ho	00	an	D	00	Ha	
	<b>6</b> 3	ULL		ec	Ш٤	Ū

Directions. Define a function of	alled sticker	, will	:h dr	aws 50px stars in v	whatever color is in	put.		
Contract and Purpose State	ment							
Every contract has three parts								
# sticker::		(	r	:: Row)		->	Image	
function name				domain			range	_
# Consumes a color and prod	luces a solid, 5	50px s	star	in that color				
		w	vhat do	es the function do?				
Examples								
Write some examples, then circle and la	abel what changes							
examples:								
sticker ( '	"\"red\""	) :	is	star(50, "sol	id", "red")			
function name	input(s)	_′			what the function produ	ces		
	\"green\""	) :	ic	star(50, "sol	• •			
function name		<b>—</b> ′	13	3 (30, 30)	what the function produ			
end end	input(s)				what the function produ	ces		
Definition								
Write the definition, giving variable nan	nes to all your input	values						
fun sticker(	color ):							
function name	variable(s)							
star(50, "solid", co	lor)							
		hat the fun	ction o	loes with those variable(s)				
end		,		**				
<b>Directions</b> : Define a function c	<b>alled</b> nametag	, whic	:h co	onsumes a Row of	the animals table a	and draws	their name in	n purple,
<b>Directions</b> : Define a function con 10px letters. (Assume you have					the animals table a	and draws	their name in	n purple,
	rows animalA				the animals table a	and draws	their name in	n purple,
10px letters. (Assume you have	rows animalA				the animals table a	and draws	their name in	n purple,
10px letters. (Assume you have Contract and Purpose States Every contract has three parts	rows animalA	and	ani	malB <b>defined.)</b>	the animals table a			n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  nametag::	rows animalA	and	ani	malB defined.) :: Row)	the animals table a	and draws	Image	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag::  function name	rows animalA ment	and (	ani r	malB defined.) :: Row) domain				n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  nametag::	rows animalA ment	and ( nimal's	r s na	.malB defined.)  :: Row)  domain  ame in purple, 10			Image	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag::  function name	rows animalA ment	and ( nimal's	r s na	malB defined.) :: Row) domain			Image	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag::  function name	rows animalA ment	and ( nimal's	r s na	.malB defined.)  :: Row)  domain  ame in purple, 10			Image	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag::  function name  # Consumes an animal, and p	rows animalA ment produces that a	and ( nimal's	r s na	.malB defined.)  :: Row)  domain  ame in purple, 10			Image	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag:: function name  # Consumes an animal, and purpose  Examples  Write some examples, then circle and letters	rows animalA ment produces that a	and ( nimal's	r s na	.malB defined.)  :: Row)  domain  ame in purple, 10			Image	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag::  function name  # Consumes an animal, and p  Examples  Write some examples, then circle and locations  examples:	rows animalA ment produces that a	and ( nimal's	ani r s na	.malB defined.)  :: Row)  domain  ame in purple, 10	px letters.	->	Image range	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag::  function name  # Consumes an animal, and purpose  Examples  Write some examples, then circle and leeexamples:	rows animalA ment  produces that a abel what changes	and ( nimal's	ani r s na	:: Row)  domain ame in purple, 10 es the function do?	px letters.	->_ "purple	Image range	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag::  function name  # Consumes an animal, and p  Examples  Write some examples, then circle and location examples:  nametag (  function name	rows animalA ment  produces that a abel what changes "animalA" input(s)	and ( nimal's	r s na	:: Row)  domain  ame in purple, 10  es the function do?	name"], 10,	->_ "purple'	Image range	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag:: function name  # Consumes an animal, and purpose  Examples  Write some examples, then circle and loce examples:	rows animalA ment  produces that a abel what changes  "animalA"  input(s)  "animalB"	and ( nimal's	r s na	:: Row)  domain ame in purple, 10 es the function do?	name"], 10, what the function produ	-> "purple'	Image range	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag::  function name  # Consumes an animal, and p  Examples  Write some examples, then circle and location examples:  nametag (  function name	rows animalA ment  produces that a abel what changes "animalA" input(s)	and ( nimal's	r s na	:: Row)  domain  ame in purple, 10  es the function do?	name"], 10,	-> "purple'	Image range	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag::  function name  # Consumes an animal, and p  Examples  Write some examples, then circle and loce examples:  nametag (  function name  nametag (  function name  nametag (	rows animalA ment  produces that a abel what changes  "animalA"  input(s)  "animalB"	and ( nimal's	r s na	:: Row)  domain  ame in purple, 10  es the function do?	name"], 10, what the function produ	-> "purple'	Image range	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag:: function name  # Consumes an animal, and p  Examples  Write some examples, then circle and location examples:  nametag ( function name	rows animalA ment  produces that a abel what changes  "animalA"  input(s)  "animalB"  input(s)	and ( nimal': "):):	r s na what do	:: Row)  domain  ame in purple, 10  es the function do?	name"], 10, what the function produ	-> "purple'	Image range	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag::     function name  # Consumes an animal, and purpose  Examples  Write some examples, then circle and locate and	rows animalA ment  produces that a abel what changes  "animalA"  input(s)  "animalB"  input(s)	and ( nimal': "):):	r s na what do	:: Row)  domain  ame in purple, 10  es the function do?	name"], 10, what the function produ	-> "purple'	Image range	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag:: function name  # Consumes an animal, and p  Examples  Write some examples, then circle and location name examples:  nametag ( function name	rows animalA ment  produces that a abel what changes "animalA" input(s) "animalB" input(s)	and ( nimal': "):):	r s na what do	:: Row)  domain  ame in purple, 10  es the function do?	name"], 10, what the function produ	-> "purple'	Image range	n purple,
10px letters. (Assume you have  Contract and Purpose States  Every contract has three parts  # nametag::  function name  # Consumes an animal, and p  Examples  Write some examples, then circle and location name  examples:  nametag (  function name  nametag (  function name  end  Definition  Write the definition, giving variable name  fun nametag (  function name	rows animalA ment  produces that a abel what changes  "animalA" input(s) "animalB" input(s)  the input(s)  mes to all your input  r ):	and ( nimal': "):):	r s na what do	:: Row)  domain  ame in purple, 10  es the function do?	name"], 10, what the function produ	-> "purple'	Image range	n purple,

end