

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