In this class, we are going to write the code to remove the fruit body from the rope, when the user will press the “cut” button and the fruit will then fall. We will also create the body for the bunny and add images for it as well as for the background.

We now need to load the images in our code By using a loadImage() and the preload() function

We load the assets in the preload() function because we want to load all the images before our main code begins, and the preload() function runs at first, all the other functions and instructions are executed after the preload function.

Load the image using the loadImage() function and pass the image path as the parameter within the loadImage() function. In this way, images will be loaded, and we will assign them to the variables we created earlier.

set the image mode as the center. With this, while displaying the image on the canvas, we will have the center point of the images.

By default, when we create the image on the canvas, it takes the top left corner as the origin point, but here we are changing it to the center, so now when we draw the image on the canvas, we will specify its center position as the x and y coordinates. To do that we need to write the imageMode() function and add CENTER as the parameter in the function.

To display the image on the canvas, we will use the image() function. This function will take the image’s x, y position, and width and height of the image as arguments. In the draw() function,

s create the Sprite for the bunny. Fruit, ground, and rope are physics bodies, which are using the physics library matter.js. For the bunny, we will not create a physics body, because we only need it to play its animation and detect the collision with fruit.

Now we need to create a function to drop the melon and break the rope.

We have two things to perform:

● Cut the rope from the top point where it is connected.

● Cut the fruit from the rope. To cut the rope from the top, we are going to call a function from the rope class, which is called break(). It will release the rope and remove it from the scene. Next, to detach the fruit from the rope, we need to delete the constraint between the rope and fruit by removing that constraint from the world.

We will create this function in the Link class so that we can call it using the link object. In this, we will use the **World.remove()** function, this will remove the constraint from the world. To call the **detach() & break()** function we will create a new function in the sketch.js file named drop() which we will call(execute) by a button press. We will also make the fruit constraint as null. So that it does not affect the fruit.

Now we will define the **drop**() function, where we're going to break the rope using the rope.break() function, and we remove the fruit constraint by using the fruit\_con.detach() function and making fruit\_con as null.

We have created our function, now let’s quickly create a button and add this function with that button, so that we can drop our fruit. In p5.js we have in-built functions to create buttons on the screen, and we have already done that exercise in multiple classes before as well. Where we used the **createButton**() function. But that creates a simple button on the canvas, and it does not look very visually appealing. So, now we will use a different function called **createImg**(). This takes an image as a parameter and makes that image work as a button.

define a variable as a var button. Then in the setup() function create the image button using the createImg() function and pass the image we want to display on this button. We also need to specify the size and position of the button. Then finally, we will add the function with the help of the **mouseClicked**() function. In the mouseClicked() function, we are passing the function which we want to execute when we click the button.