Chapter 19:

In this chapter, we are going to further elaborate on the distribution of points on faces by creating a donut with sprinkles. For this chapter's project make sure to turn on the Timing overlay.

Download the two Chapter 19 files from https://github.com/rbarbosa51/GeometryNodesByTutorials/tree/main/Chapter19. Open the Chapter19Start.blend file so you can follow along.

Connect a **Join Geometry(3)** node between the **Group Input(1)** and **Group Output(2)** nodes. Create a secondary connection between the **Group Input(1)** and the **Join Geometry(3)** nodes by placing two reroutes (Figure 19-1). On the top connection add a **Set Material(4)** node, and set its material to *Dough*. Connect a **Set Shading Smooth(5)** in between the **Join Geometry(3)** and the **Group Output(2)** nodes. Your node tree should resemble Figure 19-2. You should see a smooth torus type object (Figure 19-3). That is going to be the donut.

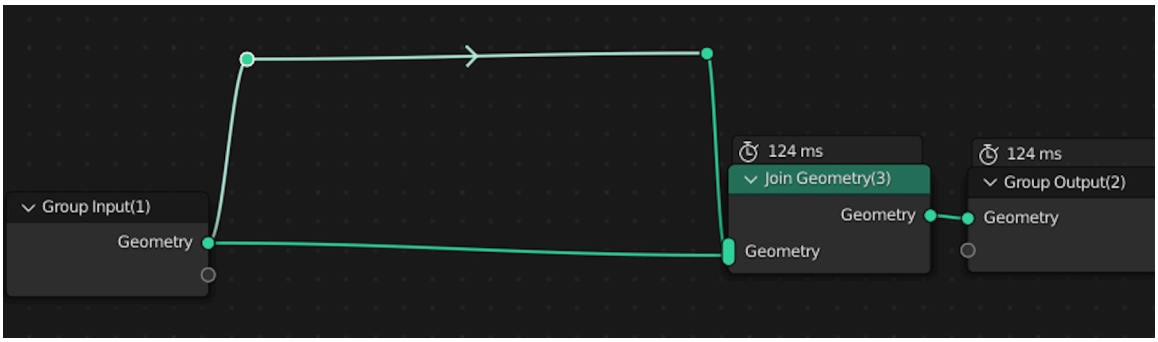

Figure 19-1

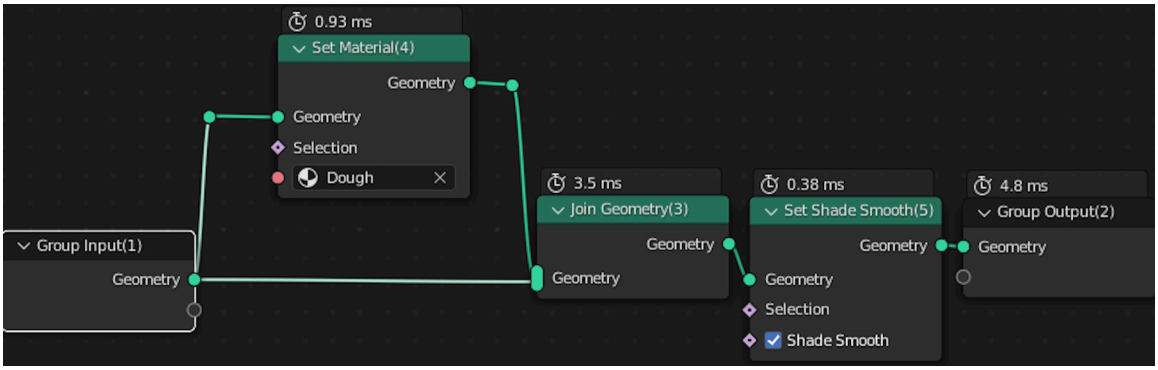

Figure 19-2

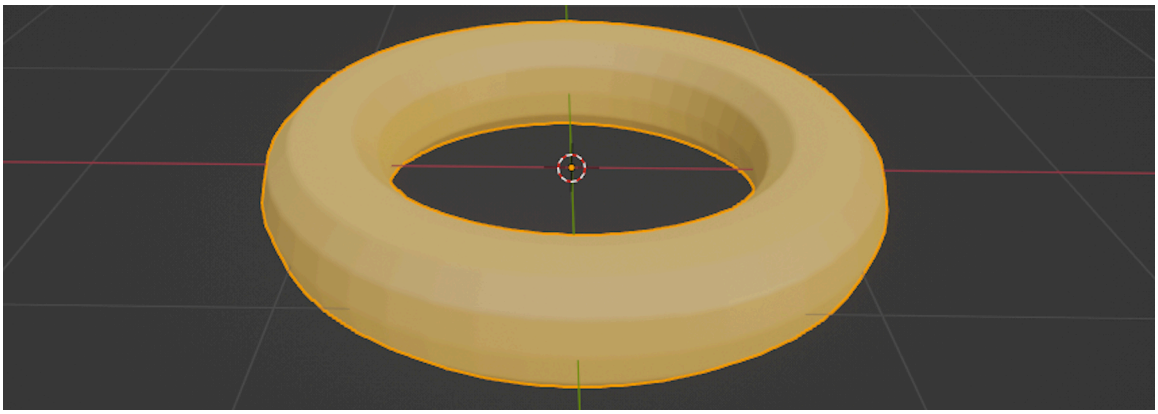

Figure 19-3

Connect a **Distribute Points on Faces(6)** node to the bottom connection between the **Group Input(1)** and the **Join Geometry(3)** nodes. Drag a connection between the *Density* socket of the **Distribute Points on Faces(6)** to an empty socket in the **Group Input(1)** node. Connect an **Instance on Points(7)** node in between the **Distribute Points on Faces(6)** and the **Join Geometry(3)** nodes. Connect an **Ico Sphere(8)** node to the *Instance* socket of the **Instance on Points(7)** node. Change the **Ico Sphere(8)'s** *Radius* to 0.02 (Figure 19-4).

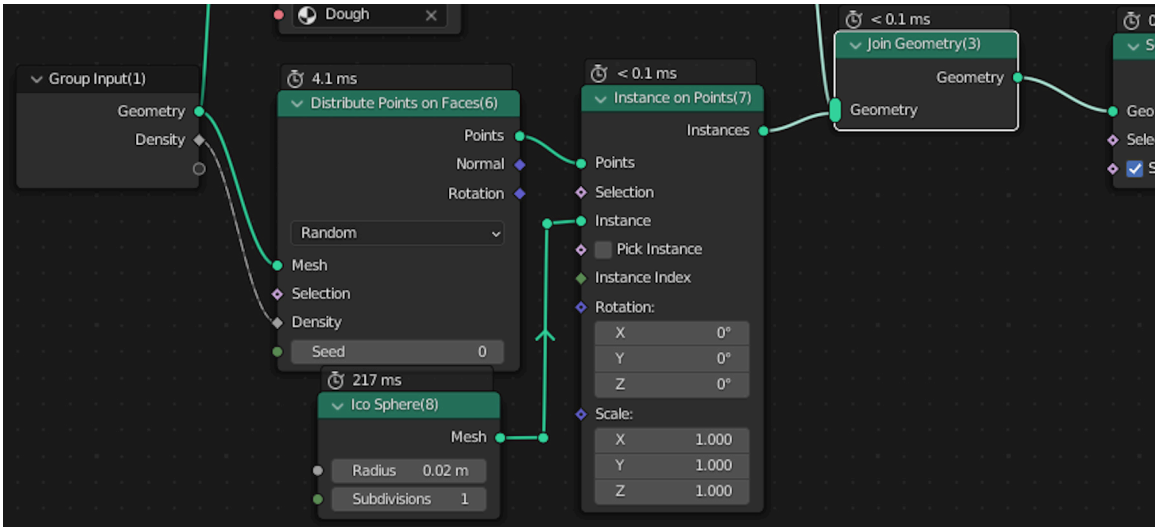

Figure 19-4

Change the value of the Density Interface to 3000. Pay attention to your computer, if it becomes slow, lower you settings to 1000. You should see that the Donut is completely surrounded by sprinkles (Figure 19-5). Grab a **Random Value(9)** node, and change its type to *Vector*. For its values, the minimum X, Y, and Z are going to be 0.2 and maximum values are 1.2. This makes sure that the sprinkles have random varying sizes. Connect the **Random Value(9)'s** outbound *Value* socket to the **Instance on Points(7)'** *Scale* socket. You can minimize the **Random Value(9)** node.

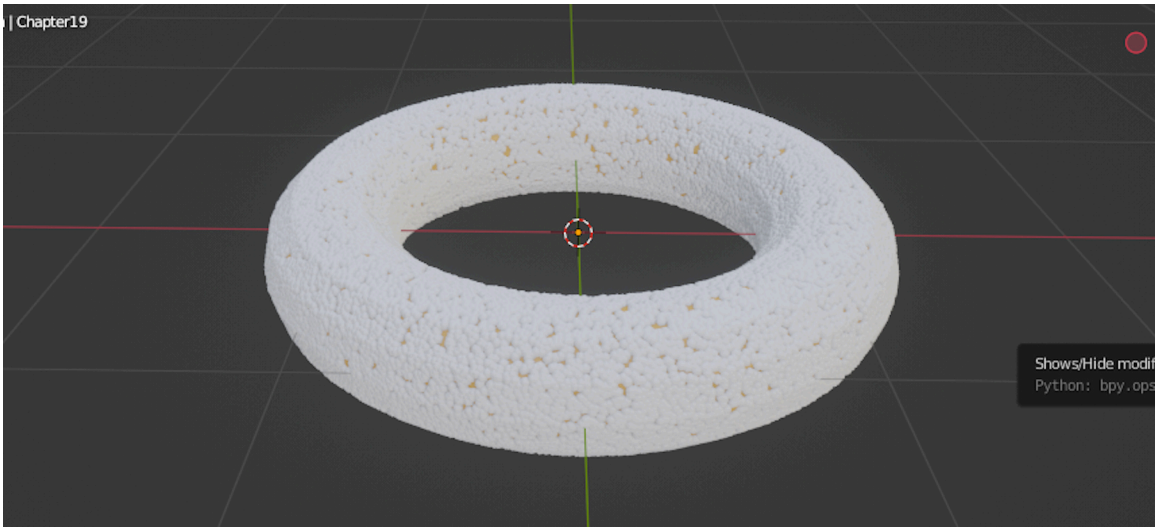

Figure 19-5

This is the tricky part. We want to only have sprinkles on the top of the donut, not the bottom or the sides. Therefore we need to do some math. We will use the **Vector Math** node *Dot Product* function. While the topic of linear algebra is outside the scope of this book, this simple concept should suffice. The dot product function produces a scalar value based on two vectors. If two vector are perpendicular (90 degrees) their value is zero. If they are facing in a similar direction the value is positive. If they are facing an opposite direction, the dot product is a negative value.

The **Distribute Points on Faces** node takes in a *Selection* boolean socket. As previously stated, the way that blender converts values to boolean is as follows: zero is false, positive numbers are true, and negative numbers are false. Try the following experiment. Drag a **Value(10)** node connect it to the *Selection* socket of the **Distribute Points on Faces** node. Set the *Value* on the **Value(10)** node to 0, -1, 1, and random values in between. We are going to take advantage by using the dot product on the faces normals.

All faces have a normal. That is the direction that the face is facing. To view all of the objects normals, go to the Layout workspace, then change to Edit mode. Once on Edit mode, under overlays, turn on Display Normals (Faces) and set its size to 0.3. For clarity, place the viewport in Front Orthographic view. You will see all of the Normals facing outwards. We will use the Dot product on the up direction (0,0,1), in other words, relative to the up direction all of the faces facing upwards up to 90 degrees will have a positive value. See Figure 19-6. Change the mode to Object mode. Go back to the Geometry Nodes workspace.

Figure 19-6

Disconnect the **Value(10)** node. You can delete or move it to the side. Grab a **Normal(11)** node and connect it to the first *Vector* value of a **Vector Math(12)** node. Change the **Vector Math(12)** mode to *Dot Product*. On the second *Vector* value of the **Vector Math** node set the value to: X:0.0, Y: 0.0, and Z: 1.0. In other words, it's looking for the dot product of the individual face normals relative to the up (0,0,1) direction. Connect the **Vector Math(12)** value output to the *Selection* socket of the **Distribute Points on Faces(6)** node (Figure 19-7). The result should look like Figure 19-8.

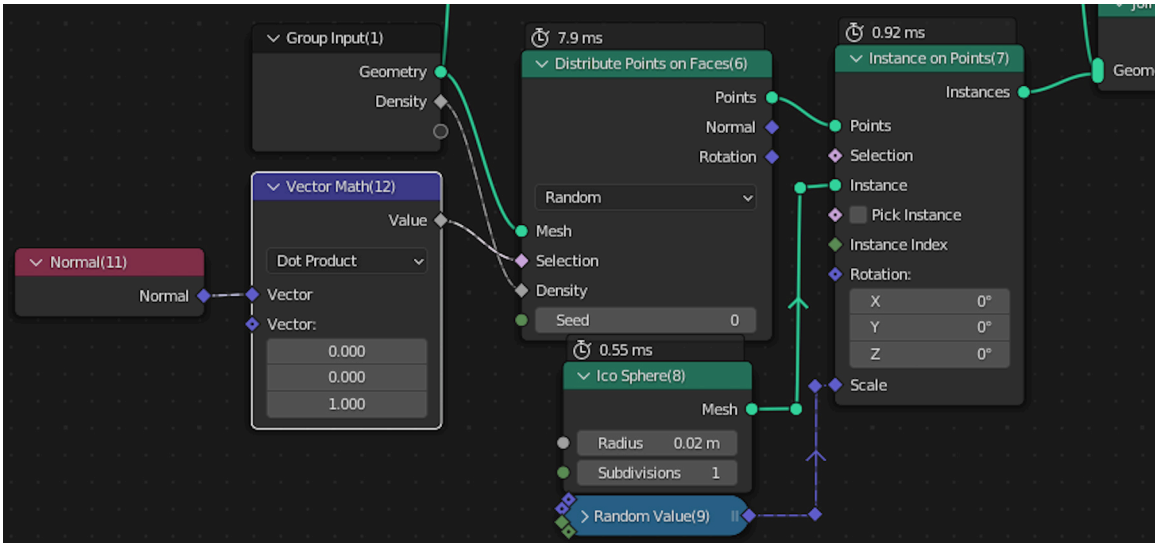

Figure 19-7

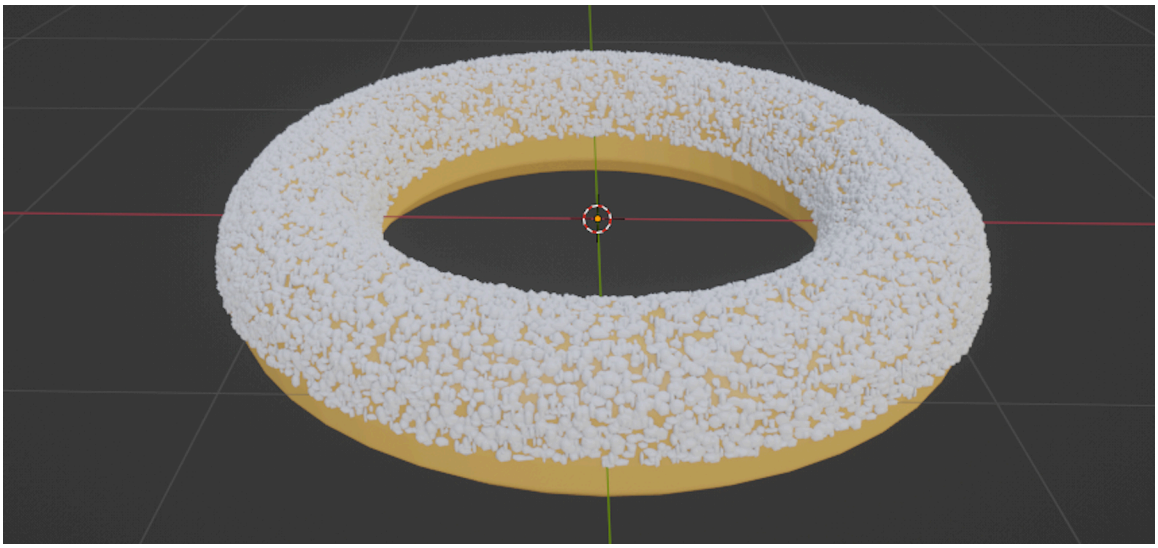

Figure 19-8

As you can see, the faces from up direction to the side direction are covered in points. However we want some more control. In between the **Normal(11)** and **Vector Math(12)** nodes, add another **Vector Math(13)** node. Set this new **Vector Math(13)** node to *Subtract*. If the **Vector Math(13)** node disconnects from the **Vector Math(12)**, redo the connection. Connect a **Combine XYZ(14)** node to the second *Vector* value of the **Vector Math(13)** node. Grab a new **Group Input(15)** node and drag a connection between the *Z* value of the **Combine XYZ(14)** node and an empty socket of the **Group Input(15)**. Change the name of the interface from Z to *Threshold*. This threshold will allow us to further limit surface of the points being distributed. Change the value of the Threshold interface to 0.7854 (45 degrees in radians). Observe the surface. Now change the value to 0.2618 (thats 15 degrees in radians). Look how the donut feels more realistic. Your node tree should resemble Figure 19-9 and result should look like Figure 19-10.

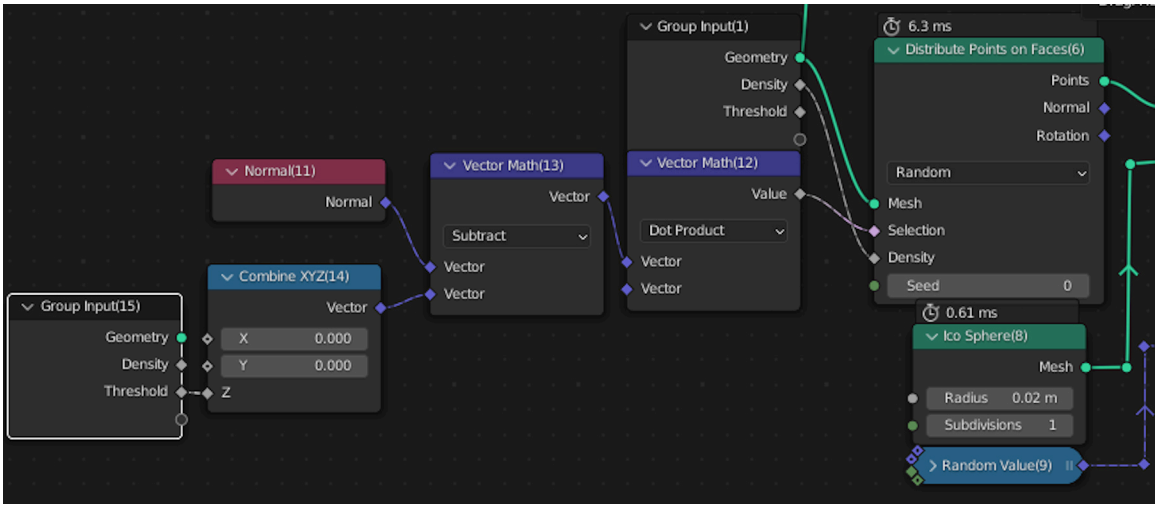

Figure 19-9

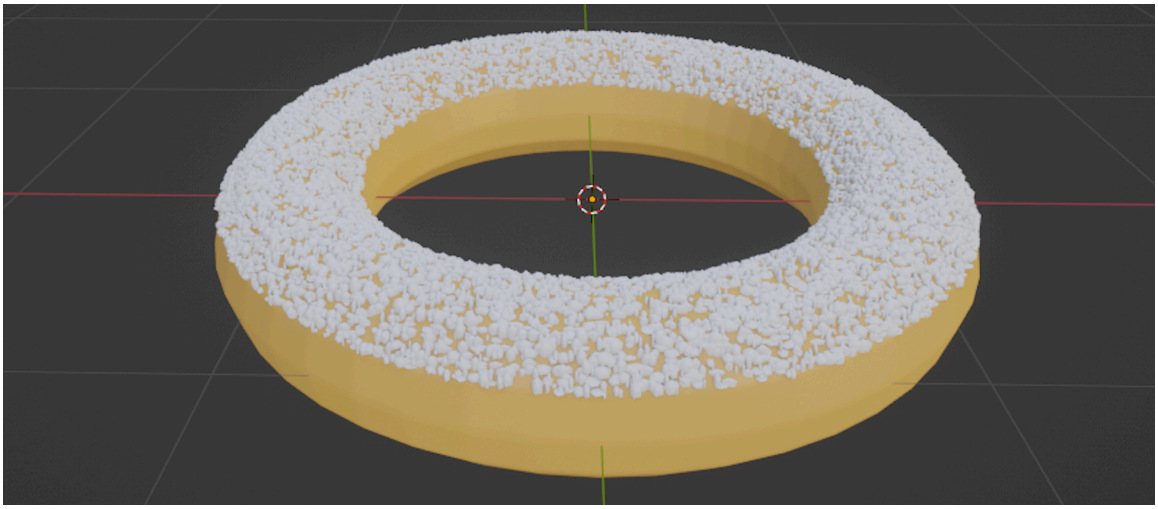

Figure 19-10

Now we need to add colors to the sprinkles. Go to the Shading Workspace. Look at the **Sprinkles** material (it should be pinned). While this book is not about Shading nodes, a simple explanation will suffice. This simple pre made material consist of an **Object Info**, **Color Ramp** and a **Principled BSDF** shader. For every instance the **Object Info** provides a random value between 0.0 and 1.0. This value is then processed as color in the color ramp, and subsequently passed to the **Principled BSDF** shader as the base color (Figure 19-11). This only works because the sprinkles are still instances, should you decide to realize the instances, they will all get asiigned a single random value.

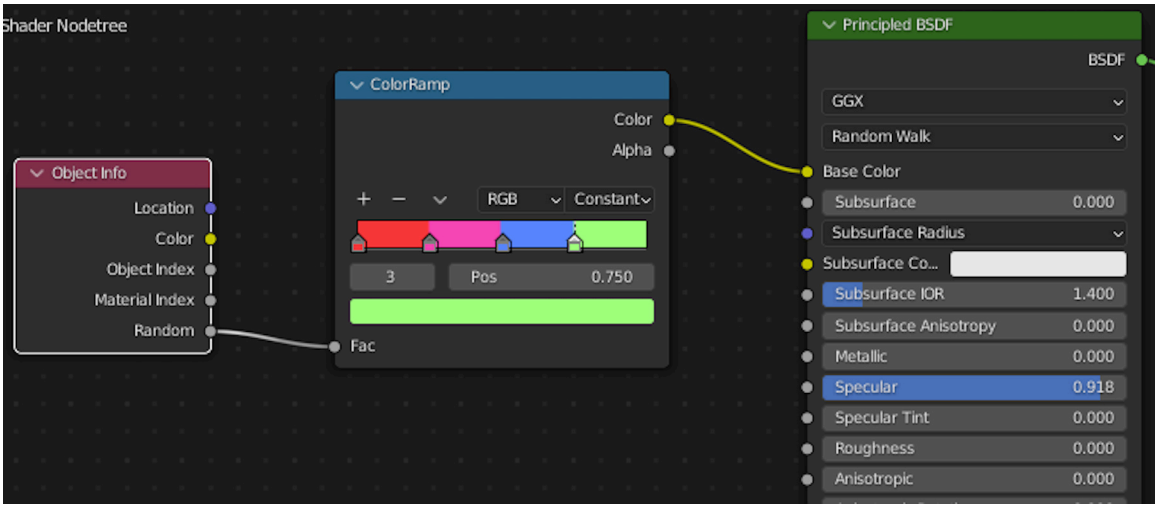

Figure 19-11

Go back to the Geometry Nodes workspace. Connect a **Set Material(16)** node in between the **Instance on Points(7)** and the **Join Geometry(3)** nodes. Set the material of the **Set Material(16)** to the pre made *Sprinkles* (Figure 19-12). Your donut should have different colored sprinkles. The results should look like Figure 19-13.

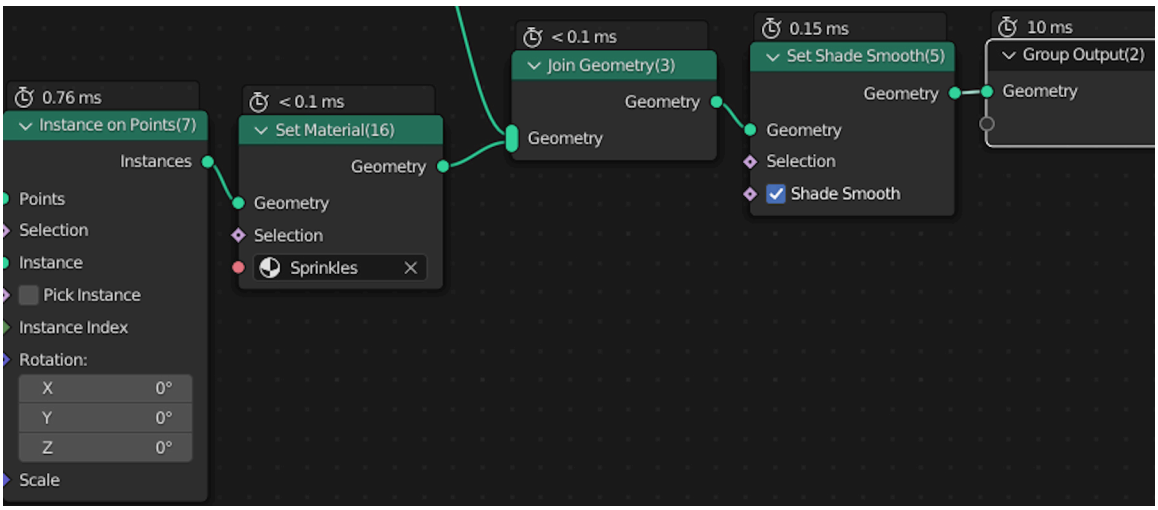

Figure 19-12

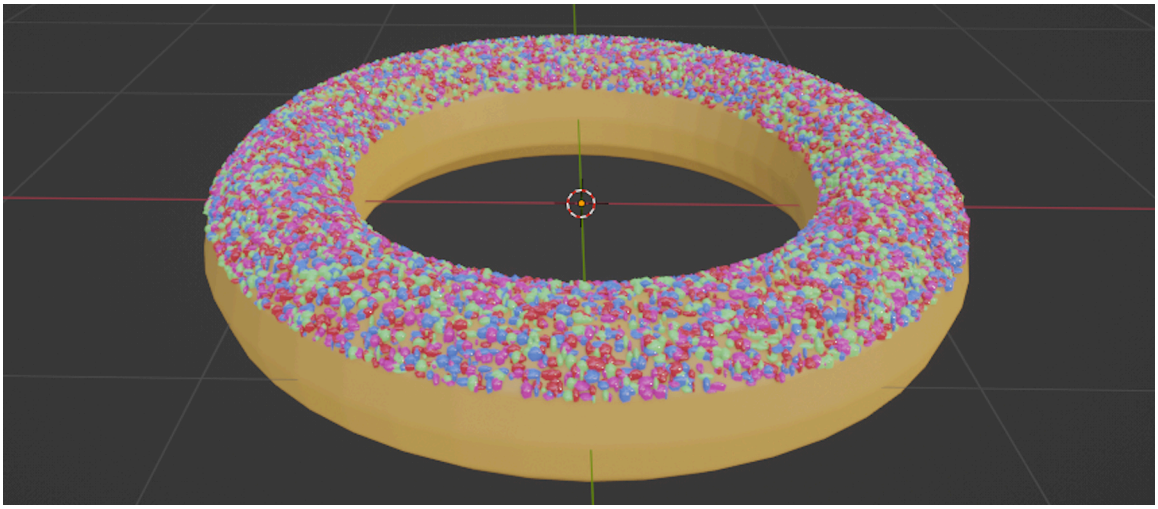

Figure 19-13

In this chapter, you learned the very basics of using using linear algebra concepts (in this case Dot Product) as the basis of instance selection. Feel free to experiment with the angles (in radians) to see different results. Be careful if you want to change the Density to higher values.

Compare your results and node tree with those of the downloaded Chapter19Final.blend file.