Our challenge today is to complete the ‘coming soon/ app, an utility that will first get the ten most popular movie trailers of the last year from the YouTube API, and then build functionality into our app that will allow a user to cycle forward and backward through these videos, which will display to a div.

s

1. Go to this GitHub address: <https://github.com/codeslo/intro-to-code-11> and clone the index.html, style.css and main.js files, along with the images and bootstrap folders.
2. Create a project in your work environment (VS Code, etc) with these three files and the two subfolders.
3. The YouTube API requires a key to work. You can either register as a Google developer and get your own unique key by following instructions: <http://help.dimsemenov.com/kb/wordpress-royalslider-tutorials/wp-how-to-get-youtube-api-key>
4. If you’ve never embedded a YouTube video into a web page before, or if it’s been awhile, go to YouTube, pick a video and go share => embed. Take a good long look at how that embed works. Try adding it to the page, and make sure it displays where it is supposed to, and that the video actually plays.
5. To make this project work, you will need to format a valid request for the YouTube API, and add it to the URL section of the jQuery ajax method already written into main.js. Your request must contain the following parameters:
   1. part = snippet
   2. maxResults = 1
   3. order by viewCount
   4. only allow videos published in 2016
   5. type = video
   6. category should be entertainment
      1. hint #1: categories are chosen by category id number. see the file on YouTube categories
      2. hint #2: not all entertainment videos are movie trailers!
      3. movie trailers almost always contain the words “teaser” or “trailer” in their title.
   7. Embeddable = true
   8. videoCaption = any
   9. Your API key
   10. Google provides a great tool for helping to build a valid HTTP request, which you can find here: <https://developers.google.com/apis-explorer/#p/youtube/v3/youtube.search.list?part=snippet&q=YouTube+Data+API&type=video&videoCaption=closedCaption>
6. Try running the loadVideo function. You should get back JSON data about your one, hit movie trailer. See if you can use this data to make a dynamically created video embed for your project.
7. increase the maxResults part of your HTTP request to 10. You should now get back data on ten trailers instead of one.
8. Use this data to enable cycling through your videos. What is the best way to do that? If you can figure it out and make it work, you’ve completed the challenge!