Racket Programming Assignment #1

Learning Abstract

This assignment is designed to introduce students to DrRacket by simple interactions. For the first three tasks, I had to mimic example interactions from Racket lesson #1. For tasks 4 and 5, I had to recreate an image of 5 concentric circles and compute the area of the blue region in the image.

## (a)   Simple Numeric Processing

```
Welcome to DrRacket, version 8.6 [cs].
Language: racket, with debugging; memory limit: 128 MB.
> x
    x: undefined;
 cannot reference an identifier before its definition
> 55
55
> 55.2
55.2
> pi
3.141592653589793
> ( * 3 8 )
24
> ( + ( * 3 8 ) 6 )
30
> ( expt 2 8 )
256
> ( * pi ( expt 7 2 ) )
153.93804002589985
> ( expt 9 50 )
515377520732011331036461129765621272702107522001
>
```

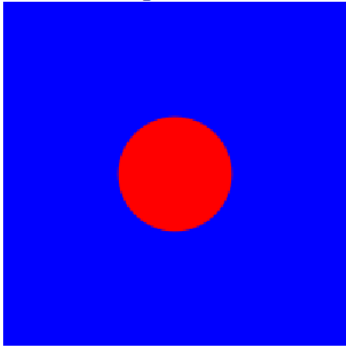## (b)   Solution to the blue and red tile area problem

```
Welcome to DrRacket, version 8.6 [cs].
Language: racket, with debugging; memory limit: 128 MB.
> ( define side-of-tile 200 )
> ( define diameter-of-dot ( / side-of-tile 3 ) )
> ( define radius-of-dot ( / diameter-of-dot 2 ) )
> ( define total-tile-area ( expt side-of-tile 2 ) )
> ( define red-dot-area ( * pi ( expt radius-of-dot 2 ) ) )
> ( define blue-tile-area ( - total-tile-area red-dot-area ) )
> side-of-tile
200
> diameter-of-dot
```

$66\frac{2}{3}$

```
> radius-of-dot
```

$33\frac{1}{3}$

```
> total-tile-area
40000
> red-dot-area
3490.658503988659
> blue-tile-area
36509.341496011344
>
```

(c)   Painting the blue and red tile

```
Welcome to DrRacket, version 8.6 [cs].
Language: racket, with debugging; memory limit: 128 MB.
> ( require 2htdp/image )
> ( define side-of-tile 200 )
> ( define diameter-of-dot ( / side-of-tile 3 ) )
> ( define radius-of-dot ( / diameter-of-dot 2 ) )
> ( define tile ( square side-of-tile "solid" "blue" ) )
> tile
```
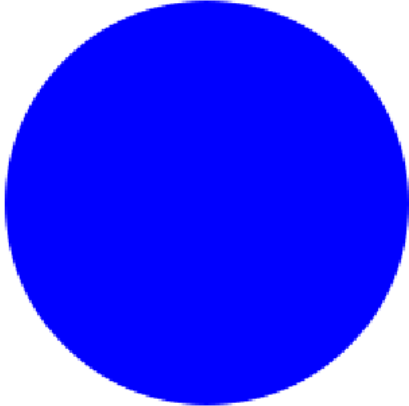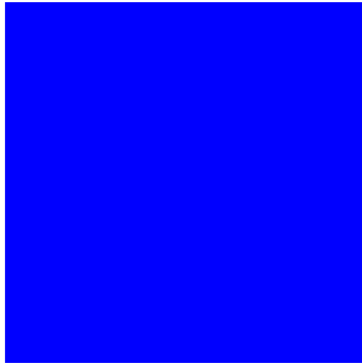
```
> (define dot ( circle radius-of-dot "solid" "red" ) )
> dot
```
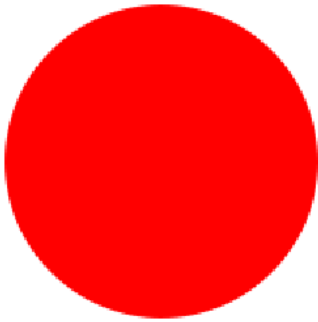


```
> ( overlay dot tile )
```



```
>
```

(d)   Painting the blue and red concentric disks image

```
> ( require 2htdp/image )
> ( define blue-base-radius 100 )
> ( define red-one-radius ( - blue-base-radius 20 ) )
> ( define blue-two-radius ( - red-one-radius 20 ) )
> ( define red-two-radius ( - blue-two-radius 20 ) )
> ( define blue-top-radius ( - red-two-radius 20 ) )
> ( define bottom-blue ( circle blue-base-radius "solid" "blue" ) )
> bottom-blue
```



```
> ( define bottom-red ( circle red-one-radius "solid" "red" ) )
```
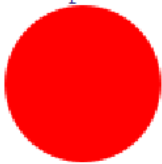
> bottom-red

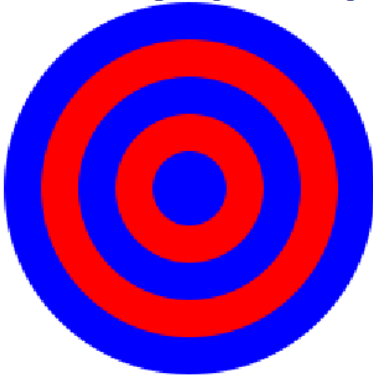> (define middle-blue (circle blue-two-radius "solid" "blue"))
> middle-blue

> (define top-red (circle red-two-radius "solid" "red"))
> top-red

> (define top-blue (circle blue-top-radius "solid" "blue"))
> top-blue

> (overlay top-blue top-red middle-blue bottom-red bottom-blue)

>

(e)   Painting the blue and red concentric disks image

```
Welcome to DrRacket, version 8.6 [cs].
Language: racket, with debugging; memory limit: 128 MB.
> ( define blue-base-radius 100 )
> ( define red-one-radius ( - blue-base-radius 20 ) )
> ( define blue-two-radius ( - red-one-radius 20 ) )
> ( define red-two-radius ( - blue-two-radius 20 ) )
> ( define blue-top-radius ( - red-two-radius 20 ) )
> ( define blue-base-area ( * pi ( expt blue-base-radius 2 ) ) )
> ( define red-one-area ( * pi ( expt red-one-radius 2 ) ) )
> ( define blue-two-area ( * pi ( expt blue-two-radius 2 ) ) )
> ( define red-two-area ( * pi ( expt red-two-radius 2 ) ) )
> ( define blue-top-area ( * pi ( expt blue-top-radius 2 ) ) )
> blue-base-area
31415.926535897932
> red-one-area
20106.192982974677
> blue-two-area
11309.733552923255
> red-two-area
5026.548245743669
> blue-top-area
1256.6370614359173
> ( define blue-area ( + blue-base-area blue-two-area blue-top-area ) )
> blue-area
43982.2971502571
> ( define red-area ( + red-one-area red-two-area ) )
> red-area
25132.741228718347
> ( define total-blue-area ( - blue-area red-area ) )
> total-blue-area
18849.555921538755
>
```