<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="urf-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title></title>

<link rel="stylesheet" href="./styles.css"/>

**Step 4**

Use the universal selector to add box-sizing: border-box; to your CSS. This ensures elements include padding and border in their specified width and height.

\*{

  box-sizing:border-box;

# Step 5

Give your body element a background-color of #c9d2fc.

# Step 6

Back in your HTML, create a main element. Inside that main element, add a div element with the class cat-head.

<main>

    <div class="cat-head">

        </div>

    </main>

# Step 7

Using a class selector, give the .cat-head element a width of 205px and a height of 180px. Also, give it a border of 1px solid #000 and a border-radius of 46%.

.cat-head{

  width:205px;

  height:180px;

  border:1px solid #000;

  border-radius:46%;

}

# Step 8

To see the cat-head element, give it a linear gradient background with #5e5e5e at 85% and #45454f at 100%.

You might not notice the difference between these two colors, but they are there.

background:linear-gradient(#5e5e5e 85%, #45454f 100%);

}

body{

  background-color:#c9d2fc;

}

# Step 9

CSS positioning lets you set how you want an element to be positioned in the browser. It has a position property you can set to static, absolute, relative, sticky or fixed.

Once you set the position property of the element, you can move the element around by setting a pixel or a percentage value for one or more of the top, right, left, or bottom properties.

static is the default positioning for all elements. If you assign it to an element, you won't be able to move it around with top, right, left, or bottom.

Give .cat-head a position property of static, then set the top and left properties to 100px each.

position:static;

top:100px;

left:100px;

# Step 10

You could see that nothing happens. The .cat-head element did not move despite setting a top and left of 100px each. But that's not the case with relative positioning.

When you use the relative value, the element is still positioned according to the normal flow of the document, but the top, left, bottom, and right values become active.

Instead of static, give you .cat-head a position of relative, and leave both top and left properties as they are.

 position:relative;

  top: 100px;

  left: 100px;

# Step 11

The next position property is absolute. When you use the absolute value for your position property, the element is taken out of the normal flow of the document, and then its position is determined by the top, right, bottom, and left properties.

Set the position property of your .cat-head element to absolute, then set top and left properties to any pixel value.

# Step 12

fixed is a position property value that lets you make an element fixed to the page no matter where the user scrolls to on the page.

You'll have to do some more markups to see how fixed positioning works. In your HTML, create a div element with the class box.

<div class="box"></div>

**Step 13**

Use a class selector to give your .box element a width of 200px, a height of 600px, and a background color of #000. Also, give it a position of absolute, a top of 800px and a left of 650px.

.box{

  width:200px;

  height:600px;

  background-color:#000;

  position:absolute;

  top:800px;

  left:650px;

}

 position:fixed;

  top: 100px;

  left: 100px;

# Step 15

The last position property value is sticky. sticky positioning is a hybrid of relative and fixed positioning. It allows an element to **stick** to a specific position within its containing element or viewport, based on the scroll position.

Change the value of the position property of .cat-head to sticky, set top to 0, then remove left and its value.

**Note**: To see how sticky works, you have to place a couple of texts before and after your .cat-head div element. If you scroll up after that, you'll see that the .cat-head gets stuck to the top and remains

 position:sticky;

  top: 0;

# Step 16

You should now center the cat head.

Give the .cat-head element a position property set to absolute. Set a value of 0 for the right, left, top, bottom properties, then set its margin property on all sides to auto. That's one way to center an element vertically and horizontally using CSS positioning.

 position:absolute;

 right: 0;

 left:0;

 top:0;

 bottom:0;

 margin:auto;

here it will both vertically and horizontally align items

# Step 17

Remove the div element with class box because you don't need it anymore.

# Step 20

Inside your .cat-ears element, create two div elements with the classes cat-left-ear and cat-right-ear respectively.

<div class="cat-ears">

          div class="cat-left-ears"></div>

          <div class="cat-right-ears"></div>

        </div>

# Step 21

Inside your .cat-left-ear element, create another div element with the class cat-left-inner-ear.

  <div class="cat-left-ear">

            <div class="cat-left-inner-ear"></div>

          </div>

# Step 22

Inside your .cat-right-ear element, create another div element with the class cat-right-inner-ear.

<div class="cat-right-ear">

            <div class="cat-right-inner-ear"></div>

          </div>

# Step 23

You are going to make each ear look like a triangle.

Using a class selector, give the .cat-left-ear element a left and right border of 35px solid transparent each. Also, set the bottom border to 70px solid #5e5e5e.

  border-left: 35px solid transparent;

  border-right: 35px solid transparent;

  border-bottom: 70px solid #5e5e5e;

# Step 24

Move the left ear into position by setting a position of absolute, a top of -26px, and a left of -31px.

position:absolute;

top:-26px;

left:-31px;

# Step 25

Those edges are too sharp for an ear. So, set the border-top-left-radius to 90px and the border-top-right-radius to 10px.

  border-top-left-radius:90px;

  border-top-right-radius:10px;

# Step 26

To position the left ear properly, you can use CSS transform to rotate it in a certain degree.

The transform property allows you to modify the shape, position, and size of an element without changing the layout or affecting the surrounding elements. It has functions such as translate(), rotate(), scale(), skew(), and matrix().

Set the transform property to rotate(-45deg) and see what happens.

# Step 27

Now you can work on the right ear of the cat. You have the HTML for it already.

Using a class selector, give the .cat-right-ear element a left and right border of 35px solid transparent each. Also, set the bottom border to 70px solid #5e5e5e.

# Step 26 Passed

To position the left ear properly, you can use CSS transform to rotate it in a certain degree.

The transform property allows you to modify the shape, position, and size of an element without changing the layout or affecting the surrounding elements. It has functions such as translate(), rotate(), scale(), skew(), and matrix().

Set the transform property to rotate(-45deg) and see what happens.

transform:rotate(-45deg);

# Step 27

Now you can work on the right ear of the cat. You have the HTML for it already.

Using a class selector, give the .cat-right-ear element a left and right border of 35px solid transparent each. Also, set the bottom border to 70px solid #5e5e5e.

.cat-right-ear {

  position: absolute;

  top: -26px;

  left: 163px;

  transform: rotate(45deg);

  border-left: 35px solid transparent;

  border-right: 35px solid transparent;

  border-bottom: 70px solid #5e5e5e;

}

# Step 30

Remove the sharp border of the right ear by setting the border-top-left-radius to 90px and the border-top-right-radius to 10px.

# Step 31

The ears should always be placed above the part of the head it overlaps. You can do this with the z-index property.

z-index is a property you can use to define the order of overlapping HTML elements. Any element with a higher z-index will always be positioned over an element with a lower z-index.

To see z-index in action, set the z-indexproperty of the left ear to -1.

 z-index:-1;

# Step 32

That's not the behavior you want. You should make the ears display over the head so the borders that overlap with them don't show.

Instead of -1, set the z-index property of the left ear to 1.

 z-index: 1;

# Step 34

Most cats have different colors in their ear and the inner part of the same ear. You can implement the same too. That's why you defined a div element for both right and left inner ears a while ago.

Using a class selector, give your .cat-left-inner-ear element a left and right border of 20px solid transparent each. Also give it a bottom border of 40px solid #3b3b4f.

.cat-left-inner-ear{

  border:20px solid transparent;

  border-bottom:40px solid #3b3b4f;

}

# Step 35

Move the inner ear into position with a position property set to absolute, a top of 22px, and a left of -20px.

  position:absolute;

  top:22px;

  left:-20px;

# Step 36

To remove all the pointed edges of the ear, set a bottom-right and bottom-left border radius of 40% each, a top-left border radius of 90px, and a top-right border radius of 10px.

border-bottom-right-radius:40%;

  border-bottom-left-radius:40%;

  border-top-left-radius:90px;

  border-top-right-radius:10px;

# Step 37

It's time to work on the right inner ear. Using a class selector, give your .cat-right-inner-ear element a left and right border of 20px solid transparent. Also, give it a bottom border of 40px solid #3b3b4f.

.cat-right-inner-ear{

    position: absolute;

  top: 22px;

  left: -20px;

  border-top-left-radius: 90px;

  border-top-right-radius: 10px;

  border-bottom-right-radius: 40%;

  border-bottom-left-radius: 40%;

  border-left: 20px solid transparent;

  border-right: 20px solid transparent;

  border-bottom: 40px solid #3b3b4f;

}

# Step 40

You will now start working on the cat's eyes. Like the ears, the eyes will have inner eyes.

Create a div element with the class cat-eyes. Inside the cat-eyes element, create two div elements with the class cat-left-eye and cat-right-eye respectively.

 <div class="cat-eyes">

        <div class="cat-left-eye"></div>

 <div class="cat-right-eye"></div>

        </div>

# Step 41

Inside the .cat-left-eye element, create another div element with the class cat-left-inner-eye.

<div class="cat-left-eye">

            <div class="cat-left-inner-eye"></div>

          </div>

# Step 42

Inside the .cat-right-eye element, create another div element with the class cat-right-inner-eye.

<div class="cat-right-eye">

            <div class="cat-right-inner-eye">

          </div>

          </div>

# Step 43

Using a class selector, give your .cat-left-eye element a width of 30px and a height of 40px. Also, give it a background-color of #000.

.cat-left-eye{

  width:30px;

  height:40px;

  background-color:#000;

}

# Step 44

Move the left eye into position with a position property of absolute a top of 54px, and a left of 39px.

position:absolute;

  top:54px;

  left:39px;

# Step 45

To make the left eye look like an eye, give it a border radius of 60%. Also, using the transform property, rotate it at 25 degrees.

 border-radius:60%;

  transform:rotate(25deg);

# Step 46

Now you will work on the right eye by using the same approach.

Using a class selector, give your .cat-right-eye element a width of 30px and a height of 40px. Also, give it a background color of #000.

.cat-right-eye{

  width:30px;

  height:40px;

  background-color:#000;

}

# Step 47

Move the right eye into position with a position property of absolute a top of 54px, and a left of 134px.

 position:absolute;

  top:54px;

  left:134px;

# Step 48

To make the right eye look like an eye, give it a border radius of 60%. Also, using the transform property, rotate it at -25 degrees.

 border-radius:60%;

  transform:rotate(-25deg);

# Step 49

Those look like eyes, but you can still make them better. That's why you created two inner eyes div elements.

Using a class selector, give your .cat-left-inner-eye element a width of 10px and a height of 20px. Also, give it a background color of #fff.

.cat-left-inner-eye{

  width:10px;

  height:20px;

  background-color:#fff;

}

# Step 50

Move the left inner eye into position with a position property of absolute, a top of 8px, and a left of 2px. Also, give it a border radius of 60% and rotate it at 10 degrees.

position:absolute;

  top:8px;

  left:2px;

  border-radius:60%;

  transform:rotate(10deg);

# Step 51

Using a class selector, give your .cat-right-inner-eye element a width of 10px and a height of 20px. Also, give it a background color of #fff.

.cat-right-inner-eye {

  position: absolute;

  top: 8px;

  left: 2px;

  width: 10px;

  height: 20px;

  transform: rotate(10deg);

  background-color: #fff;

  border-radius: 60%;

}

# Step 53

It's time to work on the nose. In your HTML, create a div element with the class cat-nose.

 <div class="cat-nose"></div>

# Step 54

Using a class selector, give your .cat-nose element a left and right border of 15px solid transparent each. Also give it a bottom border of 20px solid #442c2c.

.cat-nose{

  border-left:15px solid transparent;

  border-right:15px solid transparent;

  border-bottom:20px solid #442c2c;

}

# Step 55

Move the nose into position with a position property of absolute, a top of 108px, and a left of 85px.

position:absolute;

  top:108px;

  left:85px;

# Step 56

Remove the sharp edges of the nose with border radius of 50% each on the top-left, bottom-right, and bottom-left corners. Also, rotate it at 180 degrees.

 border-bottom-left-radius:50%;

   border-bottom-right-radius:50%;

    border-top-left-radius:50%;

     border-top-right-radius:50%;

     transform:rotate(180deg);

# Step 57

Now you will start working on the mouth. There will be a right line and left line for the mouth.

Create a div element with the class cat-mouth.

  <div class="cat-mouth"></div>

<div class="cat-mouth">

          <div class="cat-mouth-line-left">

        </div>

        <div class="cat-mouth-line-right">

        </div>

        </div>

# Step 59

Using a descendant selector, select the two div elements inside the div with class cat-mouth. Give it a width of 30px, a height of 50px, and a border of 2px solid #000.

.cat-mouth div{

width:30px;

height:50px;

border:2px solid #000;

}

# Step 60

You aborder-color:black transparent transparent transparent;

  border-radius:190%/190% 150px 0 0;

re going to make the two mouth lines into an elliptical shape. So, give the .cat-mouth div selector a border color of black transparent transparent transparent and a border radius of 190%/190px 150px 0 0.

cat-mouth div {

  width: 30px;

  height: 50px;

  border: 2px solid #000;

  border-radius: 190%/190px 150px 0 0;

  border-color: black transparent transparent transparent;

}

.cat-mouth-line-left{

    position:absolute;

    top:88px;

    left:74px;

  }

transform:rotate(170deg)

cat-mouth-line-right {

  position: absolute;

  top: 88px;

  left: 91px;

  transform: rotate(165deg);

}

# Step 66

Inside the .cat-whiskers element, create two div elements with the class cat-whiskers-left and cat-whiskers-right.

   <div class="cat-whiskers"></div>

<div class="cat-whiskers">

           <div class="cat-whiskers-left">

        </div>

         <div class="cat-whiskers-right">

        </div>

        </div>

<div class="cat-whiskers-left">

             <div class="cat-whisker-left-top">

          </div>

           <div class="cat-whisker-left-middle">

          </div>

           <div class="cat-whisker-left-bottom">

          </div>

          </div>

# Step 68

Inside the .cat-whiskers-right element, create 3 div elements with the class cat-whisker-right-top, cat-whisker-right-middle, and cat-whisker-right-bottom.

 <div class="cat-whiskers-right">

            <div class="cat-whisker-right-top">

          </div>

          <div class="cat-whisker-right-middle">

          </div>

          <div class="cat-whisker-right-bottom">

          </div>

          </div>

# Step 69

Use a descendant selector to target the three div elements inside your .cat-whiskers-left element. Give it a width of 40px, a height of 1px, and a background-color of #000.

.cat-whiskers-left div{

  width:40px;

  height:1px;

  background-color:#000;

}

# Step 70

As you did in the previous step, use a descendant selector to target the three div elements inside your .cat-whiskers-right element. Give it a width of 40px, a height of 1px, and a background-color of #000.

.cat-whiskers-right div{

  width:40px;

  height:1px;

  background-color:#000;

}

.cat-whisker-left-top{

  position:absolute;

  top:120px;

  left:52px;

}

 transform:rotate(10deg);

 .cat-whisker-left-middle {

  position: absolute;

  top: 127px;

  left: 52px;

  transform: rotate(10deg);

}

transform:rotate(3deg);

# Step 75

Using a class selector, move the .cat-whisker-left-bottom into position with a position of absolute, a top of 134px, and a left of 52px.

 .cat-whisker-left-bottom {

  position: absolute;

  top: 134px;

  left: 52px;

  transform: rotate(3deg);

}

transform:rotate(-3deg);

# Step 77

Now you will work on moving the right whiskers into place. Use class selector to target the .cat-whisker-right-top element and give it a position of absolute, a top of 120px, and a left of 109px.

.cat-whisker-right-top {

  position: absolute;

  top: 120px;

  left: 109px;

  transform: rotate(10deg);

}

transform:rotate(-10deg);

transform:rotate(-3deg);