Transforming and transitioning elements

Open Lab

Instructions

Introduction

In this lab you have been provided with an HTML code. You will practice using the transform and transition properties in CSS by applying them to specific selectors inside your code.

Goal

- To animate the letter boxes created on the webpage using the transform and transition properties inside CSS to add interactivity to the page

Objectives

- Add a set of rules to a specific selector present inside your CSS code for the corresponding HTML code
- Add transform and transition properties for the same element in order to add interactivity

Learner Instructions

Initial code for the HTML is already provided. Additionally, rules have already been added for the ‘container’ class and the ‘letters’ class along with a few more properties defined using the Universal selector. You have to add the rules for other selectors as per the instructions below:

Add the following rules for the 'p' element present inside the class 'letters' using descendant selector:

Step 1: Align the text to center

```
text-align: center;
```

Step 2: Change the background color to RGB value of 250, 150, 100 respectively

```
background-color: rgb(250, 150, 100);
```

Step 3: Assign the width to be 70 pixels

```
width: 70px;
```

Step 4: The right side of the margin should be assigned a value of 1.5 pixels

```
margin-right: 1.5px;
```

Step 5: Change the radius of the border to 15 percent

```
border-radius: 15%;
```

Step 6: Now assign the color of the border an RGB value of 250, 125 and 75 respectively

```
border-color: rgb(250, 125, 75);
```

```
border-color: rgb(250, 125, 79);
```

Step 7: Finally change the border style to be solid

```
border-style: solid;
```

Create new rules for the 'even' nth child of the 'p' element that follows the class 'letters'.

Make use of the descendant selector such that, once you 'hover' over the class 'letters', for the 'nth-child' of the 'p' element inside it, the rules below must be assigned:
Step 8: Add a transformation where you will rotate the 'p' element tags on Y-axis by 360 degrees.

```
transform: rotateY(360deg);
```

Step 9: You must also show transition for this over a time span of 0.5 seconds.

```
transition: 0.5s;
```

Hint: The rules for 'even' valued children of a given selector can be specified by using a syntax such as: selector:n-th-child(even)

Similar to the rules above, you must now create rules for the 'odd' nth-child of the 'p' element that follows the class 'letters'.

Make use of the descendant selector such that, once you 'hover' over the class 'letters', for the odd 'nth-child' of the 'p' element inside it, the rules below must be assigned:
Hint: The rules for 'odd' valued children of a given selector can be specified by using a syntax such as: selector:n-th-child(odd)

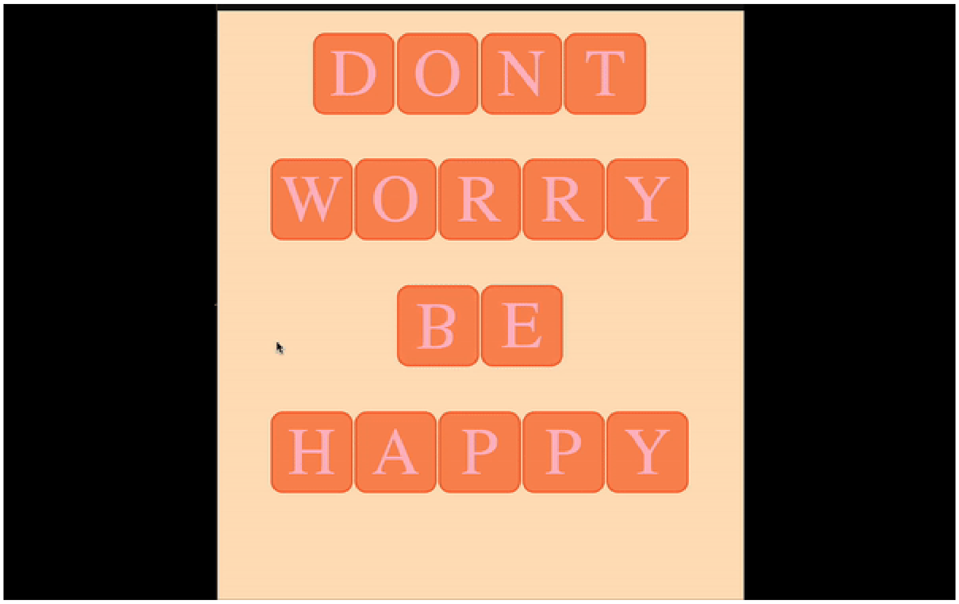
Step 10: Add a transformation where you will rotate the 'p' element tags on X-axis by 360 degrees.

```
transform: rotateX(360deg);
```

Step 11: You must also show transition for this over a time span of 1.5 seconds.

```
transition: 1.5s;
```

Save the CSS file and open Live Preview. The result should be similar to the following animation.


D O N T
W O R R Y
B E
H A P P Y