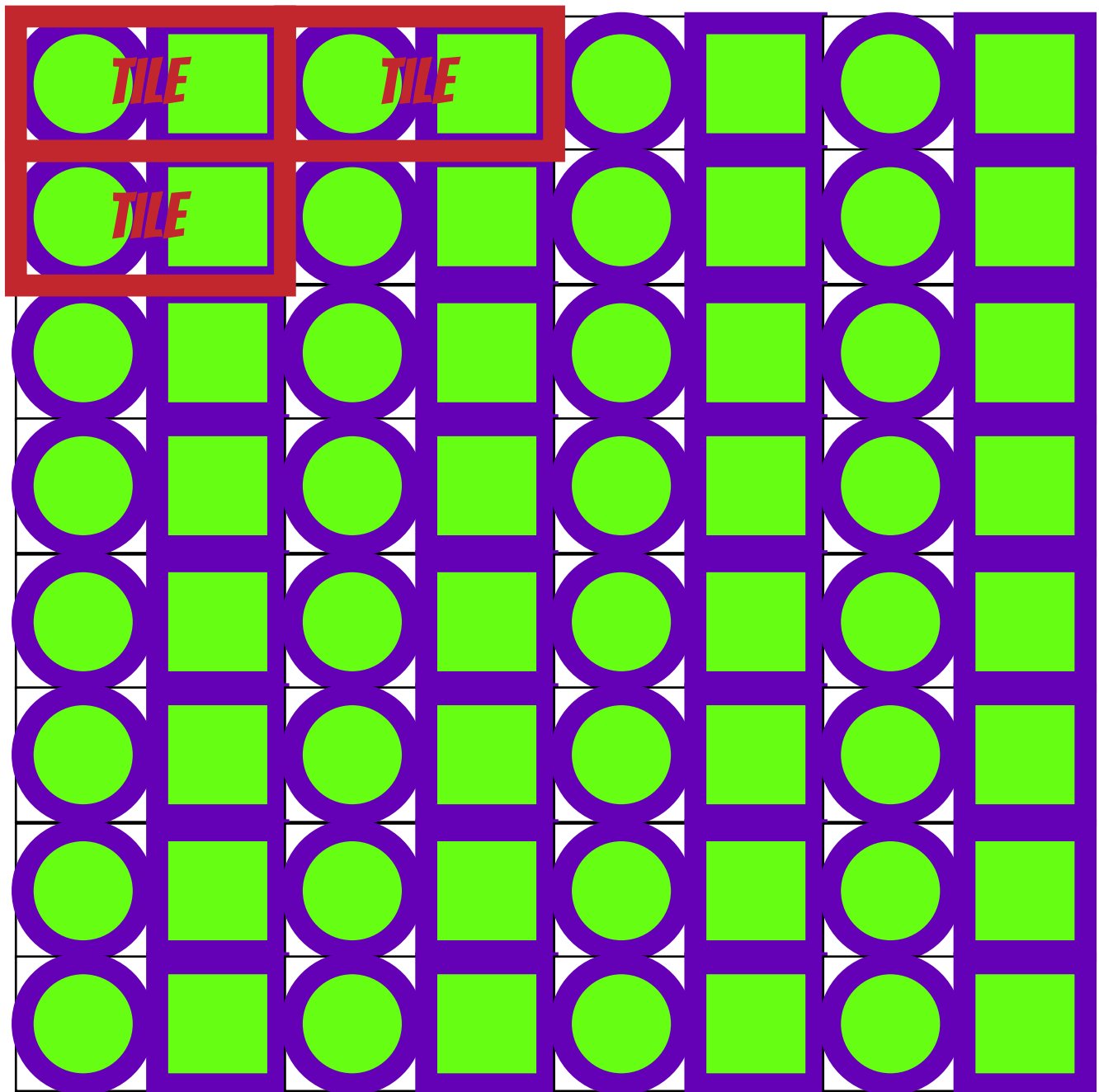


lesson 6

PATTERN



1. In the first module, you can see the pattern element that contains the circle element and the rectangle element. The `cx` geometry property of the circle element is equal to 50, the `cy` geometry property is equal to 50, and the `radius` is equal to 45. The geometry property `x` of the `rect` element is equal to 105, the geometry property `y` is equal to 5, and the geometry properties `width` and `height` are both equal to 90. Let's take a closer look at the pattern element. The attributes `x` and `y` are both equal to 0. The `width` attribute is equal to 200 and the `height` attribute is equal to 100. The `viewBox` attribute is equal to "0 0 200 100".

Now let's look at the `style.css` file. We can see that the `stroke-width` of the circle and rectangle inside the pattern element is 10 pixels. The pattern element is used as the

lesson 6

PATTERN

value of the fill property of a rectangle with id equal to pattern_one.

We can see that our code is not working. This is because the pattern units are set to objectBoundingBox units by default. Let's change this behavior by adding the patternUnits attribute to the pattern element with the value userSpaceOnUse. As we can see in the style.css, the pattern element is used as the value of the fill property.

The patternUnits attribute tells to user-agent which units to use when it will interpret the values of the pattern element. In our case, the user-agent uses the local coordinate system units to create the viewport and viewBox of the pattern.

2. In module two, I added the patternContentUnits attribute with the value "objectBoundingBox". In the previous module, the values of the geometry properties of the circle element and the rect element were interpreted as the units of the local coordinate system. It was so because the viewBox attribute was set. The viewBox attribute overrides the patternContentUnits attribute, regardless of its value.

After I have added the patternContentUnits attribute with the value "objectBoundingBox" all values of the circle and rect elements are interpreted in the objectBoundingBox coordinate system, and that means, that cx, cy, r, x,y width, and height values are percentages of height, width, or normalized diagonal of the bounding box of the shape.

You should rewrite the values of the rect and circle elements from the previous example. Use the dimensions of the rectangle to which the pattern element is applied as the fill value as the dimensions of the object bounding box.

The values must be written in the format 0.12, where the number after the decimal point is a percentage value. In our case $0.12 = 12\%$. Recalculate all the values of the geometric properties of the rect and circle elements that are inside the pattern element. You should also recalculate the stroke-width value inside the in css style file.

The result should be the same as in the first module.

3. In the third module, the patternUnits attribute is set to "objectBoundingBox" and the patternContentUnits attribute is set to "userSpaceOnUse". Using the values of all elements from the first module, recalculate the values for those attributes and properties where it is necessary. The result should be the same as in the first module.

4. In the fourth module, the patternUnits attribute is set to "userSpaceOnUse" and the patternContentUnits attribute is set to "objectBoundingBox". Using the values of all elements from the first module, recalculate the values for those attributes and properties where it is necessary. The result should be the same as in the first module.

5. In the fifth module, add the patternTransform attribute with a value of skew(-25). Look at the result. Then add the rotate(45) function to the list of transformations, and then add the scale(1.1) function to the transformations list.