lesson 4

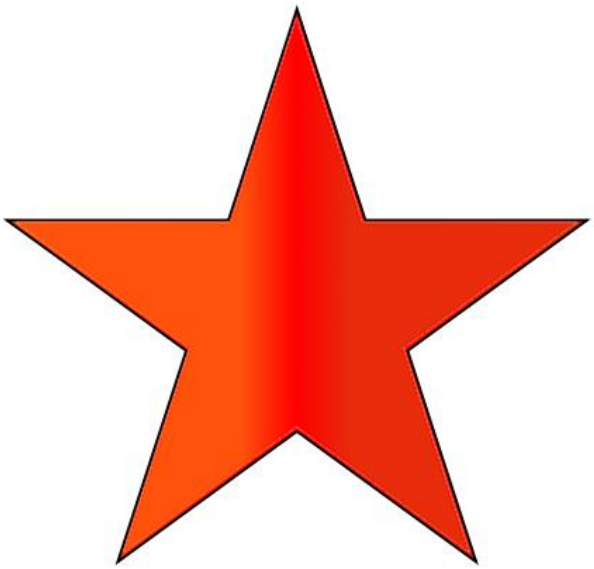
LINEARGRADIENT

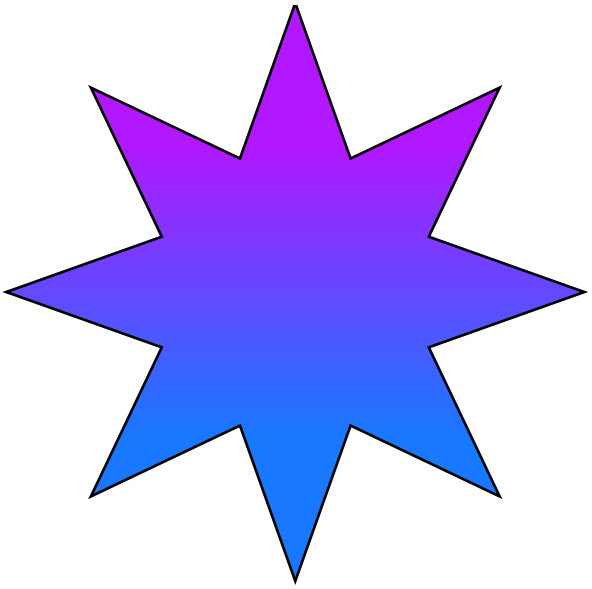
Create a linear gradient with the following vector coordinates. **x1** is equal to 0, **y1** is equal to 0, **x2** is equal to 1920, and **y2** is equal to 1080. Set the gradientUnits attribute to userSpaceOnUse so that the gradient is drawn not within the bounding box, but using the local coordinate system. Set five stops on the vector with the following offset attributes values: 0%, 25%, 50%, 75%, 100% and with the following stop-color attributes values: #B24610, #FF6619, #FF5500, #00B2A6, #00FFED.

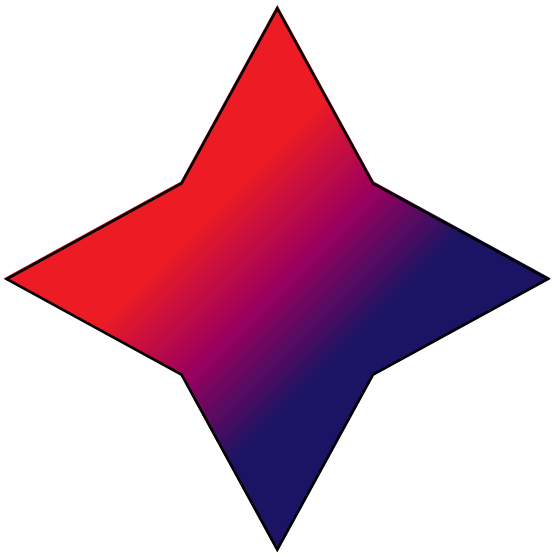
Compare the result with the image. Change the value of the gradientUnits attribute from user-SpaceOnUse to objectBoundingBox and make a conclusion.

lesson 4

LINEARGRADIENT



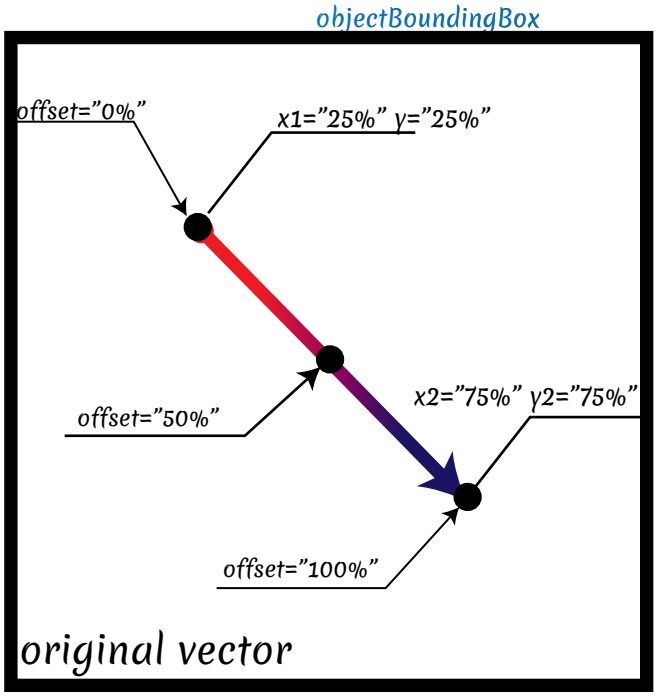


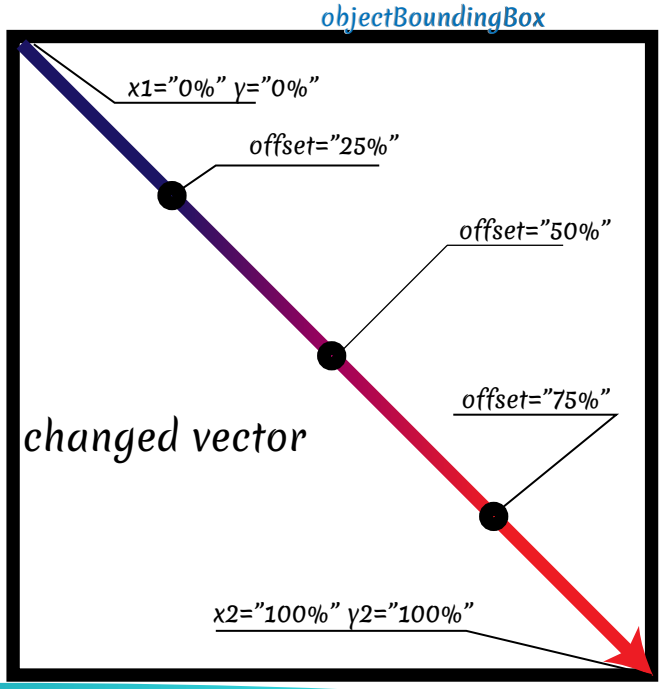


Look at the code of the second module. Draw the vectors with a pencil for the shapes that you can see on the left hand side. Use the vector data from the module 2. (Your drawnings shouldn't be absolutelyprecise).

After you will finish the first part of the assignment. You should rewrite the code. The x1,y1, x2, and y2 attributes should be equal to 0(0%) or to 1(100%).

Use the algorithm illustrated on theimages.

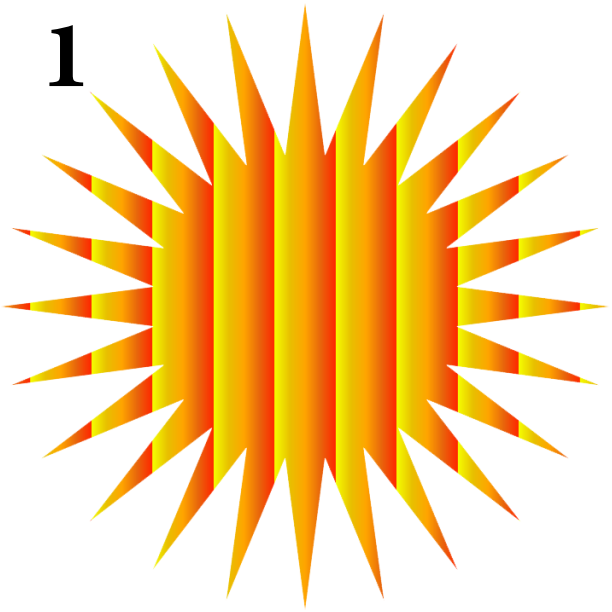




lesson 4

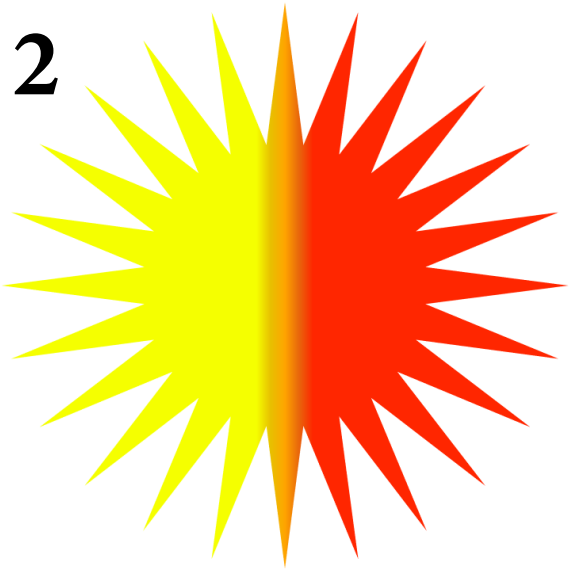
LINEARGRADIENT

1



You already know that vector may not occupy all the space of the object bounding box or local coordinate system. In thisassignment, you will learn how to use the **spreadMethod** attribute, which specifies a rule for painting outside the vector area.
There are three gradients in module three. The first gradient with **id="myGradient"** has five stop elements. Use a gradient with **id="myGradient "** as the source of the content for gradients with **id"=myGradient-1"** and **id="myGradient-2"**. Add the **spreadMethod** attribute with a value of **"reflect"** to the **linearGradient** element with **id="myGradient-1"**. Add the **spreadMethod** attribute with a value of **"repeat"** to the **linearGradient** element with **id="myGradient-2"**.

2



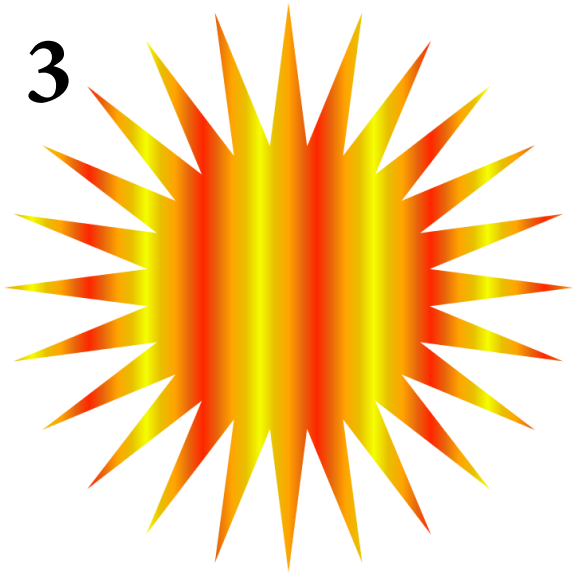
Sign the spreadMethod value that is used in each case.

1) _____

2) _____

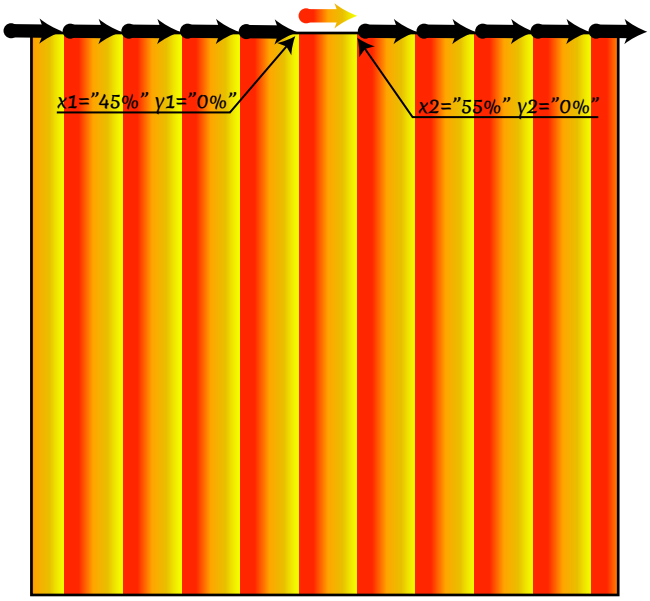
3) _____

3



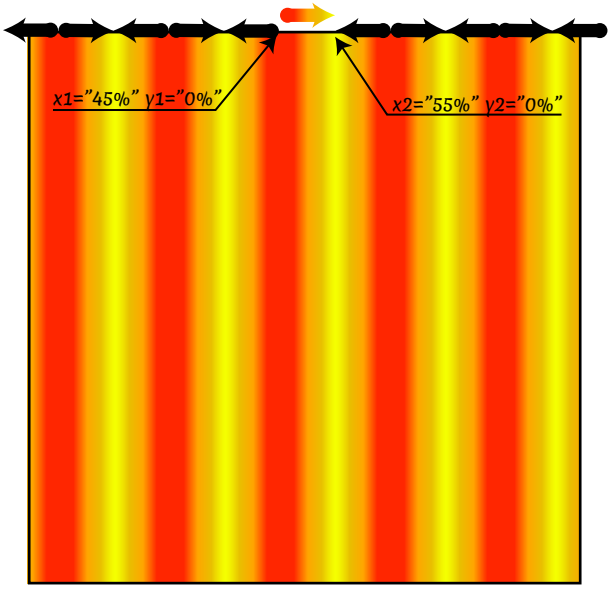
lesson 4

LINEARGRADIENT

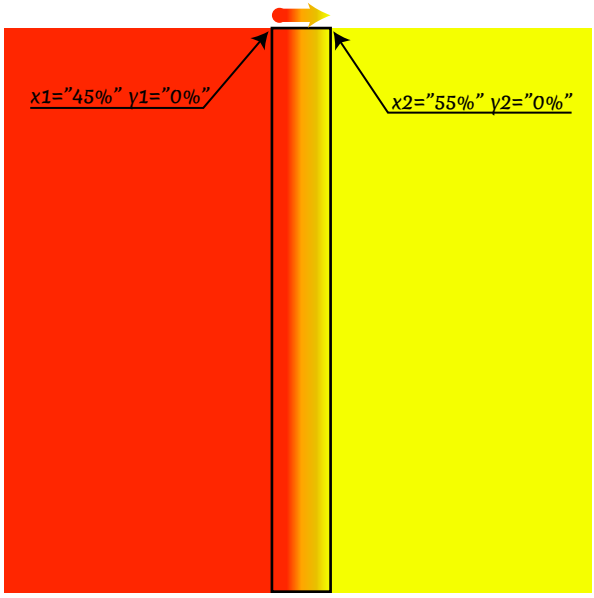


spreadMethod="repeat"

The pictures describe how different spreadMethod values effects on the gradient spread. Use these pictures to solve the assingment.



spreadMethod="reflect"



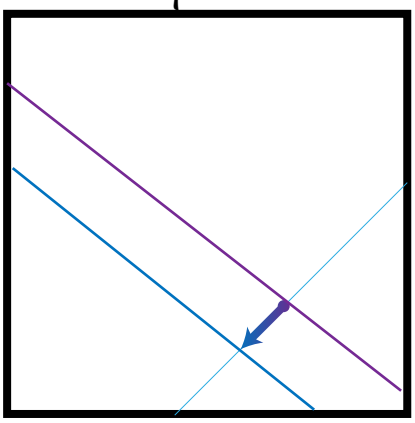
spreadMethod="pad"

lesson 4

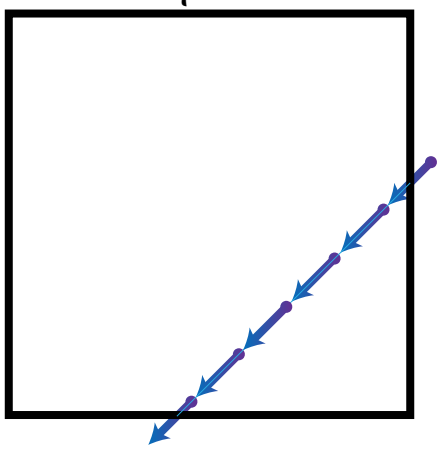
LINEARGRADIENT

Each row has three bounding boxes. Each bounding box has a vector inside of it. Draw the vectors for each spreadMethod values as it is shown in the example.

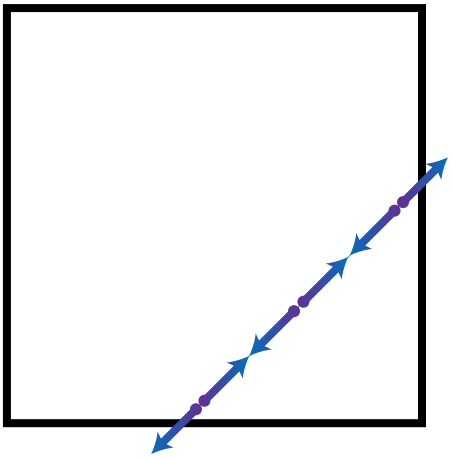
pad



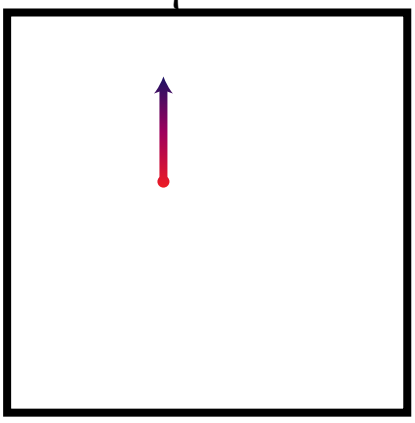
repeat



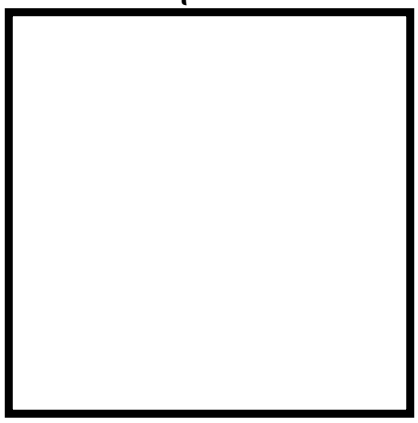
reflect



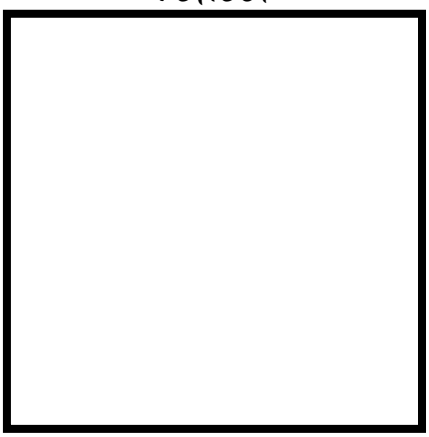
pad



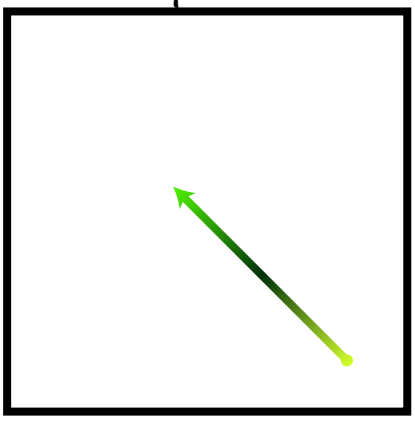
repeat



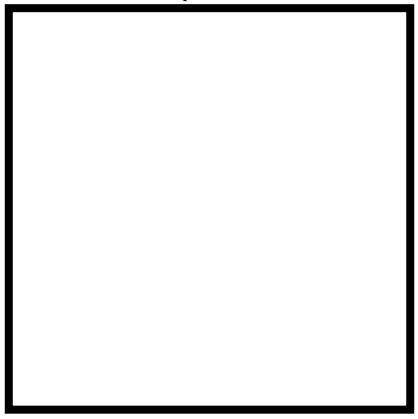
reflect



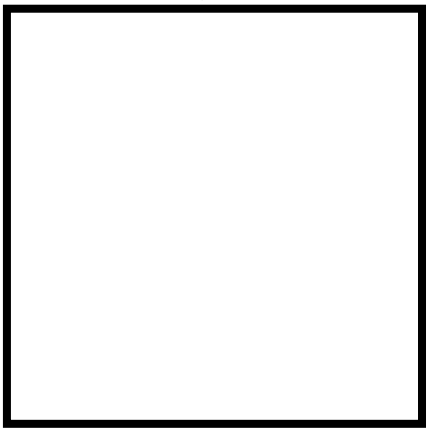
pad



repeat

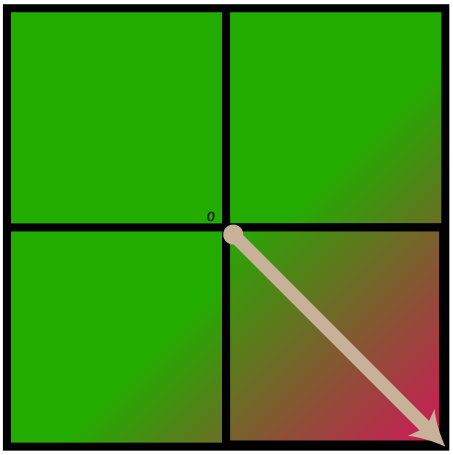


reflect



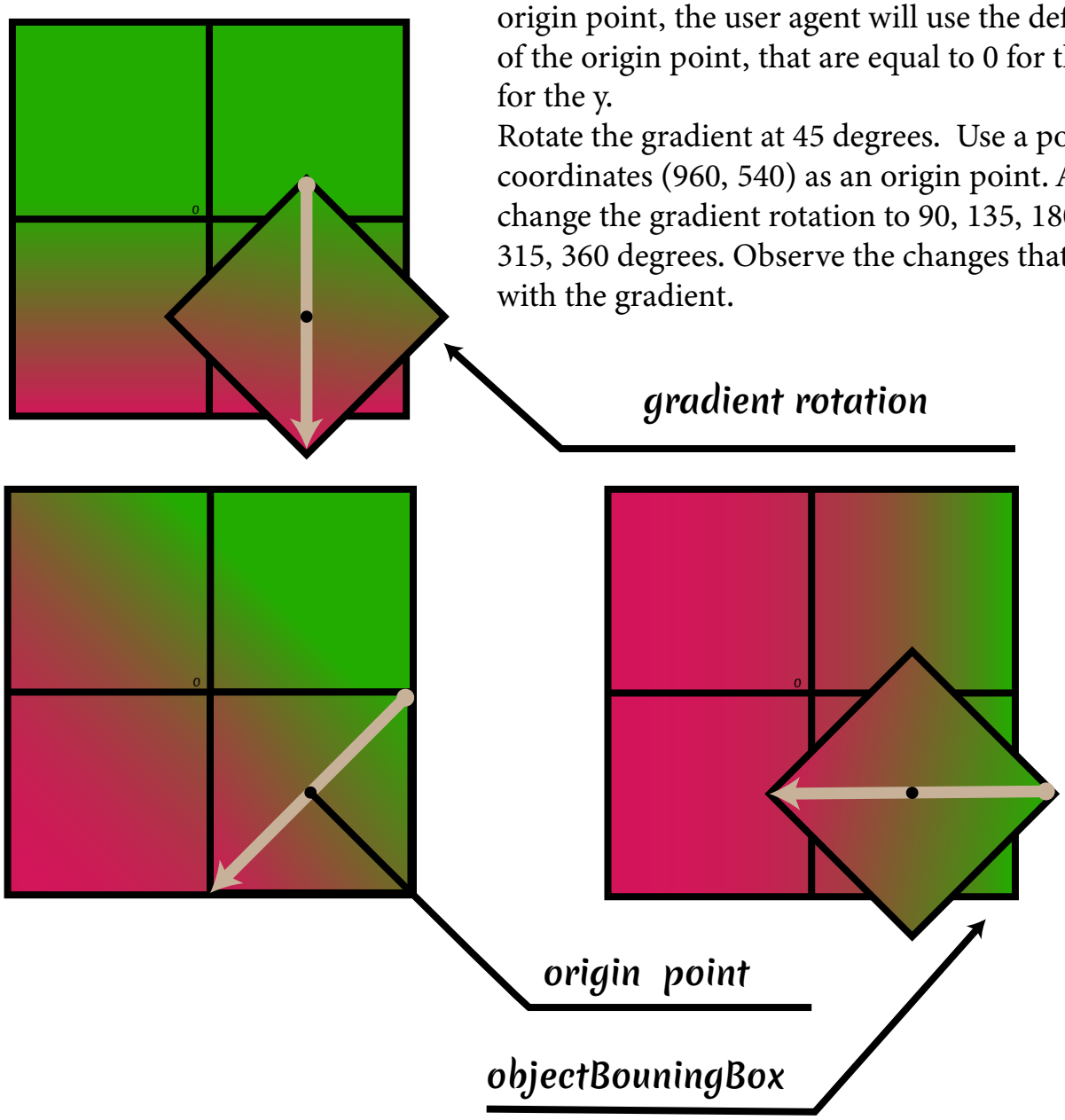
lesson 4

LINEARGRADIENT



In the module 4, add the **gradientTransform** attribute to the **linearGradient** element with the **rotate()** function as a value. The rotate function takes three parameters. The first parameter is the rotation angle, the second parameter is the coordinate of the origin point on x-axis in the **local coordinate system** or the coordinate system of the **objectBoundingBox**, and the third parameter is the coordinate of the origin point on y-axis in the local coordinate system or the coordinate system of objectBoundingBox. The **origin** point is the point around which the gradient will rotate. If you do not specify the coordinates of the origin point, the user agent will use the default values of the origin point, that are equal to 0 for the x and 0 for the y.

Rotate the gradient at 45 degrees. Use a point with coordinates (960, 540) as an origin point. Alternately change the gradient rotation to 90, 135, 180, 225, 270, 315, 360 degrees. Observe the changes that happen with the gradient.



lesson 4

LINEARGRADIENT