



ICT in SES

Circles and spheres

Lesson №11



Polygon

Polygon in Suica



Polygon

- Graphical object with properties
- Used to draw regular polygons

Creating a polygon

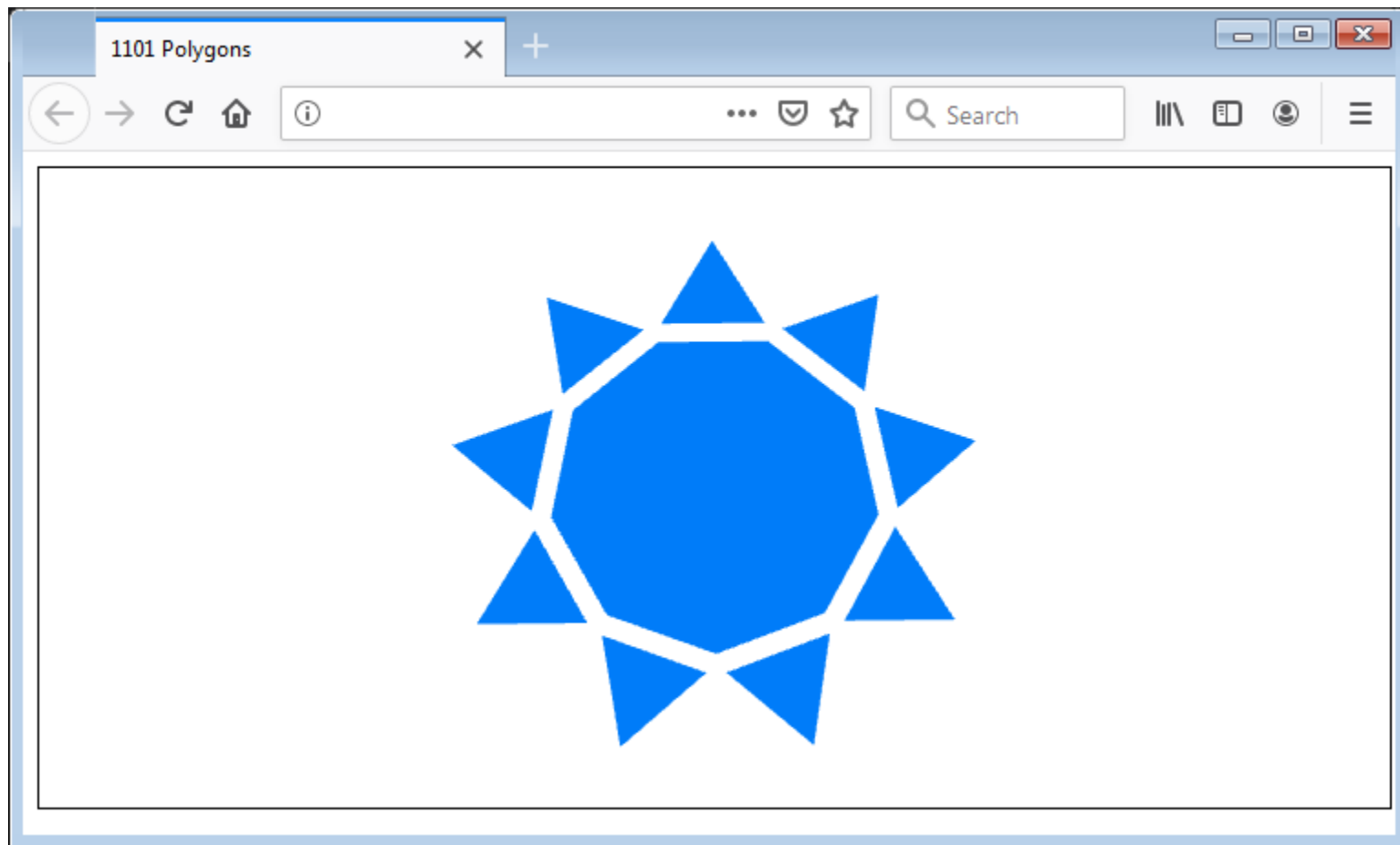
- With class `new Suica.Polygon (center, radius, count)`
- With function `polygon (center, radius, count)`
- Center is coordinates of a point, an array of three numbers
- Radius is a number, the radius of the circumflex circle
- Count is an integer number for the number of sides

Example

- Create an n-gon
- Add n triangles next to its sides

```
n = 9;
polygon([0,0,0],4,n);

for (var i=0; i<n; i++)
{
    a = (i+0.5)/n*2*Math.PI;
    b = polygon([5*Math.cos(a),5*Math.sin(a),0],1.5,3);
    b.spin = -a;
}
```



TRY IT

Circle and ellipse

Circle in Suica



Circle

- Graphical object with properties
- Used to draw circles

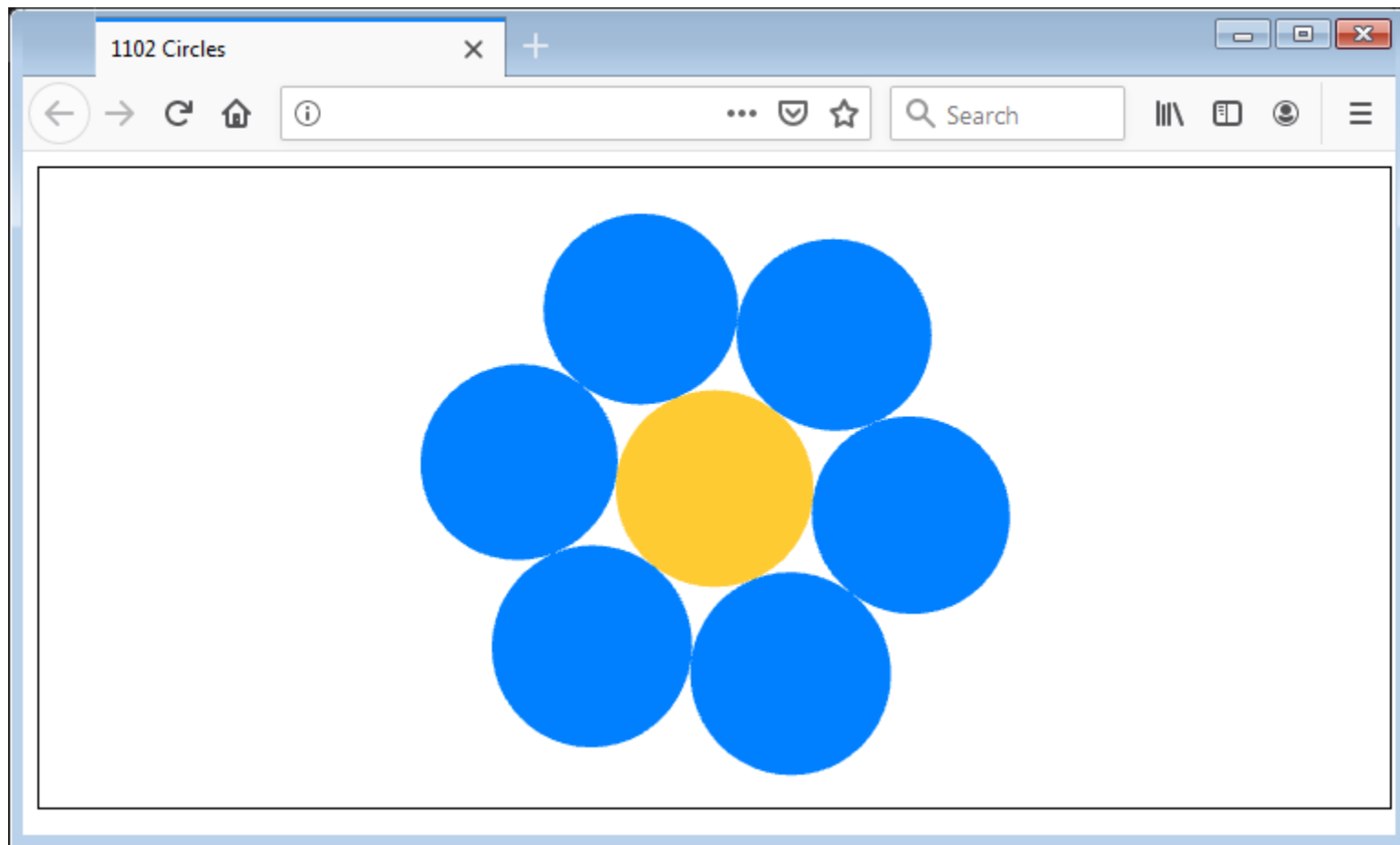
Creating a circle

- With class `new Suica.Circle (center, radius)`
- With function `circle (center, radius)`
- Center is coordinates of a point, an array of three numbers
- Radius is a number

Example

- Circle with 6 touching circle

```
a = circle([0,0,0],2.5);  
a.color = [1,0.8,0.2];  
  
for (var i=0; i<6; i++)  
{  
    a = 2*Math.PI*i/6;  
    circle([5*Math.cos(a),5*Math.sin(a),0],2.5);  
}
```

TRY IT

Ellipse in Suica



Ellipse

- Graphical object with properties
- Used to draw ellipses and circles

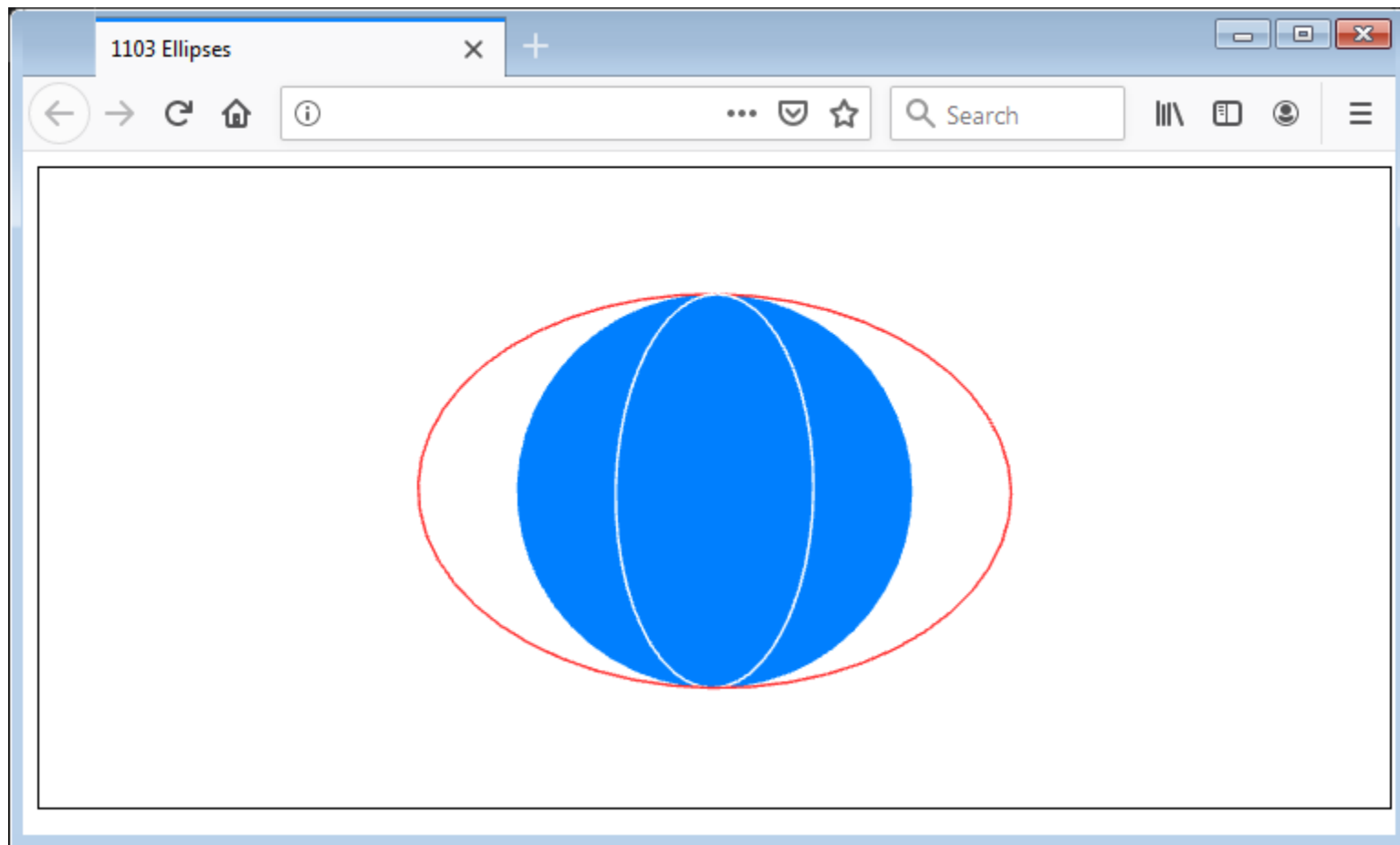
Creating an ellipse

- With class `new Suica.Ellipse (center, radii)`
- With function `ellipse (center, radii)`
- Center is coordinates of a point, an array of three numbers
- Radii is an array of two numbers

Example

- A circle with two ellipses
- One is outscribed, the other is inscribed

```
circle([0,0,0],5);  
  
a = ellipse([0,0,0],[7.5,5]);  
a.color = [1,0.2,0.2];  
a.mode = Suica.LINE;  
  
a = ellipse([0,0,0],[2.5,5]);  
a.color = [1,1,1];  
a.mode = Suica.LINE;
```



TRY IT

Sphere and spheroid

Sphere in Suica



Sphere

- Graphical object with properties
- Used to draw spheres and balls

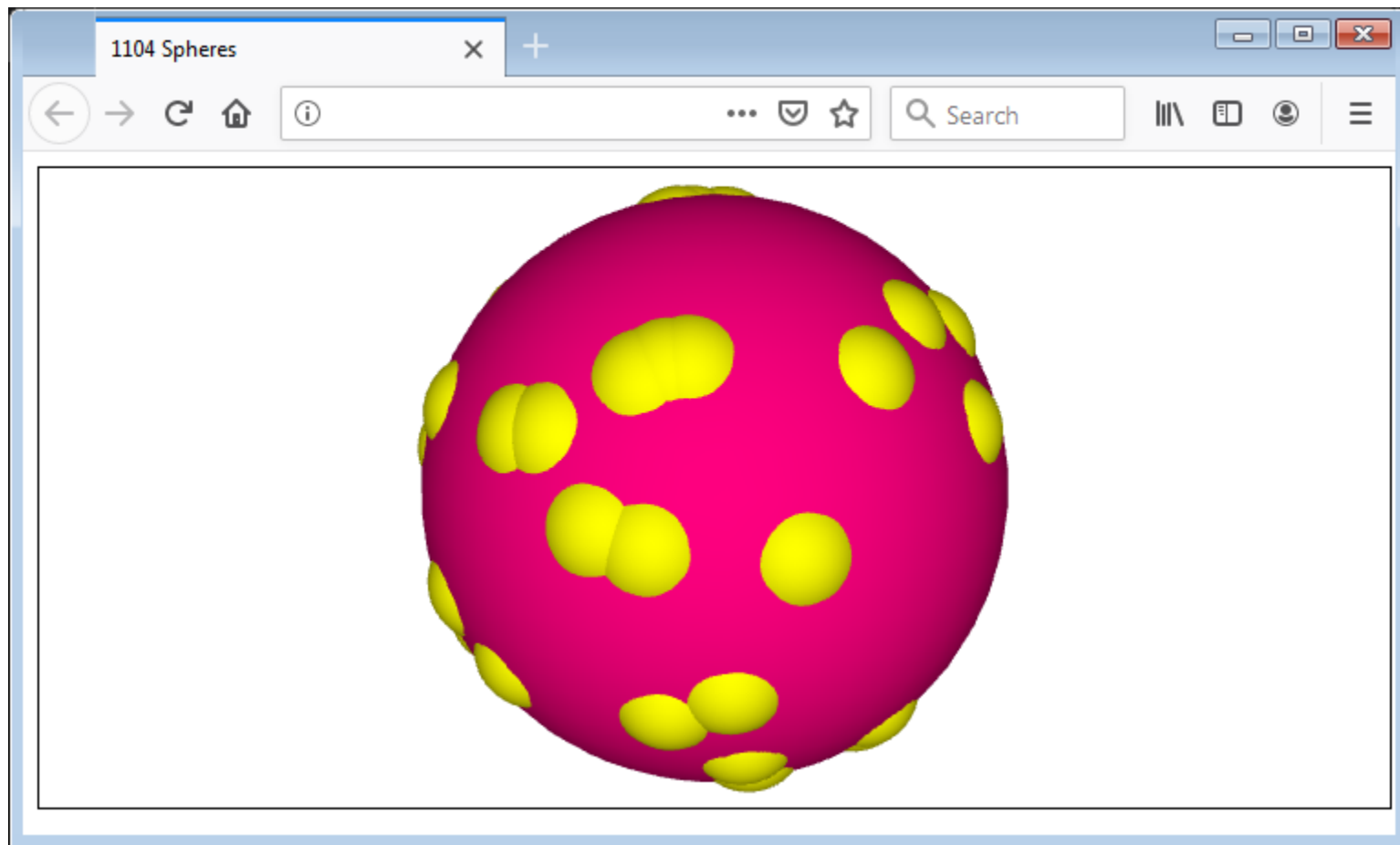
Creating a sphere

- With class `new Suica.Sphere (center, radius)`
- With function `sphere (center, radius)`
- Center is coordinates of a point, an array of three numbers
- Radius is a number

Example

- Big sphere
- Many small spheres submerged in it

```
sphere([0,0,0],5);  
  
for (var i=0; i<50; i++)  
{  
    a = sphere([0,0,0],1);  
    a.color = [1,1,0];  
    a.origin = [0,0,2.1];  
    a.focus = [random(-1,1),random(-1,1),random(-1,1)];  
}
```



TRY IT

Spheroid in Suica



Spheroid

- Graphical object with properties
- Used to draw spheroids and spheres

Creating a spheroid

- With class `new Suica.Spheroid (center, radii)`
- With function `spheroid (center, radii)`
- Center is the coordinates of a point, an array of three numbers
- Radii is an array of three numbers

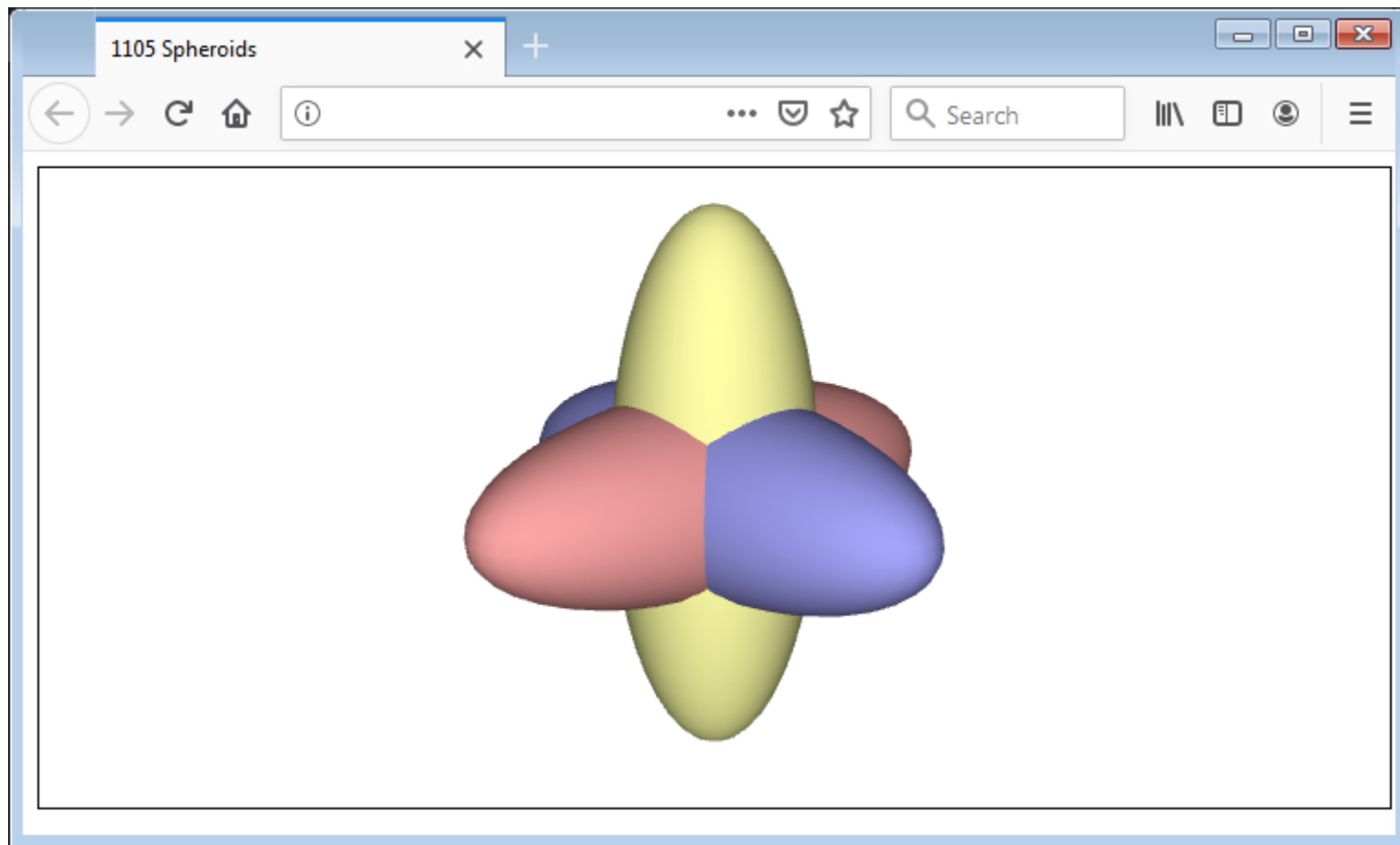
Example

- Three spheroids
- Each is along one of the axes

```
a = spheroid([0,0,0],[8,3,3]);  
a.color = [1,0.65,0.65];
```

```
a = spheroid([0,0,0],[3,8,3]);  
a.color = [0.65,0.65,1];
```

```
a = spheroid([0,0,0],[3,3,8]);  
a.color = [1,1,0.65];
```



TRY IT

Prism and pyramid

Prism in Suica



Prism

- Graphical object with properties
- Used to draw regular prisms

Creating a prism

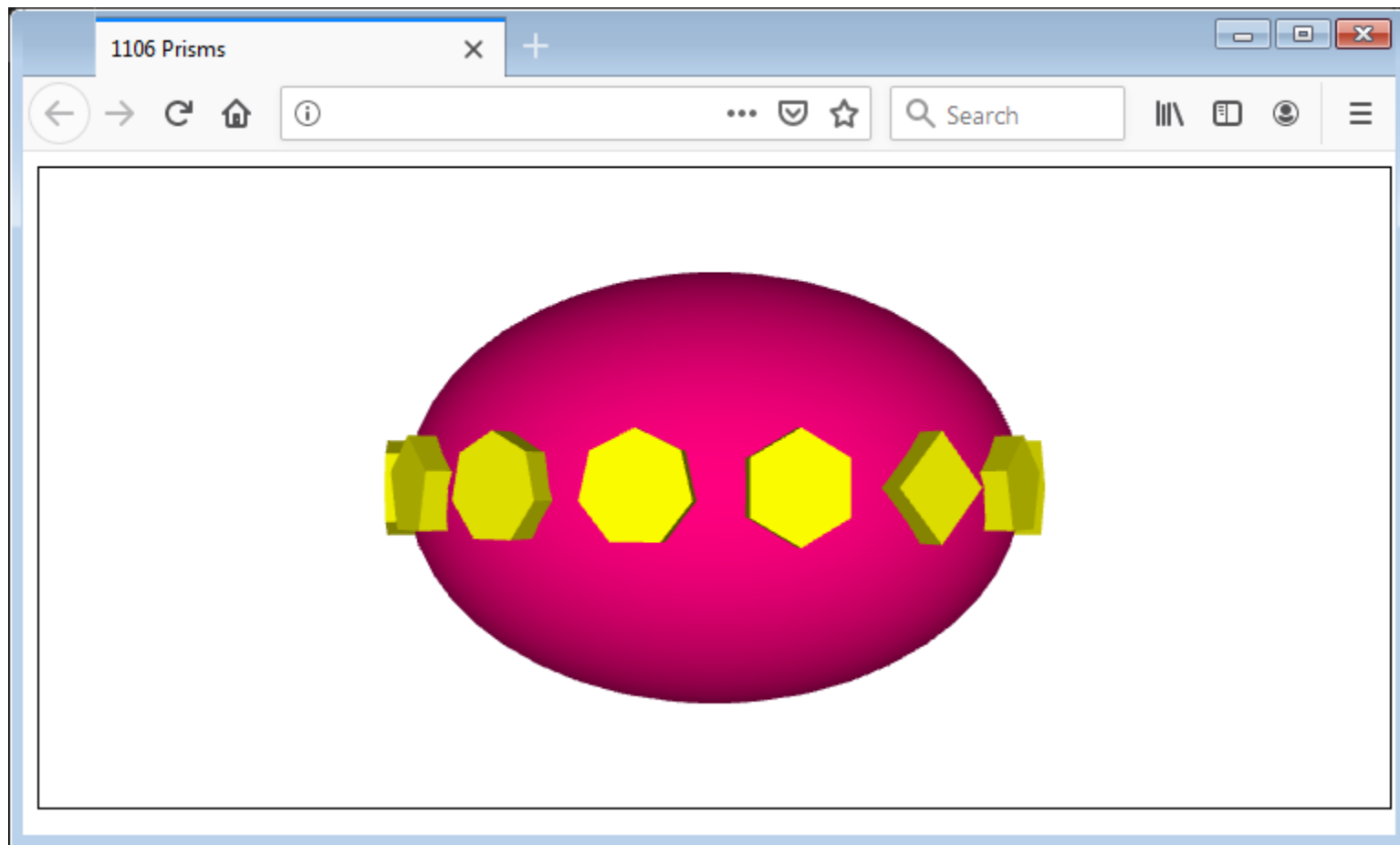
- With class `new Suica.Prism (center, radius, height, count)`
- With function `prism (center, radius, height, count)`

- Center is the coordinates of a point, an array of three numbers, the center of the base
- Radius is a number for the circumcircle of the base
- Height is a number for the prism height
- Count is the number of sides

Example

- Create 16 random prisms with random number of sides
- The prisms are position radially

```
spheroid([0,0,0],[5,5,3.5]);  
for (var i=0; i<16; i++)  
{  
    a = radians(360*i/16);  
    b = prism([Math.cos(a),Math.sin(a),0],0.75,4.3,  
              Math.round(random(3,8)));  
    b.color = [1,1,0];  
    b.focus = b.center;  
    b.spin = Math.PI;  
}
```



TRY IT

Pyramid in Suica



Pyramid

- Graphical object with properties
- Used to draw regular pyramids

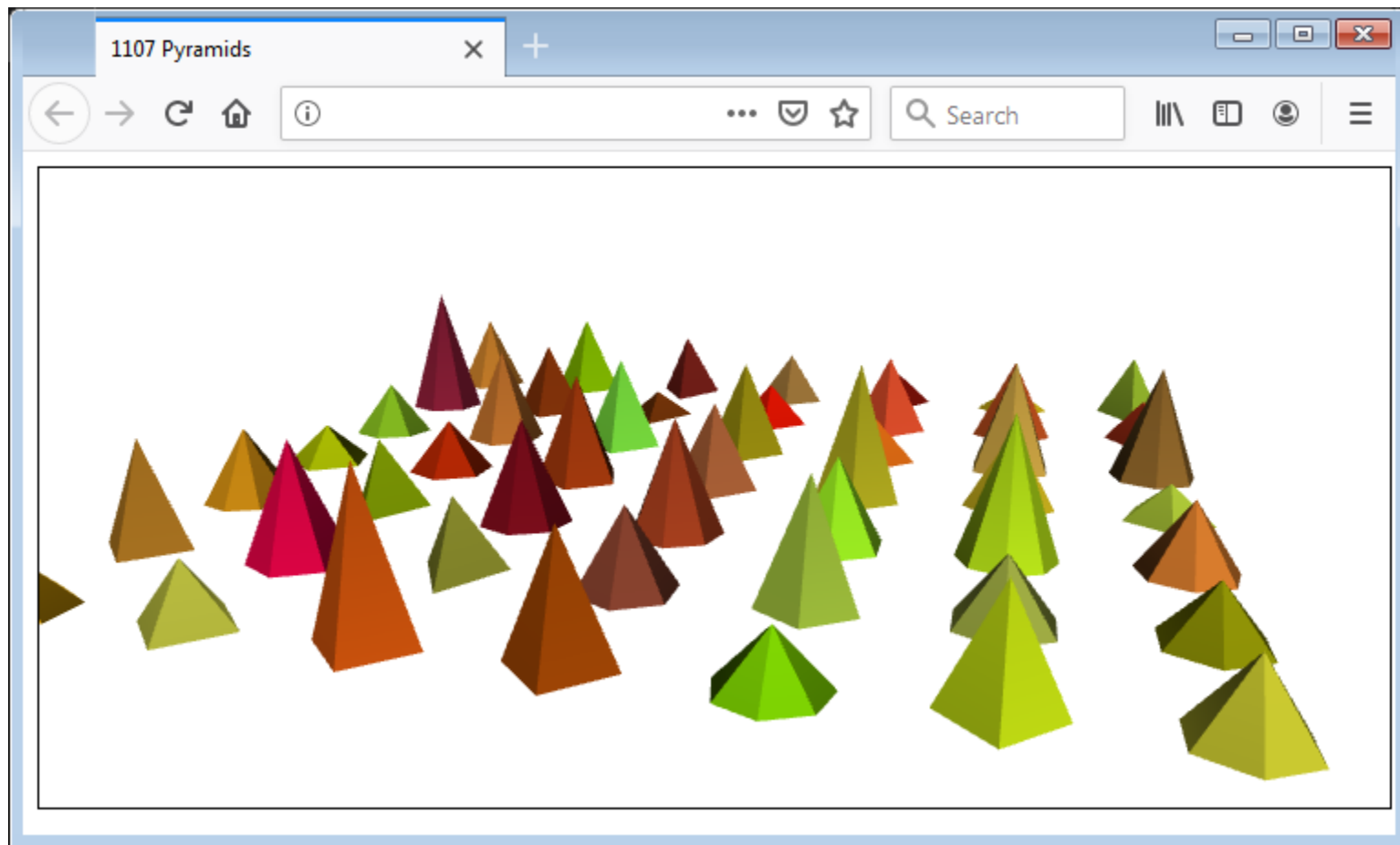
Creating a pyramid

- With class `new Suica.Pyramid (center, radius, height, count)`
- With function `pyramid (center, radius, height, count)`
- All properties are the same as in the prism

Example

- A matrix of random regular pyramids
- Various heights, colours and number of sides

```
for (var x=-3; x<4; x++)  
for (var y=-3; y<4; y++)  
{  
    a = pyramid([5*x,5*y,0],1.5,random(1,5),  
                Math.round(random(3,8)));  
    a.color=[random(0.5,1),random(0,1),random(0,0.3)];  
}
```



TRY IT

Cylinder and cone

Cylinder in Suica



Cylinder

- Graphical object with properties
- Used to draw cylinders

Creating a cylinder

- With class `new Suica.Cylinder (center, radius, height)`
- With function `cylinder (center, radius, height)`
- All properties are the same as in the prism except for count

Elliptical cylinder

- Graphical object with properties
- Used to draw cylinders with elliptical bases

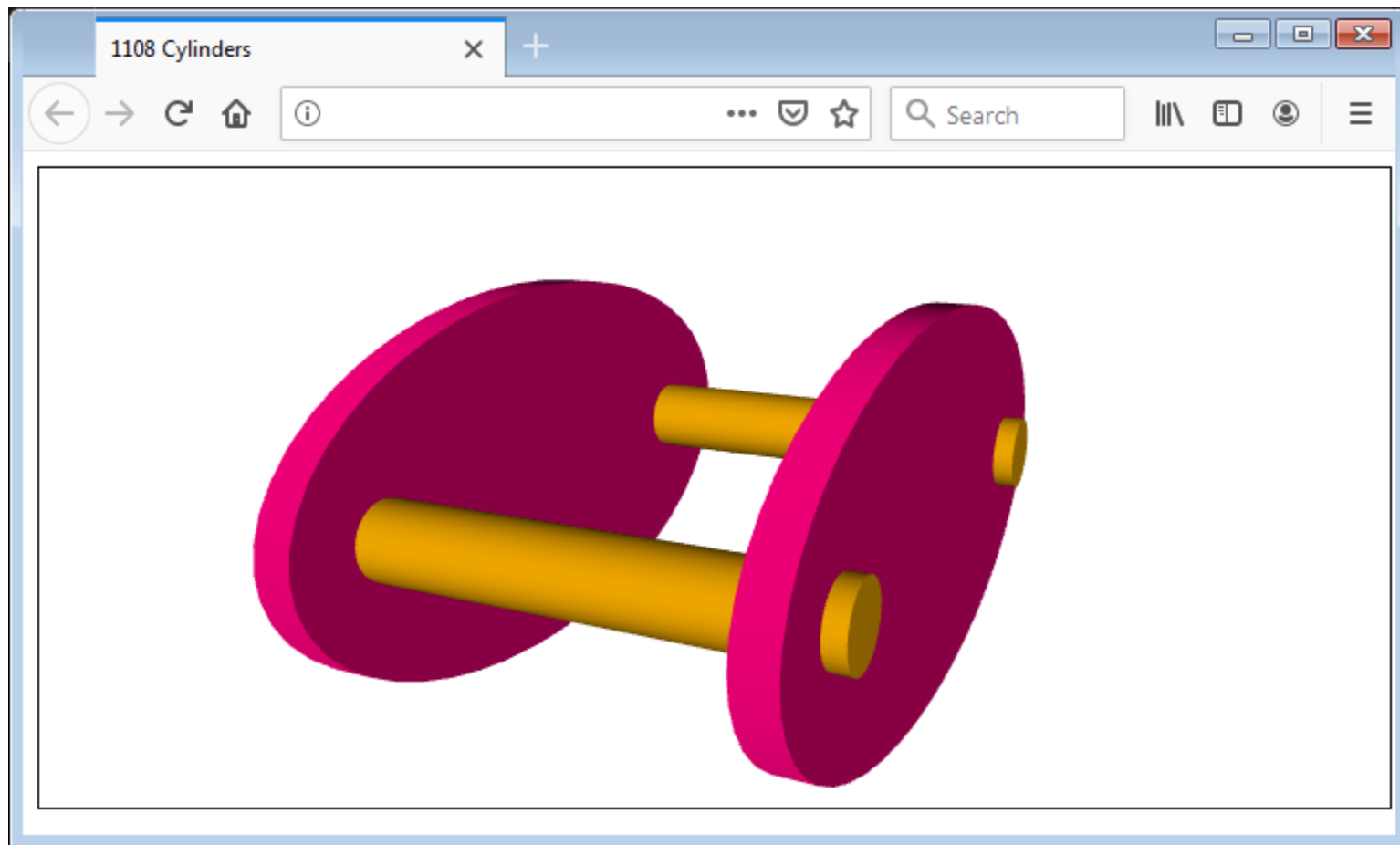
Creating elliptical cylinder

- With class `new Suica.Cylindroid (center, radii, height)`
- With function `cylindroid (center, radii, height)`
- All properties are as in the cylinder, except for the radii, which is an array of two numbers – the radii of the base ellipse

Example

- Two elliptical plates
- Linked by two cylinders

```
a = cylindroid([0,5,0],[5,10],1);  
a.focus = [0,1,0];  
a = cylindroid([0,-5,0],[5,10],1);  
a.focus = [0,-1,0];  
  
a = cylinder([7,-6.5,0],1,13);  
a.focus = [0,1,0];  
a.color = [1,0.7,0];  
a = cylinder([-7,-6.5,0],1,13);  
a.focus = [0,1,0];  
a.color = [1,0.7,0];
```



TRY IT

Cone in Suica



Cone

- Graphical object with properties
- Used to draw cones

Creating a cone

- With a class `new Suica.Cone (center, radius, height)`
- With function `cone (center, radius, height)`
- All properties are as in the cylinder

Elliptical cone

- Graphical object with properties
- Used to draw cones with elliptical bases

Creating an elliptical cone

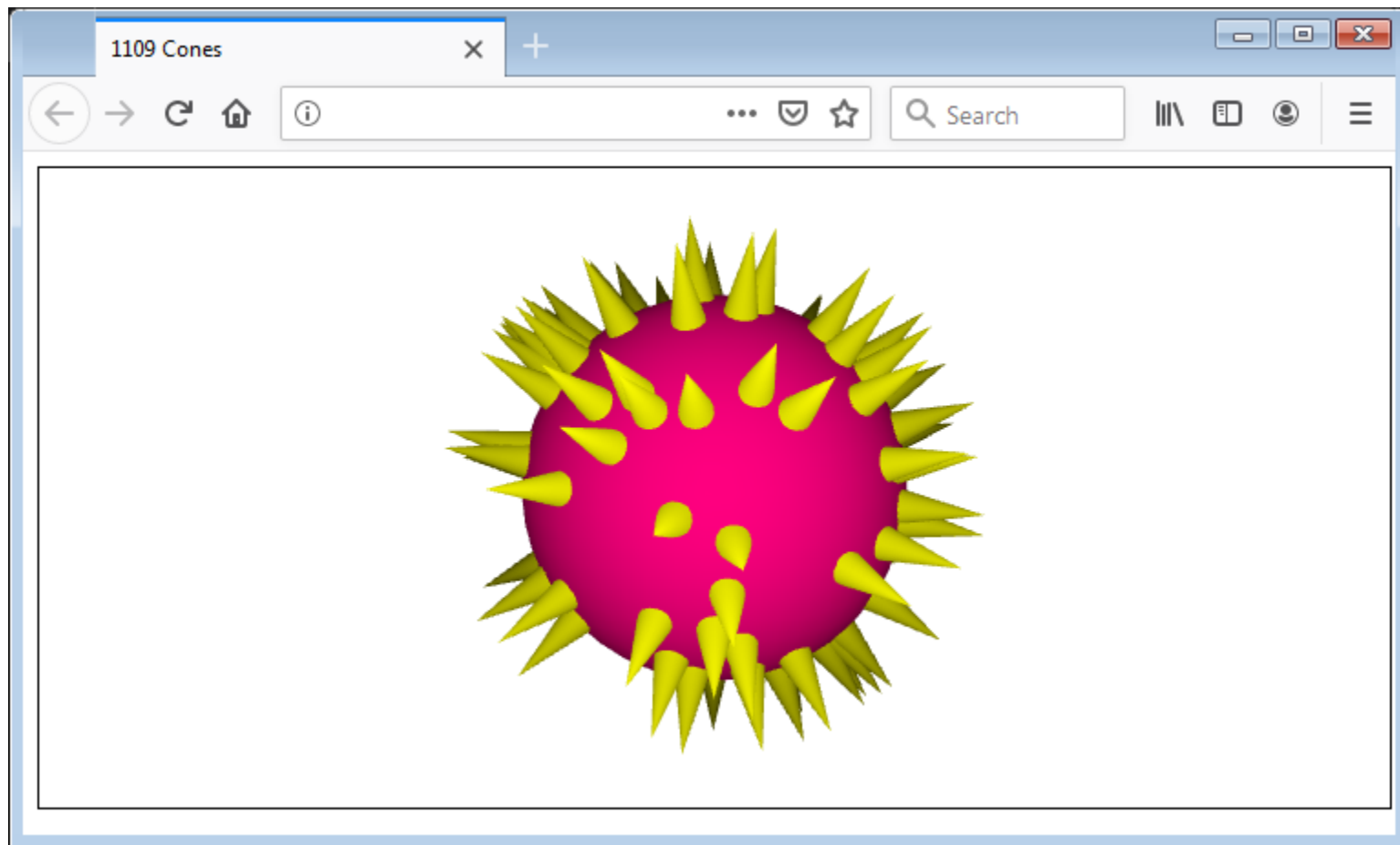
- With class `new Suica.Conoid (center, radii, height)`
- With function `conoid (center, radii, height)`
- Properties are as in the cylinder, except for radii, which is an array of two numbers – radii of the elliptical base

Example

- Sphere with spikes
- Oriented randomly

```
sphere([0,0,0],5);

for (var i=0; i<100; i++)
{
    a = cone([0,0,0],1.5,7);
    a.color = [1,1,0];
    a.focus = [random(-1,1),random(-1,1),random(-1,1)];
}
```



TRY IT



Additional property



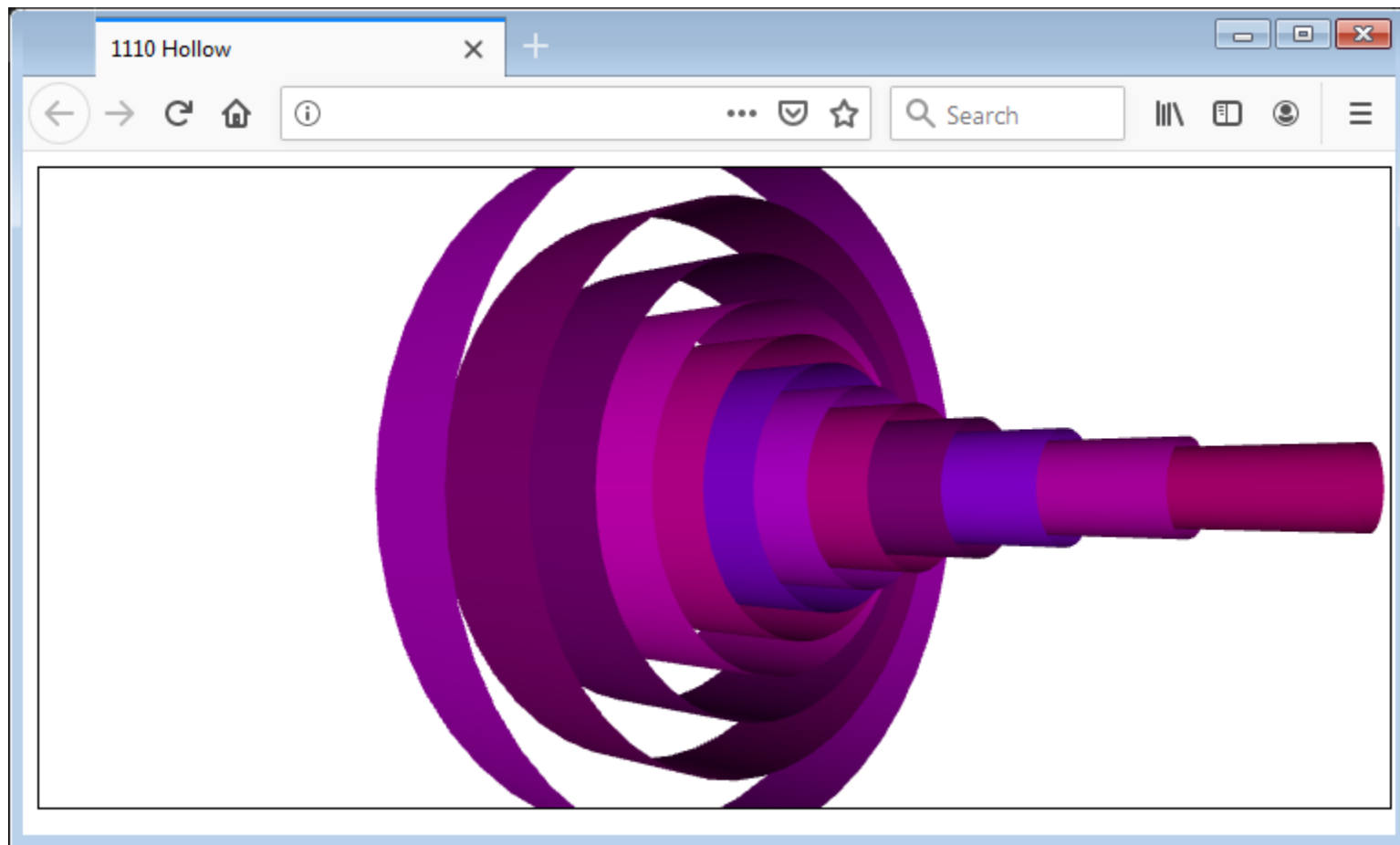
Property hollow

- Defines whether to draw the bases of the object
- Boolean value, by default it is false
- If it is true, the bases are not drawn, if it is false – they are drawn
- Used for objects with bases (e.g. prism, pyramid, cylinder, cone)
- Used to model “hollow” objects

Example

- A group of concentric tubes with common axis
- Long tubes are narrow, short are wide

```
h = 20;  
r = 1/2;  
for (var i=0; i<12; i++)  
{  
  a = cylinder([0,-h/2,0],r,h);  
  a.focus = [0,1,0];  
  a.hollow = true;  
  a.color = [random(0.5,1),0,random(0.5,1)];  
  h = h*0.8;  
  r = r/0.8;  
}
```





Examples

Example №1



Fire eye

- Draw a fire eye – two red frames
- Red halo around them

Idea

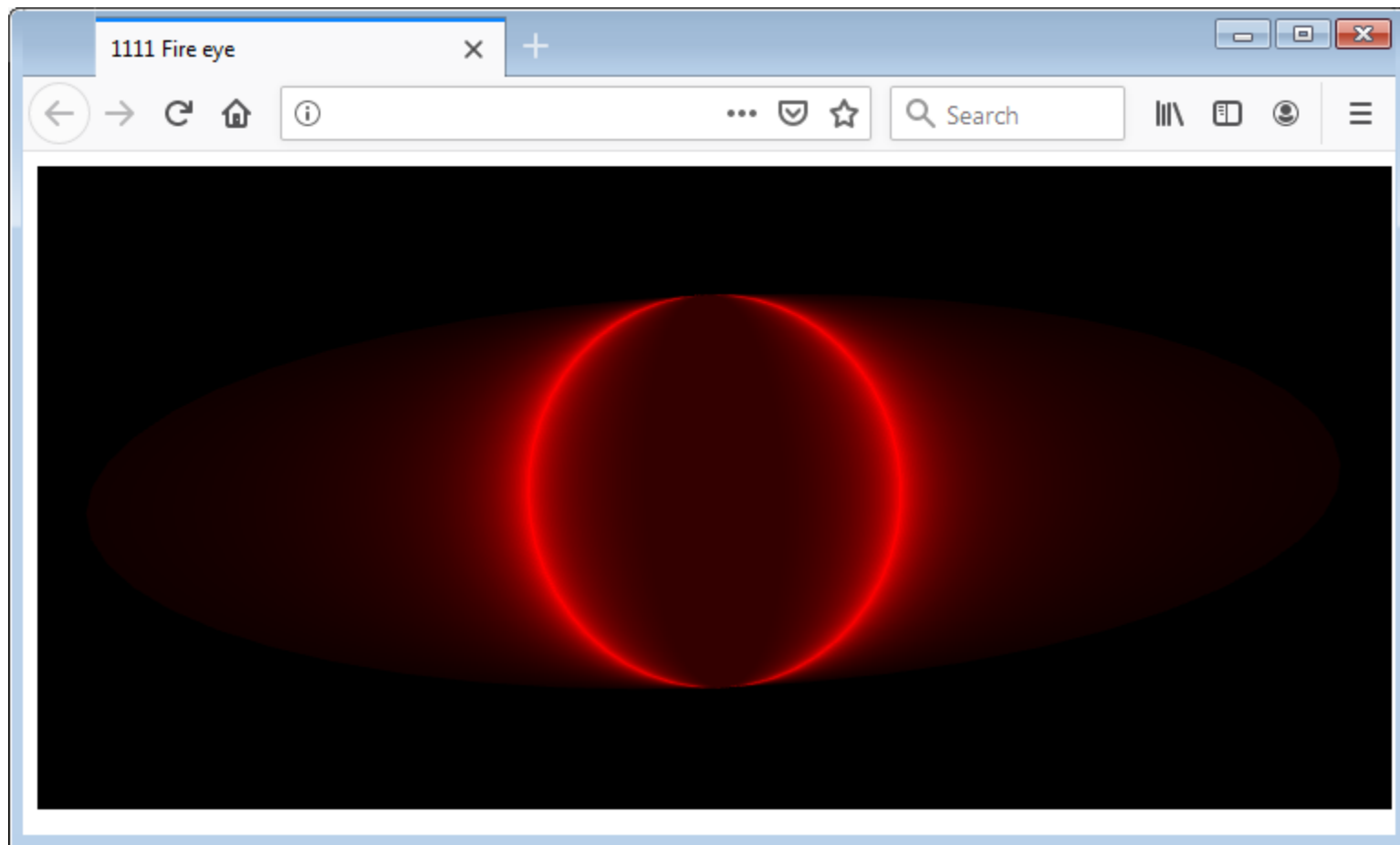
- Starting with a narrow ellipse
- Creating other ellipses, which gradually become wider
- One of the ellipses will be red, the others are shades between red and black

Implementation

- Black background, merges with the darkest ellipses
- One of the radii is fixed, the other is variable
- The level of redness is maximal at $i=5$, when the ellipse proportion is 10×9.5

```
background([0,0,0]);

n = 20;
for (var i=1; i<=n; i+=0.1)
{
    a = ellipse([0,0,-i/100],[10,2+1.5*i]);
    a.color = [1/(1+Math.abs(5-i)),0,0];
}
```



TRY IT

Example Nº2



Japanese pillow

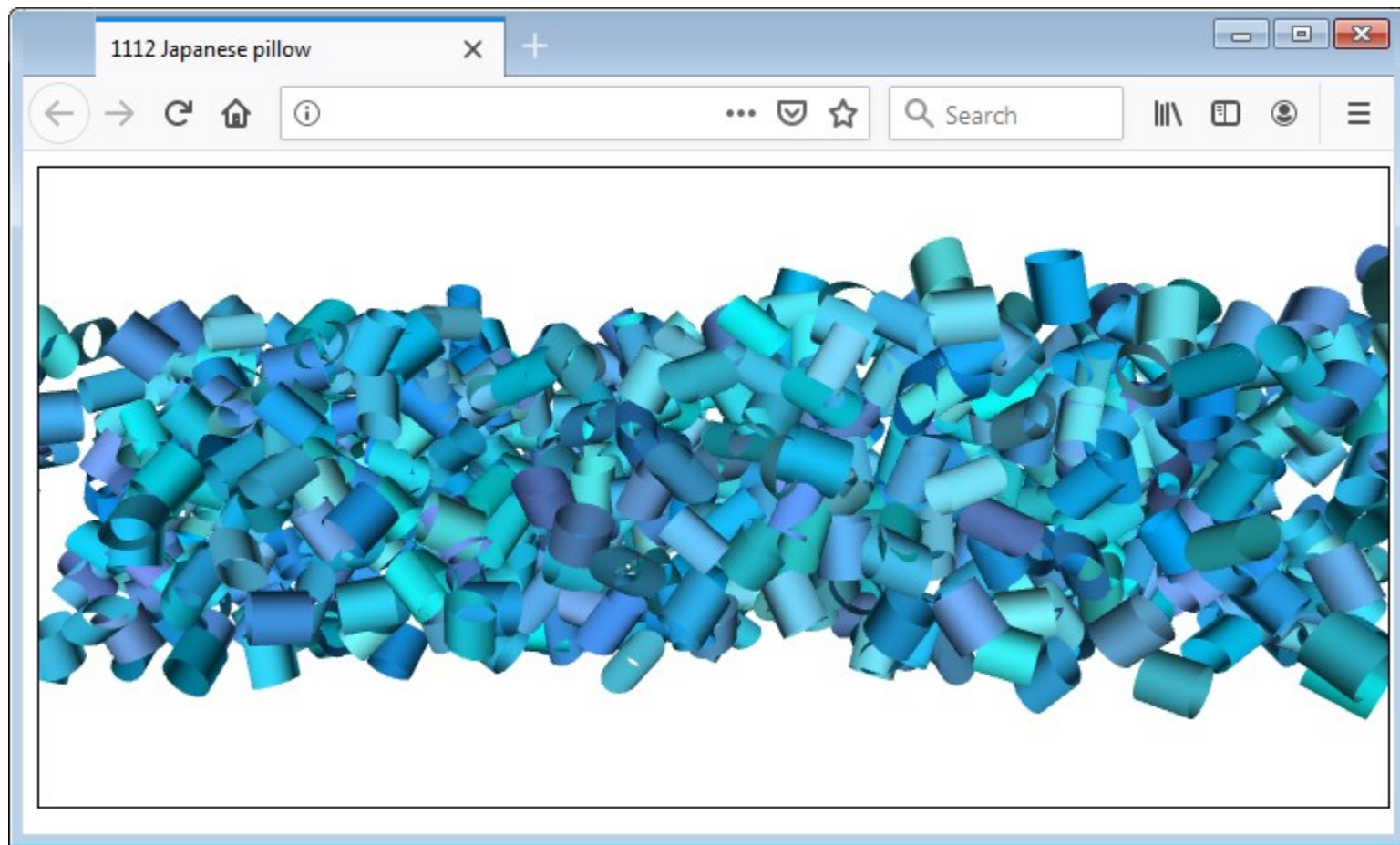
- Design a model of its interior



Implementation

- Multiple short hollow cylinders
- All are slightly flattened and bluish
- Random coordinates and orientation
- Help function $r(x)$ return a random number from $-x$ to x

```
for (var i=0; i<n; i++)  
{  
    a = cylindroid([r(20),r(10),r(5)], [0.75,0.5], 1.5);  
    a.color = [random(0,0.5),random(0.6,1),1];  
    a.spin = random(0,2*Math.PI);  
    a.focus = [r(1),r(1),r(1)];  
    a.hollow = true;  
}
```



TRY IT



Summary

Graphical objects



Polygon

- Created with `new Suica.Polygon` or `polygon`
- Has `center`, `radius` and `count` sides
- Supports `mode`, `origin`, `spin`, `focus` and `light`

Circle

- Created with `new Suica.Circle` or `circle`
- Has `center` and `radius`
- Supports polygon's properties except `count`

Ellipse

- Created with `new Suica.Ellipse` or `ellipse`
- Has `center` and two `radii`
- Supports the other circle's properties

Sphere

- Created with `new Suica.Sphere` or `sphere`
- Has `center` and `radius`
- Supports the other circle's properties

Spheroid

- Created with `new Suica.Spheroid` or `spheroid`
- Has `center` and three `radii`
- Supports the other sphere's properties

Regular prism

- Created with `new Suica.Prism` or `prism`
- Has `center`, `radius`, `height` and number of `count` sides
- Supports properties `mode`, `origin`, `spin`, `focus` and `light`

Regular pyramid

- Created with `new Suica.Pyramid` or `pyramid`
- Has the same properties as the prism

Cylinder

- Created with `new Suica.Cylinder` or `cylinder`
- Has `center`, `radius` and `height`
- Support properties `mode`, `origin`, `spin`, `focus` and `light`

Elliptical cylinder

- Created with `new Suica.Cylindroid` or `cylindroid`
- Has `center`, `radii` and `height`
- Supports the same properties as the cylinder

Cone

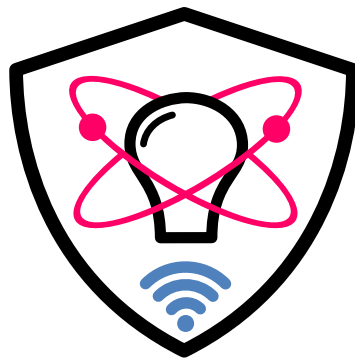
- Created with `new Suica.Cone` or `cone`
- Has `center`, `radius` and `height`
- Supports the same properties as the cylinder

Elliptical cone

- Created with `new Suica.Conoid` or `conoid`
- Has `center`, `radii` and `height`
- Supports the same properties as the elliptical cylinder

Common properties

- **count** – number of sides of objects
- **radius** – radius of objects or their bases
- **radii** – radius of objects or their bases
- **hollow** – flag for drawing the bases of objects



ICT in SES

End

Comments, questions