Workbook v0.9

Brought to you by the Bootstrap team:
- Emma Youndtsmith
- Emmanuel Schanzer
- Kathi Fister
- Joe Politz
- Shriram Krishnamurthi

Visual Design: Colleen Murphy

Bootstrap is licensed under a Creative Commons 3.0 Unported License. Based on a work from www.BootstrapWorld.org. Permissions beyond the scope of this license may be available at schanzer@BootstrapWorld
## Unit 1

<table>
<thead>
<tr>
<th>Numbers</th>
<th>Racket Code</th>
<th>Pyret Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>(define AGE 14)</td>
<td>AGE = 14</td>
<td></td>
</tr>
<tr>
<td>(define A–NUMBER 0.6)</td>
<td>A–NUMBER = 0.6</td>
<td></td>
</tr>
<tr>
<td>(define SPEED -90)</td>
<td>SPEED = -90</td>
<td></td>
</tr>
</tbody>
</table>

Two of your own:

<table>
<thead>
<tr>
<th>Strings</th>
<th>Racket Code</th>
<th>Pyret Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>(define CLASS “Bootstrap”)</td>
<td>CLASS = “Bootstrap”</td>
<td></td>
</tr>
<tr>
<td>(define PHRASE &quot;Coding is fun!&quot;)</td>
<td>PHRASE = “Coding is fun!”</td>
<td></td>
</tr>
<tr>
<td>(define A–STRING &quot;2500&quot;)</td>
<td>A–STRING = “2500”</td>
<td></td>
</tr>
</tbody>
</table>

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:

(define BOOL true)

(define BOOL2 false)

BOOL = true

One of your own:

; 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

examples:

double (5) is 2 * 5

double (7) is 2 * 7

end

fun double (n):

2 * n

end

# double : Number

examples:

double (5) is ________________

double (7) is ________________

end

fun double (n):

______________________________________________________

end
**Fast Functions!**

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

<table>
<thead>
<tr>
<th>name</th>
<th>domain</th>
<th>range</th>
</tr>
</thead>
</table>

**examples:**

\[
\text{___________(________)} \text{ is } \text{____________________}
\]

\[
\text{___________(________)} \text{ is } \text{________________}
\]

end

fun \text{___________(_________________):}

\[
\text{______________________________________________________}
\]

end

<table>
<thead>
<tr>
<th>name</th>
<th>domain</th>
<th>range</th>
</tr>
</thead>
</table>

**examples:**

\[
\text{___________(________)} \text{ is } \text{____________________}
\]

\[
\text{___________(________)} \text{ is } \text{________________}
\]

end

fun \text{___________(_________________):}

\[
\text{______________________________________________________}
\]

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

<table>
<thead>
<tr>
<th>#</th>
<th>name</th>
<th>domain</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td>examples:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

fun  

end

<table>
<thead>
<tr>
<th>#</th>
<th>name</th>
<th>domain</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td>examples:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

fun  

end
<table>
<thead>
<tr>
<th>#</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
</table>
| #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 | |
| #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”) | |
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

```
xamples:

________________________(_______________________)

the user types...

is __________________________

...which should become

________________________(_______________________)

the user types...

is __________________________

...which should become

dend
```

**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 ______________________(_______________________) :


dend
```
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:

<table>
<thead>
<tr>
<th># ______________________________</th>
<th>Domain</th>
<th>→</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
#

What does the function do?

**Give Examples**

Write examples of your function in action

**examples:**

\( \text{_______________________________} \)

the user types...

\( \text{is } \text{_______________________________} \)

...which should become

\( \text{_______________________________} \)

the user types...

\( \text{is } \text{_______________________________} \)

...which should become

\( \text{end} \)

**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 DeliveryState, 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:

__________ (__________________________)

the user types...

is ___________________________

...which should become

__________ (__________________________)

the user types...

is ___________________________

...which should become

e nd

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 ______________ (__________________________):
# A CakeType is a flavor, layers, & is-iceCream

data CakeType:
    | cake(__________________________________________
    |
    |
    |

eend

To make instances of this structure, I would write:

cake1 = __________________________________________

cake2 = __________________________________________

To access the fields of cake2, I would write:

_______________________________________

_______________________________________

_______________________________________
Word Problem: taller-than

Write a function called taller-than, which consumes two CakeTypes, and produces true if the number of layers in the first CakeType is greater than the number of layers in the second.

Contract + Purpose Statement

# ______________ : ___________________________ → __________
# __________________________

Give Examples

Write examples of your function in action

examples:

_____________(_______________________)
the user types...

is_________________________

...which should become

_____________(_______________________)
the user types...

is_________________________
end

...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!

fun ______________________(_______________________) :

___________________________________________________

end
Word Problem: will-melt

Write a function called will-melt, which takes in a CakeType and a temperature, and returns true if the temperature is greater than 32 degrees, AND the CakeType is an ice cream cake.

Contract+Purpose Statement

# __________________________ : ___________________________ → _______

# __________________________

Give Examples

Write examples of your function in action

examples:

__________ (__________________)  

the user types...

is ____________________________  

...which should become

__________ (__________________)  

the user types...

is ____________________________  

...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 __________________________ (__________________) :


function

end
Vocabulary Practice

Below is a new structure definition:

```plaintext
data MediaType:
    |   book(
        title :: String,
        author :: String,
        pubyear :: Number)
end
```

# an example book:

Fill in the blanks below with the vocabulary term that applies to each name. Here are the terms to choose from:

- contract  - example
- header    - field
- datatype   - instance
- constructor - data block
- name       - purpose

author is a ____________________________

book is a _____________________________

MediaType is a __________________________

book1 is a _____________________________

title is a _____________________________

data ... end is a _____________________________
Identifying Animation Data Worksheet: Sunset

Draw a sketch for three distinct moments of the animation

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG...)</th>
<th>Datatype (Number, String, Image, Boolean...)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(worksheet continues on the next page)
Define the Data Structure

# a ________State is _________________________

data ________State:

| ________ (__________________________________ |

| ____________________________________ |

| ____________________________________ |

end

Make a sample instance for each sketch from the previous page:

__________ = __________________________________________

__________ = __________________________________________

__________ = __________________________________________

__________ = __________________________________________
Word Problem: draw-state

Write a function called draw-state, which takes in a SunsetState and returns an image in which the sun (a circle) appears at the position given in the SunsetState. The sun should be behind the horizon (the ground) once it is low in the sky.

Contract+Purpose Statement

# draw-state : ______________________________ → Image

Write an expression for each piece of your final image

SUN =
GROUND =
SKY =

Write the draw-state function, using put-image to combine your pieces

fun ______________________________(____________________________) :

_____________________________________________________________

_____________________________________________________________

_____________________________________________________________

end
Word Problem: next-state-tick

Write a function called `next-state-tick`, which takes in a SunsetState and returns a SunsetState in which the new x-coordinate is 8 pixels larger than in the given SunsetState and the y-coordinate is 4 pixels smaller than in the given SunsetState.

Contract+Purpose Statement

```plaintext
# ___________ : ___________________________ → ____________
# ____________________________
```

Give Examples

Write examples of your function in action

examples:

```plaintext
_____________ (_______________________)

the user types...

is _____________________________

...which should become

_____________ (_______________________)

the user types...

is _____________________________

end ...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!

```plaintext
fun ___________________________ (_______________________) :


end
```
Identifying Animation Data Worksheet

Draw a sketch for three distinct moments of the animation

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG...)</th>
<th>Datatype (Number, String, Image, Boolean...)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Define the Data Structure

# a _______State is __________________________
data _______State:

| _______ (_________________________________
| _______________________________________
| _______________________________________
| _______________________________________
end

Make a sample instance for each sketch from the previous page:

_______ = __________________________________

_______ = __________________________________

_______ = __________________________________

_______ = __________________________________
## Identifying Animation Data Worksheet

Draw a sketch for three distinct moments of the animation

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

### What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG...)</th>
<th>Datatype (Number, String, Image, Boolean...)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(worksheet continues on the next page)
Define the Data Structure

# a _________State is _______________________

data _________State:

| _________(___________________________

| _________________________________

| _________________________________

| _________________________________

end

Make a sample instance for each sketch from the previous page:

_________ = __________________________________

_________ = __________________________________

_________ = __________________________________

_________ = __________________________________
Identifying Animation Data Worksheet

Draw a sketch for three distinct moments of the animation

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG…)</th>
<th>Datatype (Number, String, Image, Boolean…)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(worksheet continues on the next page)
Define the Data Structure

# a __________State is ______________________
data __________State:
  | __________(_________________________________
  
  
  
  __________________________________________

end

Make a sample instance for each sketch from the previous page:

__________ = __________________________________

__________ = __________________________________

__________ = __________________________________

__________ = __________________________________
Identifying Animation Data Worksheet

Draw a sketch for three distinct moments of the animation

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG...)</th>
<th>Datatype (Number, String, Image, Boolean...)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(worksheet continues on the next page)
Define the Data Structure

# a _________State is ____________________________

data _________State:

| _________(_________________________________

| _________________________________

| _________________________________

| _________________________________

| _________________________________

end

Make a sample instance for each sketch from the previous page:

_________ = _______________________________________

_________ = _______________________________________

_________ = _______________________________________

_________ = _______________________________________

_________ = _______________________________________
Write a function called `location`, which consumes a `DeliveryState`, and produces a String representing the location of a box: either “road”, “delivery zone”, “house”, or “air”.

Contract+Purpose Statement

# __________________ : _______________________________ → ___________
# ________________________________

Give Examples

examples:

_________ (______________________) is ________________

_________ (______________________) is ________________

_________ (______________________) is ________________

_________ (______________________) is ________________

_________ (______________________) is ________________

end

(worksheet continues next page)
fun ______________________(_______________________) :

    if __________________________________________:

        __________________________________________

    else if ______________________________________:

        __________________________________________

    else if ______________________________________:

        __________________________________________

    else: _________________________________________

    end

end
<table>
<thead>
<tr>
<th>Round</th>
<th>Buggy Code</th>
<th>Correct Code / Explanation</th>
</tr>
</thead>
</table>
| 1     | `fun piecewisefun(n):`  
      |   `if (n > 0): n`  
      |   `else: 0`          | |
| 2     | `fun cost(topping):`  
      |   `if string-equal(topping,  
          "pepperoni"): 10.50`  
      |   `else string-equal(topping,  
          "cheese"): 9.00`  
      |   `else string-equal(topping,  
          "chicken"): 11.25`  
      |   `else string-equal(topping,  
          "broccoli"): 10.25`  
      |   `else: "That's not on the menu!"`  
      |   `end`          | |
| 3     | `fun absolute-value(a b):`  
      |   `if a > b: a - b`  
      |   `b - a`          | |
| 4     | `fun best-function(f):`  
      |   `if string-equal(f, "blue"):  
          "you win!"`  
      |   `else if string-equal(f, "blue"):  
          "you lose!"`  
      |   `else if string-equal(f, "red"):  
          "Try again!"`  
      |   `else: "Invalid entry!"`  
      |   `end`          | |
Animation Extension Worksheet

Describe the goal of your change: what new feature or behavior will it add to your animation?

Draw a sketch for three distinct moments of the animation

Sketch A

Sketch B

Sketch C

What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG…</th>
<th>Datatype (Number, String, Image, Boolean…</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Make a To-Do List, and check off each as “Done” when you finish each one.

<table>
<thead>
<tr>
<th>Component</th>
<th>When is there work to be done?</th>
<th>To-Do</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Structure</td>
<td>If any new field(s) were added, changed or removed</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>draw-state</td>
<td>If something is displayed in a new way or position</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-tick</td>
<td>If the Data Structure changed, or the animation happens automatically</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-key</td>
<td>If the Data Structure changed, or a keypress triggers the animation</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>reactor</td>
<td>If either next-state function is new</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
Make a sample instance for each sketch from the previous page:

_______ = __________________________________

_______ = _______________________________

_______ = __________________________________

Write at least one NEW example for one of the functions on your To-Do list

__________________________________________________________________________

__________________________________________________________________________

__________________________________________________________________________

__________________________________________________________________________

__________________________________________________________________________

__________________________________________________________________________

If you have another function on your To-Do list, write at least one NEW example

__________________________________________________________________________

__________________________________________________________________________

__________________________________________________________________________

__________________________________________________________________________

__________________________________________________________________________

__________________________________________________________________________
Write a function called `draw-sun`, which consumes a SunsetState, and produces an image of a sun (a solid, 25 pixel circle), whose color is "yellow", when the sun’s y-coordinate is greater than 225, "orange", when its y-coordinate is between 150 and 225, and "red" otherwise.

**Contract+Purpose Statement**

```
# ______________ : __________________________ → ____________
# ____________________________
```

**Give Examples**

```
examples:
    ______(__________________) is ____________
    ______(__________________) is ____________
    ______(__________________) is ____________
    ______(__________________) is ____________
```

end
fun ________________(______________________) :
    if __________________________________________________________________:
        ______________________________________
    else if __________________________________________________________________:
        ______________________________________
    else: __________________________________________________________________

end
end
Animation Extension Worksheet

Describe the goal of your change: what new feature or behavior will it add to your animation?

Decrease the cat’s hunger level by 2 and sleep level by 1 on each tick.

Draw a sketch for three distinct moments of the animation, focusing on the new behavior:

Sketch A  Sketch B  Sketch C

What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG...)</th>
<th>Datatype (Number, String, Image, Boolean...)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Make a To-Do List, and check off each as “Done” when you finish each one.

<table>
<thead>
<tr>
<th>Component</th>
<th>When is there work to be done?</th>
<th>To-Do</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Structure</td>
<td>If any new field(s) were added, changed or removed</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>draw-state</td>
<td>If something is displayed in a new way or position</td>
<td>☑</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-tick</td>
<td>If the Data Structure changed, or the animation happens automatically</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-key</td>
<td>If the Data Structure changed, or a keypress triggers the animation</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>reactor</td>
<td>If either next-state function is new</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
Make a sample instance for each sketch from the previous page:

**FULLPET** = \( \text{pet}(100, 100) \)

**MIDPET** = \( \text{pet}(50, 75) \)

**LOSEPET** = \( \text{pet}(0, 0) \)

Write at least one NEW example for one of the functions on your To-Do list:

next-state-tick(FULLPET) is pet(FULLPET.hunger - 2, FULLPET.sleep - 1)

next-state-tick(MIDPET) is pet(MIDPET.hunger - 2, MIDPET.sleep - 1)

next-state-tick(LOSEPET) is LOSEPET

If you have another function on your To-Do list, write at least one NEW example:

________________________________________________________________________________

________________________________________________________________________________

________________________________________________________________________________

________________________________________________________________________________
Animation Extension Worksheet
Describe the goal of your change: what new feature or behavior will it add to your animation?

Draw a sketch for three distinct moments of the animation

Sketch A | Sketch B | Sketch C

What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG…)</th>
<th>Datatype (Number, String, Image, Boolean…)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Make a To-Do List, and check off each as “Done” when you finish each one.

<table>
<thead>
<tr>
<th>Component</th>
<th>When is there work to be done?</th>
<th>To-Do</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Structure</td>
<td>If any new field(s) were added, changed or removed</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>draw-state</td>
<td>If something is displayed in a new way or position</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-tick</td>
<td>If the Data Structure changed, or the animation happens automatically</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-key</td>
<td>If the Data Structure changed, or a keypress triggers the animation</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>reactor</td>
<td>If either next-state function is new</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
Make a sample instance for each sketch from the previous page:

_________ = ______________________________________________________

_________ = ______________________________________________________

_________ = ______________________________________________________

Write at least one NEW example for one of the functions on your To-Do list

____________________________________________________________________

____________________________________________________________________

____________________________________________________________________

____________________________________________________________________

If you have another function on your To-Do list, write at least one NEW example

____________________________________________________________________

____________________________________________________________________

____________________________________________________________________

____________________________________________________________________
### Animation Extension Worksheet

Describe the goal of your change: what new feature or behavior will it add to your animation?

#### Draw a sketch for three distinct moments of the animation

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

#### What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name</th>
<th>Datatype</th>
</tr>
</thead>
<tbody>
<tr>
<td>(dangerX, score, playerIMG…)</td>
<td>(Number, String, Image, Boolean…)</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Make a To-Do List, and check off each as “Done” when you finish each one.

<table>
<thead>
<tr>
<th>Component</th>
<th>When is there work to be done?</th>
<th>To-Do</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Structure</td>
<td>If any new field(s) were added, changed or removed</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>draw-state</td>
<td>If something is displayed in a new way or position</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-tick</td>
<td>If the Data Structure changed, or the animation happens automatically</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-key</td>
<td>If the Data Structure changed, or a keypress triggers the animation</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>reactor</td>
<td>If either next-state function is new</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
Make a sample instance for each sketch from the previous page:

______   =   _______________________________

______   =   ________________________________

______   =   ________________________________

Write at least one NEW example for one of the functions on your To-Do list

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

If you have another function on your To-Do list, write at least one NEW example

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________
Animation Design Worksheet

Draw a sketch for three distinct moments of the animation

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG…)</th>
<th>Datatype (Number, String, Image, Boolean…)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Make a To-Do List, and check off each as “Done” when you finish each one.

<table>
<thead>
<tr>
<th>Component</th>
<th>When is there work to be done?</th>
<th>To-Do</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Structure</td>
<td>If any new field(s) were added, changed or removed</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>draw-state</td>
<td>If something is displayed in a new way or position</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-tick</td>
<td>If the Data Structure changed, or the animation happens automatically</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-key</td>
<td>If the Data Structure changed, or a keypress triggers the animation</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>reactor</td>
<td>If either next-state function is new</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
Define the Data Structure

# a _______State is ______________________

data _______State:

| _______ (_________________________________

_________________________________________

_________________________________________

_________________________________________

) end

Make a sample instance for each sketch from the previous page:

_______ = ________________________________

_______ = ________________________________

_______ = ________________________________

Write an example for one of the functions on the previous page:

________________________________________

________________________________________

________________________________________

________________________________________

________________________________________
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:

\[ \sqrt{(4 - 0)^2 + (2 - 5)^2} \]

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

Convert it into Pyret code:
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:

\[
\text{Distance}^2 = (px - cx)^2 + (py - cy)^2
\]

### 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

```markdown
# __________ : _______________________ -> ________

# __________________________________________________________________________
```

Give Examples

Write examples of your function in action

`examples:`

```markdown
__________________________(____________________)

is __________________________________________

__________________________(____________________)

is __________________________________________
```

end

Function

```markdown
fun ______________(______________) : 

c______________________________

c______________________________

end
```
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:

__________ (__________)
    the user types...

is ________________________________________________________________
    ...which should become

____________________________________________________________

__________ (__________)
    the user types...

is ________________________________________________________________
    ...which should become

den

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:

____________(____________)

the user types...

is ____________________________

...which should become

____________(____________)

the user types...

is ____________________________

...which should become

end

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

fun __________________(______________):


end
**Animation Design Worksheet**

Draw a sketch for three distinct moments of the animation

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

**What things are changing?**

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**What fields do you need to represent the things that change?**

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG…)</th>
<th>Datatype (Number, String, Image, Boolean…)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Make a To-Do List, and check off each as “Done” when you finish each one.**

<table>
<thead>
<tr>
<th>Component</th>
<th>When is there work to be done?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Structure</td>
<td>If any new field(s) were added, changed or removed</td>
</tr>
<tr>
<td>draw-state</td>
<td>If something is displayed in a new way or position</td>
</tr>
<tr>
<td>next-state-tick</td>
<td>If the Data Structure changed, or the animation happens automatically</td>
</tr>
<tr>
<td>next-state-key</td>
<td>If the Data Structure changed, or a keypress triggers the animation</td>
</tr>
<tr>
<td>reactor</td>
<td>If either next-state function is new</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>To-Do</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
Define the Data Structure

# a _________State is __________________________
data _________State:

| _________(_________________________________

_________________________________________

| _______________________________

end

Make a sample instance for each sketch from the previous page:

_________ = __________________________________

_________ = __________________________________

_________ = __________________________________

Write an example for one of the functions on the previous page:

_____________________________________________

_____________________________________________

_____________________________________________

_____________________________________________
# Animation Design Worksheet

**Draw a sketch for three distinct moments of the animation**

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

**What things are changing?**

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
</table>

**What fields do you need to represent the things that change?**

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG…)</th>
<th>Datatype (Number, String, Image, Boolean…)</th>
</tr>
</thead>
</table>

**Make a To-Do List, and check off each as "Done" when you finish each one.**

<table>
<thead>
<tr>
<th>Component</th>
<th>When is there work to be done?</th>
<th>To-Do</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Structure</td>
<td>If any new field(s) were added, changed or removed</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>draw-state</td>
<td>If something is displayed in a new way or position</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-tick</td>
<td>If the Data Structure changed, or the animation happens automatically</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-key</td>
<td>If the Data Structure changed, or a keypress triggers the animation</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>reactor</td>
<td>If either next-state function is new</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
Define the Data Structure

# a _______State is ____________________________

data _______State:
  | _______ (____________________________________
  | ___________________________________________
  | ___________________________________________
  | ___________________________________________
end

Make a sample instance for each sketch from the previous page:

_______ = ______________________________________
_______ = ______________________________________
_______ = ______________________________________
_______ = ______________________________________

Write an example for one of the functions on the previous page:

______________________________
______________________________
______________________________
______________________________
______________________________
______________________________
______________________________
______________________________
______________________________
Animation Design Worksheet

Draw a sketch for three distinct moments of the animation

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG...)</th>
<th>Datatype (Number, String, Image, Boolean...)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Make a To-Do List, and check off each as “Done” when you finish each one.

<table>
<thead>
<tr>
<th>Component</th>
<th>When is there work to be done?</th>
<th>To-Do</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Structure</td>
<td>If any new field(s) were added, changed or removed</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>draw-state</td>
<td>If something is displayed in a new way or position</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-tick</td>
<td>If the Data Structure changed, or the animation happens automatically</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-key</td>
<td>If the Data Structure changed, or a keypress triggers the animation</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>reactor</td>
<td>If either next-state function is new</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
Define the Data Structure

# a _________State is ___________________________

data _________State:

| ________|(______________________________________
|          |
|          |
|          |

end

Make a sample instance for each sketch from the previous page:

_______ = __________________________________

_______ = __________________________________

_______ = __________________________________

Write an example for one of the functions on the previous page:

_________________________________________________

_________________________________________________

_________________________________________________

_________________________________________________
Animation Extension Worksheet

Describe the goal of your change: what new feature or behavior will it add to your animation?

Draw a sketch for three distinct moments of the animation

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

What things are changing?

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What fields do you need to represent the things that change?

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG…)</th>
<th>Datatype (Number, String, Image, Boolean…)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Make a To-Do List, and check off each as “Done” when you finish each one.

<table>
<thead>
<tr>
<th>Component</th>
<th>When is there work to be done?</th>
<th>To-Do</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Structure</td>
<td>If any new field(s) were added, changed or removed</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>draw-state</td>
<td>If something is displayed in a new way or position</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-tick</td>
<td>If the Data Structure changed, or the animation happens automatically</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-key</td>
<td>If the Data Structure changed, or a keypress triggers the animation</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>reactor</td>
<td>If either next-state function is new</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
Make a sample instance for each sketch from the previous page:

_______ = ________________________________

_______ = ________________________________

_______ = ________________________________

Write at least one NEW example for one of the functions on your To-Do list

________________________________________________________________________________

________________________________________________________________________________

________________________________________________________________________________

If you have another function on your To-Do list, write at least one NEW example

________________________________________________________________________________

________________________________________________________________________________

________________________________________________________________________________
**Animation Extension Worksheet**

Describe the goal of your change: what new feature or behavior will it add to your animation?

---

**Draw a sketch for three distinct moments of the animation**

<table>
<thead>
<tr>
<th>Sketch A</th>
<th>Sketch B</th>
<th>Sketch C</th>
</tr>
</thead>
</table>

---

**What things are changing?**

<table>
<thead>
<tr>
<th>Thing</th>
<th>Describe how it changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**What fields do you need to represent the things that change?**

<table>
<thead>
<tr>
<th>Field name (dangerX, score, playerIMG…)</th>
<th>Datatype (Number, String, Image, Boolean…)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**Make a To-Do List, and check off each as “Done” when you finish each one.**

<table>
<thead>
<tr>
<th>Component</th>
<th>When is there work to be done?</th>
<th>To-Do</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Structure</td>
<td>If any new field(s) were added, changed or removed</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>draw-state</td>
<td>If something is displayed in a new way or position</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-tick</td>
<td>If the Data Structure changed, or the animation happens automatically</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>next-state-key</td>
<td>If the Data Structure changed, or a keypress triggers the animation</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>reactor</td>
<td>If either next-state function is new</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>
Make a sample instance for each sketch from the previous page:

_ = _______________________________

_ = _______________________________

_ = _______________________________

Write at least one NEW example for one of the functions on your To-Do list

_________________________________________________

_________________________________________________

_________________________________________________

_________________________________________________

If you have another function on your To-Do list, write at least one NEW example

_________________________________________________

_________________________________________________

_________________________________________________

_________________________________________________
<table>
<thead>
<tr>
<th>Contracts</th>
<th>Domain</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>example</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Range</td>
<td>↑</td>
<td></td>
</tr>
</tbody>
</table>

*Note: The table is incomplete and requires additional entries.*