The Design Recipe

Directions: Define a function called `gt`, which makes solid green triangles of whatever size we want.

Contract and Purpose Statement

Every contract has three parts…

```
# gt:: (size :: Number) -> Image
```

function name | *domain* | *range*

```
# Consumes a size, and produces a solid green triangle of that size.
```

what does the function do?

Examples

Write some examples, then circle and label what changes…

examples:

```
gt ("10") is triangle(10, "solid", "green")
gt ("20") is triangle(20, "solid", "green")
```

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

end

Definition

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

```
fun gt(size):
  triangle(size, "solid", "green")
end
```

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

Directions: Define a function called `bc`, which makes solid blue circles of whatever radius we want.

Contract and Purpose Statement

Every contract has three parts…

```
# bc:: (radius :: Number) -> Image
```

function name | *domain* | *range*

```
# Consumes a radius, and produces a solid blue circle with that radius.
```

what does the function do?

Examples

Write some examples, then circle and label what changes…

examples:

```
bc ("10") is circle(10, "solid", "blue")
bc ("20") is circle(20, "solid", "blue")
```

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

end

Definition

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

```
fun bc(radius):
  circle(radius, "solid", "blue")
end
```

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