Contracts

Contracts tell us how to use a function. For example: `ellipse :: (Number, Number, String, String) -> Image` tells us that the name of the function is `ellipse`, it takes four inputs (two Numbers and two Strings), and it evaluates to an `Image`. From the contract, we know `ellipse(50, 100, "solid", "teal")` will evaluate to an `Image`.

Name		Domain		Range
# text	::	(String, Number, String)	->	Image
text("I'm thankful for…", 50, "green")				
# image-url	::	(String)	->	Image
image-url("https://www.bootstrapworld.org/images/icon.png")				
# scale	::	(Number, Image)	->	Image
scale(0.8, triangle(30, "solid", "red"))				
# rotate	::	(Number, Image)	->	Image
rotate(35, rectangle(30, 80, "solid", "purple"))				
# overlay	::	(Image, Image)	->	Image
overlay(star(30, "solid", "gold"),circle(30, "solid", "blue"))				
# put-image	::	(Image, Number, Number, Image)	->	Image
put-image(star(30, "solid", "red"), 50, 150, rectangle(300, 200, "outline", "black"))				
# flip-horizontal	::	(Image)	->	Image
flip-horizontal(text("Bootstrap is fun!", 60, "darkblue"))				
# flip-vertical	::	(Image)	->	Image
flip-vertical(triangle(80, "solid", "lightgreen"))				
# above	::	(Image, Image)	->	Image
above(triangle(30, "solid", "red"),square(30, "solid", "blue"))				
# beside	::	(Image, Image)	->	Image
beside(star(50, "solid", "orange"),circle(50, "solid", "green"))				