ -> _____
  name  domain  range

examples:

  _____(_____) is _____

  _____(_____) is _____

end

fun _____(_____):

  _____

end
```

Bug Hunting: Pyret Edition

#1	`SECONDS = (7)`
	
	`STRING = my string`
		_____
		
		_____

#2	`SHAPE1 = circle(50 “solid” “blue”)`
	
	`SHAPE2 = triangle(75, outline, yellow)`
		_____
		
		_____

#3	`# triple : Number -> Number`
	`# Multiply a given number by`
	`# 3 to triple it`
	
	`examples:`
	`  triple(5) = 3 * 5`
	`  triple(7) = 3 * 7`
	`end`	

#4	`fun triple(n):` `  3 * n`	

#5	`# ys : Number -> Number`
	`# Given a number, create a solid`
	`# yellow star of the given size`
	
	`examples:`
	`  ys(99) is star(99, “solid”, “yellow”)`
	`  ys(33) is star(99, “solid”, “yellow”)`
	
	`ys(size):`
	`  star(size “solid” “yellow”)`
	`end`	

Lesson 2

Word Problem: double-radius

Write a function *double-radius*, which takes in a radius and a color. It produces an outlined circle of whatever color was passed in, whose radius is twice as big as the input.

Contract+Purpose Statement

Every contract has three parts:

```
# ____________ : ____________________________ -> ___________
    name                        Domain                  Range

# ____________________________________________________________________
                        What does the function do?
```

Give Examples

Write examples of your function in action

```
examples:
    ____________(____________________)        is
        the user types...

    ____________________________________________________________
                ...which should become

    ____________(____________________)        is
        the user types...

    ____________________________________________________________
                ...which should become
end
```

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!

```
fun ________________(________________):

    ____________________________________________________________
end
```

Word Problem: double-width

Write a function *double-width*, which takes in a number (the length of a rectangle) and produces a rectangle whose width is twice the given length.

Contract+Purpose Statement

Every contract has three parts:

```
# ____________ : ____________________________ -> ___________
    name                      Domain                   Range

# ____________________________________________________________________
                        What does the function do?
```

Give Examples

Write examples of your function in action

```
examples:
    ____________(____________________)        is
        the user types...

    ____________________________________________________________
                ...which should become

    ____________(____________________)        is
        the user types...

    ____________________________________________________________
                ...which should become
end
```

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!

```
fun ________________(________________):

    ____________________________________________________________
end
```

Word Problem: next-position

Write a function *next-position*, which takes in two numbers (an x and y-coordinate) and returns a Coord, increasing the x-coordinate by 5 and decreasing the y-coordinate by 5.

Contract+Purpose Statement

Every contract has three parts:

```
# ______________ : ______________________________ -> ___________
    name                          Domain                    Range

# ____________________________________________________________________
                        What does the function do?
```

Give Examples

Write examples of your function in action

```
examples:
    ______________(____________________)        is
        the user types...

    ____________________________________________________________
                ...which should become

    ______________(____________________)        is
        the user types...

    ____________________________________________________________
                ...which should become
end
```

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!

```
fun ________________(________________):

    ____________________________________________________________
end
```

Data Structure

```
# a Car is a model, hp, rims, color, and price
data Car:
    | car(_____________________________________________
          _____________________________________________
          _____________________________________________
          _____________________________________________
          _____________________________________________)
end
```

To make examples of this structure, I would write:

car1 = _____

car2 = _____

To access the fields of **car2**, I would write:

_____

_____

_____

_____

_____

Lesson 3

Data Structure

```
# a Party is a location, theme, and number of guests
data Party:
    |   party(_____________________________________________
              _____________________________________________
              _____________________________________________)
end
```

To make examples of this structure, I would write:

party1 = _____

party2 = _____

To access the fields of **party2**, I would write:

_____

_____

_____

Word Problem: paint-job

Write a function called *paint-job* which takes in a Car and a color, and gives back a new Car that is mostly the same as the original, but now has the given color.

Contract+Purpose Statement

```
# ________________ : ________________________________ -> ___________

# ____________________________________________________________________
```

Give Examples

```
examples:

    _____________(__________________)       is

                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
    _____________(__________________)       is

                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
end
```

Function

```
fun ___________________(__________________):

                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
end
```

Word Problem: turbo-charge

Write a function called *turbo-charge,* which takes in a Car, and gives back a Car that has 20 more horsepower.

Contract+Purpose Statement

```
# ________________ : ________________________________ -> ___________

# ____________________________________________________________________
```

Give Examples

```
examples:

    _____________(__________________)        is

                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
    _____________(__________________)        is

                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
end
```

Function

```
fun ___________________(__________________):

                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
                ____________________________________
end
```

Lesson 4

Word Problem: update-world (Ninja World)

Contract+Purpose Statement

```
# _____ : _____ -> _____

# _____
```

Give Examples

```
examples:

     _____(_____)      is

               _____
               _____
               _____
               _____
               _____
     _____(_____)      is

               _____
               _____
               _____
               _____
               _____
end
```

Function

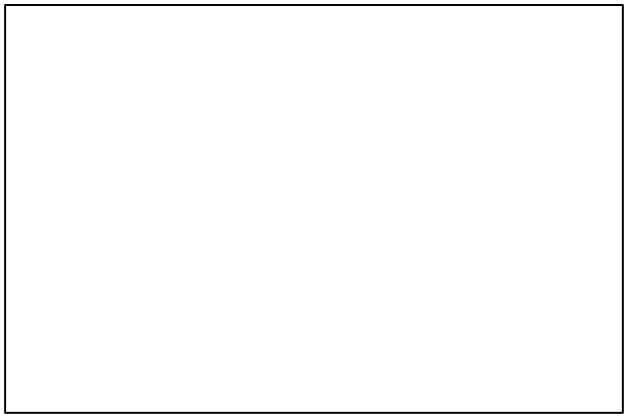
```
fun _____(_____):

               _____
               _____
               _____
               _____
               _____
end
```

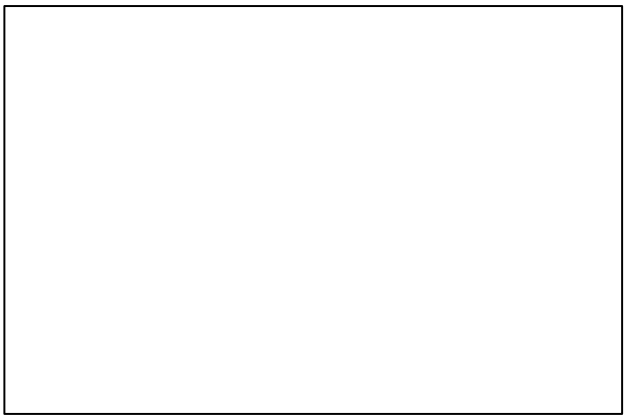
GAME DESIGN

"Start Simple, Get Complex"

Draw a rough sketch of your game when it begins, and another sketch just a moment later



A sketch at the START of the game…



A sketch for the very NEXT moment…

What images will you need for your game? Name them in the 1st column, and describe them in the 2nd

BACKGROUND	

List everything that has changed from one sketch to the other. What datatype will represent it?

Changed (position, score, color, costume…)	**Datatype** (Number, String, Image, Boolean…)

Data Structures

```
# a world is a _____________________________________
data World:
    | world(_____________________________________
            _____________________________________
            _____________________________________
            _____________________________________
            _____________________________________)
end
```

To make example worlds that represent my START and NEXT sketches from page 17, I would write…

START = _____

NEXT = _____

To access the fields of **START**, I would write:

_____

_____

_____

_____

_____

Lesson 5

Word Problem: draw-world

Contract

_____ : _____ -> _____

Definition

```
fun _____(_____) :
  put-image(_____
            _____
            _____
            _____
            _____
            _____
            _____
            _____
            _____
            _____
            _____
            _____
            _____
            _____
            _____
            _____)
end
```

Word Problem: update-world

State the problem (What changes?):

Contract+Purpose Statement

_____ : _____ -> _____
_____

Give Examples

```
examples:
     _____(_____)       is

                _____________________________________
                _____________________________________
                _____________________________________
                _____________________________________
                _____________________________________
     _____(_____)       is

                _____________________________________
                _____________________________________
                _____________________________________
                _____________________________________
                _____________________________________
end
```

Function

```
fun _____(_____):

                _____________________________________
                _____________________________________
                _____________________________________
                _____________________________________
                _____________________________________
end
```

Lesson 6

When *this* key is pressed…	..*this* field of the new world...	…changes by…

Word Problem: keypress (Ninja World)

State the Problem

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

Contract+Purpose Statement

```
# keypress : World String -> World

# Given a world and a key, produce a new world with NinjaCat's position
# moved by 10 pixels, depending on which arrow key was pressed
```

Give Examples

```
examples:
    keypress(START, "up")    is
                world(START.dogX, START.coinX, START.catX, START.catY + 10)

    keypress(START, "down")  is
                world(START.dogX, START.coinX, START.catX, START.catY - 10)

    keypress(NEXT, "left")   is
                world(NEXT.dogX, NEXT.coinX, NEXT.catX - 10, NEXT.catY)

    keypress(NEXT, "right")  is
                world(NEXT.dogX, NEXT.coinX, NEXT.catX + 10, NEXT.catY)
end
```

```
fun keypress(w, key) :

    ask:
     | string-equal(key, “up”) then:

          world(w.dogX, w.coinX, w.catX, w.catY + 10)

     | string-equal(key, “down”) then:

          world(w.dogX, w.coinX, w.catX, w.catY + 10)

     | string-equal(key, “left”) then:

          world(w.dogX, w.coinX, w.catX - 10, w.catY)

     | string-equal(key, “right”) then:

          world(w.dogX, w.coinX, w.catX + 10, w.catY)

     | otherwise: w

    end
end
```

Word Problem: keypress (My game)

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

: ->

#

Give Examples

examples:

keypress(START, _____) is

_____

_____

_____

_____

_____

keypress(START, _____) is

_____

_____

_____

_____

_____

keypress(START, _____) is

_____

_____

_____

_____

_____

end

```
fun _____(_____)
    ask:
        | _____ then:
          _____
        | _____ then:
          _____
        | _____ then:
          _____
        | _____ then:
          _____
        | _____ then:
          _____
        | _____ then:
          _____
        | _____ then:
          _____
        |_____
    end
end
```

Word Problem: red-shape

Write a function *red-shape*, which takes in the name of a shape (such as "circle", "triangle", "star", or "rectangle"), and draws that shape. All shapes should be solid and red, and can be whatever size you choose.

_____ : _____ -> _____

_____

Give Examples

```
examples:
_____(_____) is _____
_____(_____) is _____
_____(_____) is _____
_____(_____) is _____
end
```

Function

```
fun _____(_____):
    ask:
        | _____ then:
          _____
        | _____ then:
          _____
        | _____ then:
          _____
        | _____ then:
          _____
    end
end
```

Lesson 7

Word Problem: strong-password

Websites have strict password requirements. Write a function *strong-password*, which takes in a username and password, and checks to make sure they aren't the same, and then checks the string-length of the password to make sure it is greater than 8 characters. The function should return a message to the user letting them know if their password is strong enough.

_____ : _____ -> _____

_____

Give Examples

```
examples:

    _____(_____) is

        _____

    _____(_____) is

        _____

    _____(_____) is

        _____
end
```

Function

```
fun _____(_____):
    ask:
        | _____ then:

          _____

        | _____ then:

          _____

        | otherwise: _____
    end
end
```

Building Your Helper Functions

is-off-right :_____ -> _____

examples:

_____(_____) is

_____

_____(_____) is

_____

end

fun _____(_____):

_____

end

---

is-off-left :_____ -> _____

examples:

_____(_____) is

_____

_____(_____) is

_____

end

fun _____(_____):

_____

end

```
# _____:_____ -> _____
examples:
  _____(_____) is
          _____
  _____(_____) is
          _____
end
fun _____(_____):

  _____
end
```

```
# _____:_____ -> _____
examples:
  _____(_____) is
          _____
  _____(_____) is
          _____
end
fun _____(_____):

  _____
end
```

Using Helpers inside `update-world`:

How does the World structure change when....?

TEST	RESULT
	world(_____ _____ _____ _____ _____)
	world(_____ _____ _____ _____ _____)
	world(_____ _____ _____ _____ _____)
	world(_____ _____ _____ _____ _____)

TEST	RESULT
	world(_____ _____ _____ _____ _____)
	world(_____ _____ _____ _____ _____)
	world(_____ _____ _____ _____ _____)
	world(_____ _____ _____ _____ _____)

Using Helpers inside `draw-world`:

What changes the *appearance* of your game?

TEST	RESULT
	put-image(_____ put-image(_____ put-image(_____ put-image(_____ put-image(_____
	put-image(_____ put-image(_____ put-image(_____ put-image(_____ put-image(_____
	put-image(_____ put-image(_____ put-image(_____ put-image(_____ put-image(_____
	put-image(_____ put-image(_____ put-image(_____ put-image(_____ put-image(_____

TEST	RESULT
	put-image(_____ put-image(_____ put-image(_____ put-image(_____ put-image(_____
	put-image(_____ put-image(_____ put-image(_____ put-image(_____ put-image(_____
	put-image(_____ put-image(_____ put-image(_____ put-image(_____ put-image(_____
	put-image(_____ put-image(_____ put-image(_____ put-image(_____ put-image(_____

Lesson 8

Word Problem: line-length

Write a function called *line-length*, which takes in two numbers and returns the difference between them. It should always subtract the smaller number from the bigger one.

Contract+Purpose Statement

```
# _____ : _____ -> _____

# _____
```

Give Examples

```
examples:
    _____(_____)       is

    _____

    _____(_____)       is

    _____
end
```

Function Header

```
fun _____(_____):
```

function name — variable names

```
    _____:
```


```
    end
end
```

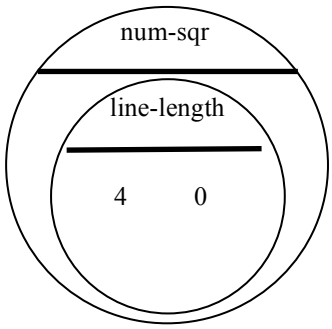
Distance:

The Player is at **(4, 2)** and the Target is at **(0, 5)**.
Distance takes in the player's x, player's y, character's x and character's y.

Use the formula below to fill in the EXAMPLE:

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

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



Convert it into Pyret code:

Word Problem: 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:

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

Contract+Purpose Statement

_____ : _____ -> _____

_____

Give Examples

Write examples of your function in action

```
examples:
    _____(_____)        is

    _____

    _____(_____)        is

    _____
end
```

Function

```
fun _____(_____):

    _____

    _____

end
```

Word Problem: is-collision

Write a function *is-collision*, 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 true if the coordinates of the player are within **50 pixels** of the coordinates of the other character. Otherwise, false.

Contract+Purpose Statement

_____ : _____ -> _____

_____

Give Examples

Write examples of your function in action

```
examples:
    _____(_____)      is
    _____
    _____

    _____(_____)      is
    _____
    _____
end
```

Function

```
fun _____(_____):
    _____
    _____
end
```

Supplemental

DESIGN RECIPE

Contract+Purpose Statement

Every contract has three parts:

_____ : _____ -> _____

name Domain Range

_____

What does the function do?

Give Examples

Write examples of your function in action

```
examples:
    _____(_____)        is
```

the user types…

_____

…which should become

```
    _____(_____)        is
```

the user types…

_____

…which should become

```
end
```

Function

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

```
fun _____(_____):


    _____
end
```

DESIGN RECIPE

Contract+Purpose Statement

Every contract has three parts:

_____ : _____ -> _____

name Domain Range

_____

What does the function do?

Give Examples

Write examples of your function in action

```
examples:
    _____(_____)        is
```

the user types…

_____

…which should become

```
    _____(_____)        is
```

the user types…

_____

…which should become

```
end
```

Function

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

```
fun _____(_____):


    _____
end
```

Contracts

Name	Domain	Range	example
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	

Contracts

Name	Domain	Range	example
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	
#	:	→	