COMP4021
Internet Computing

Controlling CSS Animations

Gibson Lam

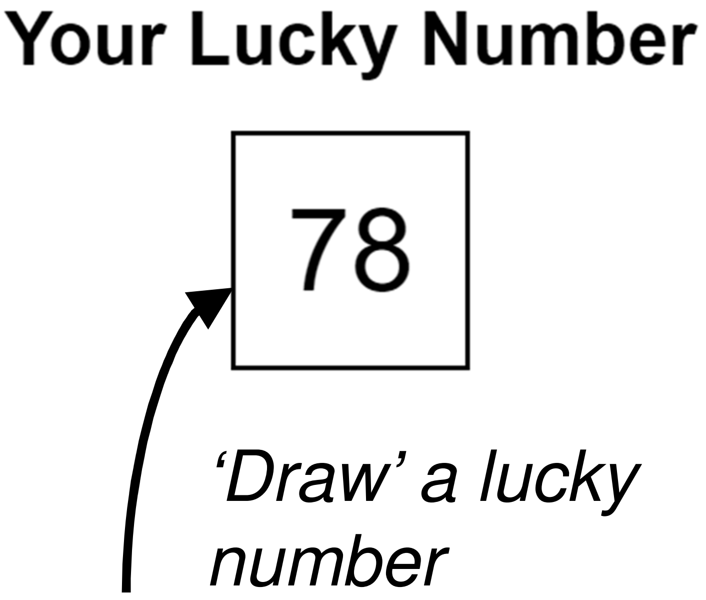
Controlling CSS Animations

- The CSS animations that you have seen in the previous discussions always start after the web page finishes loading
- It would be more useful if you can write JavaScript to control them
- In this presentation, we will look at using the animation properties and JavaScript events to do that

Your Lucky Number

- Here is a web page giving you a lucky number of the day

```
<!DOCTYPE html>
<html>
<head>
  <title>Lucky Number</title>
  <script src="https://code.jquery.com/jquery-3.6.0.min.js">
  </script>
  <style>
  body { ...CSS not shown... }
  .number { ...CSS not shown... }
  </style>
</head>
<body>
  <h1>Your Lucky Number</h1>
  <div class="number">?</div>
  <script>
  $(".number").text(parseInt(Math.random() * 100) + 1);
  </script>
</body>
</html>
```



Your Lucky Number

78

'Draw' a lucky number

Improving the Page

- The web page shown in the previous slide works fine but it is not very interesting
- We want to gradually improve the page by using CSS animations
- First, we want to apply a spinning animation to the number as if the page is drawing a new number every time
- This can be achieved by using appropriate CSS animation rules, see next slide

Spinning the Number

- We can spin the number by adding the following CSS rules to the page:

Rotate around 360 degrees

```
@keyframes spin {
    from { transform: rotate(0deg); }
    to { transform: rotate(360deg); }
}
```

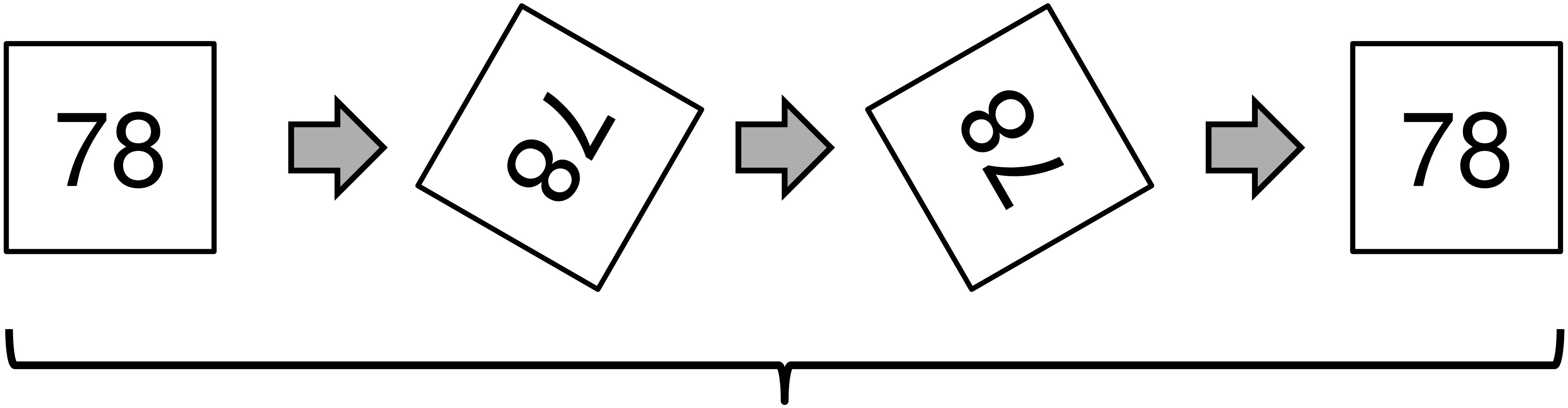
Apply the rotation to the number 5 times

```
.number {
    animation: spin 0.2s;
    animation-timing-function: linear;
    animation-iteration-count: 5;
}
```

This specifies the number of times the animation is used

The Spinning Animation

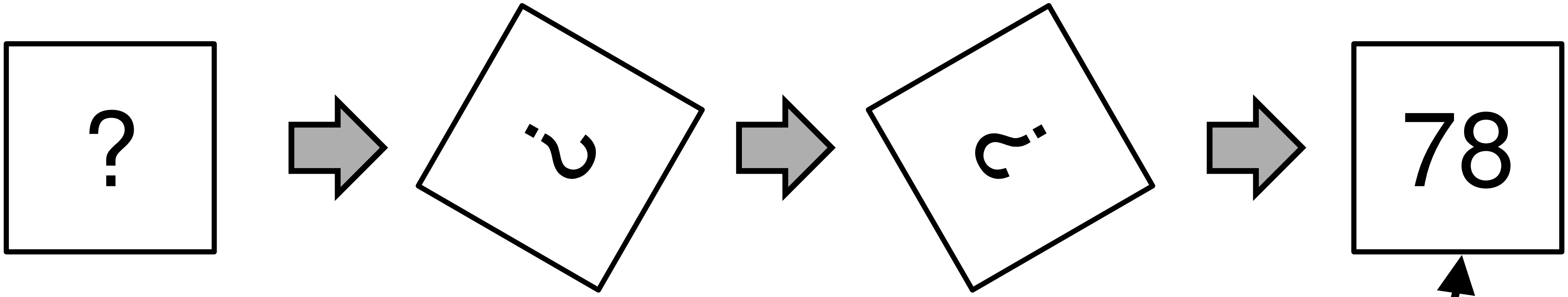
- After applying the animation, the number rotates quickly when the page is loaded:



This rotation repeats five times

Hiding the Number

- While the number is spinning, the lucky number is already shown in the HTML
- It will look more fun if the number shows a question mark ‘?’ while it is spinning and only reveals the number at the end



The animation finishes and the number is shown

Using Animation Events

- To show the number only when the animation finishes, we can use one of the JavaScript events for CSS animations
  - `animationstart`
    triggers when an animation starts
  - `animationiteration`
    triggers when one iteration of an animation is completed
  - `animationend`
    triggers when an animation finishes

This is useful for what we want to do

Setting Up the Animation Event

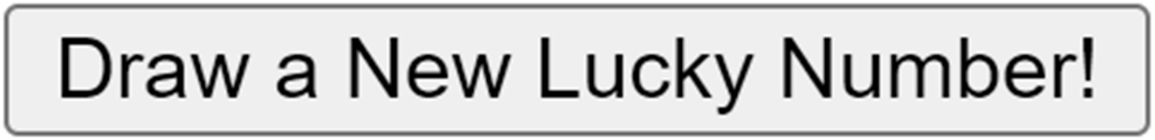
- You can use jQuery to handle an animation event just like what you have done for typical jQuery DOM events, i.e. `ready` and `click`
- Here we use jQuery to set up the event:

```
$(".number").on("animationend", function() {
  $(".number")
    .text(parseInt(Math.random()*100)+1);
});
```

'Draw' the number when the spinning animation finishes

Drawing a New Number

- The final improvement we want to make in the page is to allow you to draw the number again
- A new button is added to the page so that the number is drawn again if you click on the button



- This can be achieved by simply restarting the spinning animation

Restarting the Animation

- There are a couple of ways to restart a CSS animation
  1. You can get all animation objects on a page, find the one you want and restart it
  2. You can clear and reset the animation name of an animation to restart it
- We will use the second approach because it is simpler and shorter

Using the Animation Name

- To clear the animation of the number in the page, you can use this code:

```
$(".number").css("animation-name", "none");
```

  - You need to clear the animation only when the animation finishes, i.e. `animationend`

- Then, the animation can be started again if you reset the animation name, i.e. in the button’s click event:

```
$(".number").css("animation-name", "spin");
```