# The Design Recipe

For the word problems below, assume `animalA` and `animalB` are defined as the data rows for Felix and Midnight, respectively.

**Directions**: Define a function called `lookup-fixed`, which looks up whether or not an animal is fixed.

## Contract and Purpose Statement

*Every contract has three parts…*

# `lookup-fixed` :: `(r :: Row)` -> `Boolean`

*function name* | *domain* | *range*

# Consumes an animal, and looks up the value in the fixed column.

*what does the function do?*

## Examples

*Write some examples, then circle and label what changes…*

**examples:**

_____ ( _____ ) **is** _____

*function name* | *input(s)* | *what the function produces*

_____ ( _____ ) **is** _____

*function name* | *input(s)* | *what the function produces*

**end**

## Definition

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

**fun** `lookup-fixed(` `r` `):`

*function name* | *variable(s)*

`r["fixed"]`

*what the function does with those variable(s)*

**end**

---

**Directions**: Define a function called `lookup-sex`, which consumes a Row of the animals table and looks up the sex of that animal.

## 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…*

**examples:**

_____ ( _____ ) **is** _____

*function name* | *input(s)* | *what the function produces*

_____ ( _____ ) **is** _____

*function name* | *input(s)* | *what the function produces*

**end**

## Definition

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

**fun** _____ ( _____ ) **:**

*function name* | *variable(s)*

_____

*what the function does with those variable(s)*

**end**