Final

1. Syntax (details): Angular uses TypeScript and splits HTML + TypeScript logic apart, React uses JavaScript and a feature called “JSX” (it combines “HTML” and JavaScript logic), Vue uses regular JavaScript and splits HTML + JavaScript logic apart. React is a library, whereas Vue is a framework*. React allows DOM manipulation, component architecture, and state management (the component-level state, not Redux). ... Some examples of such companion libraries are the Vue-router, Vuex for state management, and Vue CLI. Angular JS is a front-end framework and can be used with any backend programming language like PHP, Java etc., whereas Vue JS is strictly front-end based and uses HTML, CSS and JS separately. By default, Vue uses HTML templates, but there's an option to write in JSX. In React, on the other hand, there's solely JSX. ... React's JavaScript Expressions (JSX) combine HTML and CSS together into JavaScript
2. There are several similarities shared by React and Vue. For example, both make use of the component-based architecture and the virtual DOM, each of them uses props, and debugging is done by means of the Chrome Dev tools in the two JS frameworks. AngularJS is a structural framework for developing dynamic web apps, whereas React is a javascript library that allows you to build UI components. Angular JS is based on MVC (Model View Controller) whereas React is based on Virtual DOM. Angular is based on Typescript and React is based on Javascript.
3. Officially described as “CSS with superpowers,” SCSS (or Sass) offers a way to write styles for websites with more enhanced CSS syntax. In general, browsers do not know how to process SCSS features, such as functions, mixins, and nesting. We'll need to convert them to regular CSS files to run them in the browser.
4.
```
<table class="table">
 <thead>
  <tr>
   <th scope="col">#</th>
   <th scope="col">First</th>
   <th scope="col">Last</th>
   <th scope="col">Handle</th>
  </tr>
 </thead>

 <tbody>

  <tr>
   <th scope="row">1</th>
   <td>Mark</td>
   <td>Otto</td>
   <td>@mdo</td>
  </tr>
  <tr>
   <th scope="row">2</th>
   <td>Jacob</td>
   <td>Thornton</td>
   <td>@fat</td>
  </tr>
```

```
    <tr>
      <th scope="row">3</th>
      <td>Larry</td>
      <td>the Bird</td>
      <td>@twitter</td>
    </tr>
  </tbody>
</table>
```

5. The technical difference between a framework and library lies in a term called inversion of control. When you use a library, you are in charge of the application flow. You choose when and where to call the library. When you use a framework, the framework is in charge of the flow.
6. The HTML <audio> element is used to embed sound content in documents. It may contain one or more audio sources, represented using the src attribute or the <source> element: the browser will choose the most suitable one. It can also be the destination for streamed media, using a MediaStream .
7. <video>: The Video Embed element. The HTML Video element (<video>) embeds a media player which supports video playback into the document. You can use <video> for audio content as well, but the <audio> element may provide a more appropriate user experience.
8. <canvas> is an HTML element which can be used to draw graphics via scripting (usually JavaScript). This can, for instance, be used to draw graphs, combine photos, or create simple (and not so simple) animations. The images on this page show examples of <canvas> implementations which will be created in this tutorial.
9. The animatable properties are:
   -moz-outline-radius.
   -moz-outline-radius-bottomleft.
   -moz-outline-radius-bottomright.
   -moz-outline-radius-topleft.
   -moz-outline-radius-topright.
   -ms-grid-columns.
   -ms-grid-rows.
10.
```
<!Doctype html>
<html>
        <head>
                <title>Font Awesome Example</title>
        </head>
        <body>
                <i class="fa fa-futboll-0"></i>
                <i class=" fa fa-futboll-0" style="font-size:50px;"></i>
                <i class=" fa fa-futboll-0" style="font-size:60px; color:red;"></i>
        </body>
</html>
```