Source Version : https://goo.gl/tsLwmc

SI659 w18 Lab 2- Javascript

- Please refer to the programming lab guideline for uploading, late policy and other general rules.
- Please upload both lab2_js_.html and lab2_js_data.js to canvas by next Monday midnight.
- Even if you worked in groups, please upload individually.
- Please do not hard code anything directly as html. E.g. instead of directly write <title>Licia</title> in the <head>, please use javascript to select <head>, create <title> element, and append it to <head>. I.e. *only change and write code in the javascript section (the section within the <script></script> tag) in your code*.

Section 1: select, append, and basic elements

In this section, we will work on the html and javascript basics. Mainly, you need to **understand HTML's DOM (Document Object Model) structure, be able to select and edit DOM elements.** There are two groups of tutorials. The first group gives brief introductions while the second group gives more information about selecting and tags.

We have included tutorials for Jquery, a popular javascript library, because it offers convenient methods for selecting elements and appending new elements to the document. However, it is not required, and you can use vanilla javascript to achieve the same thing. If you decide to use Jquery, we might not have all Jquery tutorials that you need, but they are easy to find using google.

1) HTML/Javascript/Jquery Introduction:

- https://www.w3schools.com/html/html_intro.asp
- https://www.w3schools.com/js/js_intro.asp (getElementById())
- https://www.w3schools.com/js/js_htmldom.asp (Intro to DOM)
- https://www.w3schools.com/jquery/jquery_intro.asp (Jquery intro)

2）Basic elements and selecting/editing elements

- https://www.w3schools.com/js/js_htmldom_document.asp (select and modify elements)
- https://www.w3schools.com/html/html_basic.asp (<html><body><head>)
- https://www.w3schools.com/html/html_elements.asp (<h><p><img><a>)
- https://www.w3schools.com/jsref/met_document_queryselector.asp (querySelector, selecting using the css selector syntax)
    - Create new element with javascript: https://www.w3schools.com/jsref/met_document_createelement.asp

Source Version : https://goo.gl/tsLwmc

- https://www.w3schools.com/jquery/jquery_selectors.asp (Jquery Selector, you can create elements in string)

After reading (and trying out) this group of examples, please answer the following questions in your lab html file.

Question 1.1: Add a <title> element with your name.

In the answer file under “Q1.1”, write code to select the <head> element and append a <title> element with “Lab 2: your name” (for example, add an element like “<title>Lab 2: Licia</title>” to the document).

Before you complete this question, your browser toolbar would display the html filename, like this:

Figure 1

After appending the <title> element, the name on your browser tab should be updated to look like this (but with your name):

Figure 2

Read more about the HTML title tag and why it is required for every webpage: https://www.w3schools.com/tags/tag_title.asp

Question 1.2

We have provided a <h1> element in the <body> with id=“title”. You can see it in your html or by inspecting in the developer mode:

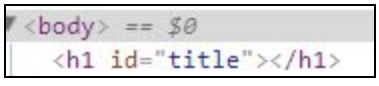


Figure 3

Modify the <h1 id=”title”></h1> element and use it to display your (and your partner's) name; e.g. display "Lab 2: Licia, section 1" within an a <h1> element.

Source Version : https://goo.gl/tsLwmc

Before editing the title, you will see nothing on the screen. But if you inspect the <h1> element, you will see an empty <h1> element in the <body> (Figure 3, above).

After adding the title, you should see something like this:

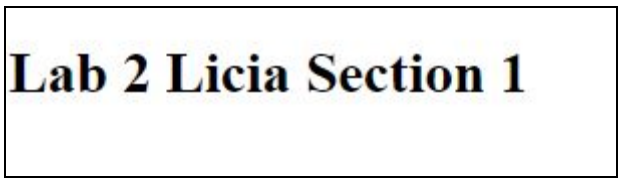


Figure 4

Section 2: display, debug, object and array.

In this section, we will look at common ways to access and display information on your webpage. Here are some relevant tutorials:

Display and modify dom elements:

- Different methods to display information: https://www.w3schools.com/js/js_output.asp
- (Js) appendChild: https://www.w3schools.com/jsref/met_node_appendchild.asp

Data: Variable, Object (aka dictionary in python) and Array:

- Variable: https://www.w3schools.com/js/js_variables.asp
- Object https://www.w3schools.com/js/js_object_definition.asp (also read property, method and constructors)
- Array: https://www.w3schools.com/js/js_arrays.asp (also read the array method tutorial)

Question 2.1: Look at the data

Take a look at `lab2_js_data.js`. It contains a smaller part of the movie dataset.

In the future, we will teach you how to load external data, but for this lab, all data is stored in an array called "mvData", and each row is stored as an object with four fields: `"Title", "Worldwide Gross", "Production Budget", "Major Genre".`

If you pass mvData to the console.log function, you should see something like figure 5.

Source Version : https://goo.gl/tsLwmc

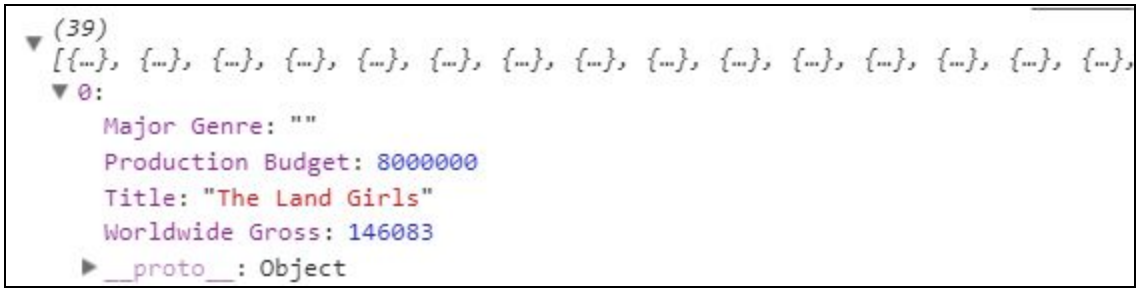


Figure 5

You don't have to write anything for this question, just make sure that you understand the structure of this dataset before you move on.

Question 2.2: Logging to console

Display the title of the **first** movie in your console (console.log). (Figure 6)

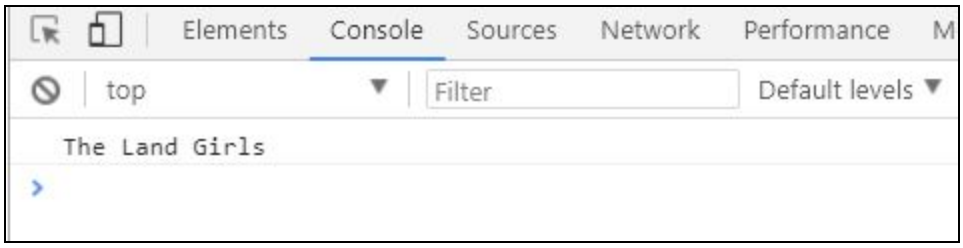


Figure 6

Question 2.3: Display second movie

Display the title of the second movie in your web page as a <p> element in the <body> of your document.

After adding the <p> element, your page should look like Figure 7.

Figure 7

Source Version : https://goo.gl/tsLwmc

Question 2.4: Display third movie

Create a div element with id=”thirdMovie” to display the information about the third movie. You will display:

- Its title as a <h4> element,
- Its worldwide gross, production budget and major genre as individual <p> elements

When you finish, you will have an element that looks like Figure 8 in the debug view:

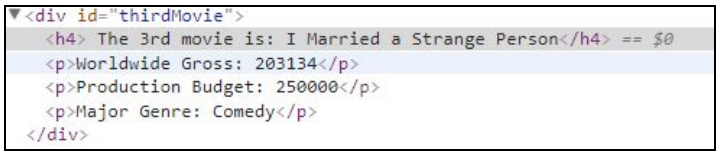


Figure 8

and your webpage will look like Figure 9:

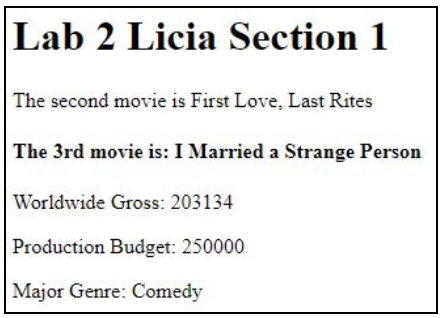


Figure 9

Section 3 Loops, Data manipulation, sort, and list.

In this section, we will write some for loops to manipulate our data. We will also take a look at custom sorting functions and present sorted results in a list.

Tutorials:

- https://www.w3schools.com/js/js_loop_for.asp (for loops)
    - If you have never written a for loop before, here’s a cute and gentle intro to programming and for loop from Google Doodle : https://www.google.com/doodles/celebrating-50-years-of-kids-coding?hl=en-GB

Source Version : https://goo.gl/tsLwmc

- Array and object sort, please pay attention to the object sort (comparison functions): https://www.w3schools.com/js/js_array_sort.asp
- Intro to list (ordered and unordered): https://www.w3schools.com/html/html_lists.asp

Question 3.1: add revenue to dataset

Add a new field to the movie dataset called revenue, which is equal to the difference between worldwide gross and production budget. To add the revenue, you can loop over the elements of your dataset and create a new field for each element.

After adding the field, console.log your dataset. You will see something like Figure 10 (you may have to hit the arrows to expand the view of the object).

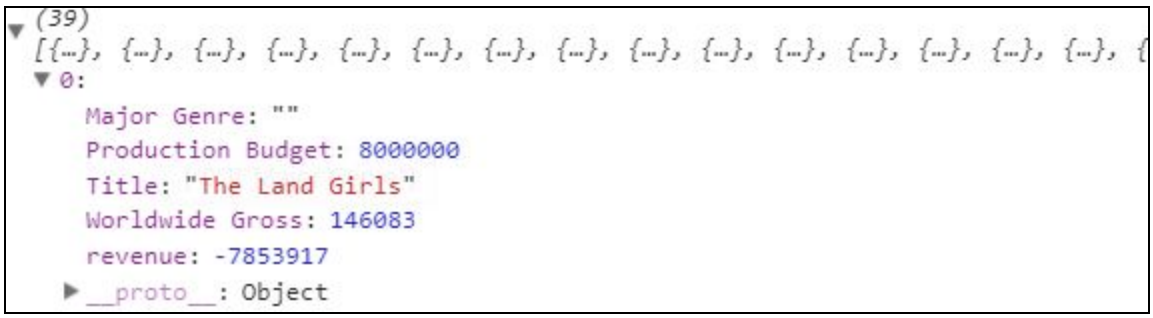


Figure 10

Question 3.2 display the revenue

Display the revenue of your last movie as a <p> element (Figure 11).

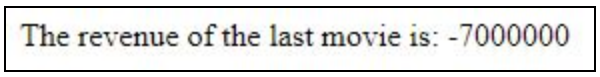


Figure 11

Do not hard code the index of the last element. Even though we know that there are 39 movies in the dataset, use mvData.length to find the last element to make your code robust. Read more about array indices here: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array

Source Version : https://goo.gl/tsLwmc

Question 3.3 Sort the dataset

Sort the movie dataset by revenue (either ascending or descending order is fine)

Hint: you can do this using multiple methods. We recommend you use the object sorting method explained in here: https://www.w3schools.com/js/js_array_sort.asp
Pay attention to the car sorting example.

If you sort it in ascending order, you can log your data to the console, which should look like Figure 12:

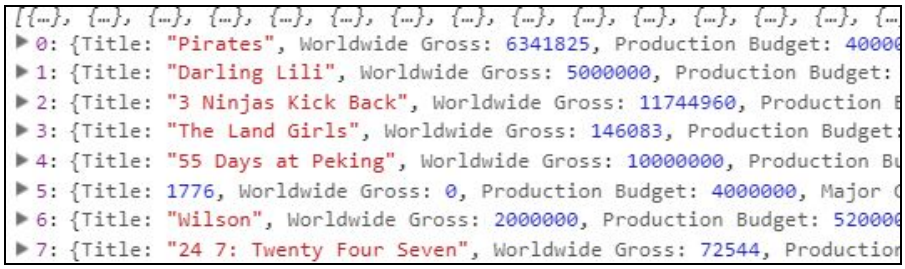


Figure 12

Question 3.4 Display top 10 movies

Display movies with the top 10 revenue in an ordered list. You can append a header (e.g. <h3>) element as title for the list.

Hint: read about list elements: https://www.w3schools.com/html/html_lists.asp

When you are done, you should have an <ol> element that looks like Figure 13:

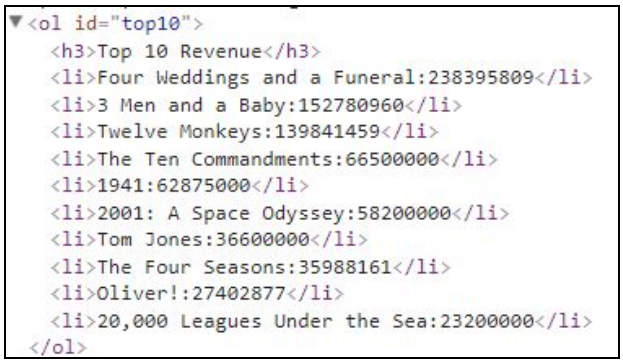


Figure 13

and your web page should display the list similar to Figure 14:

Source Version : https://goo.gl/tsLwmc

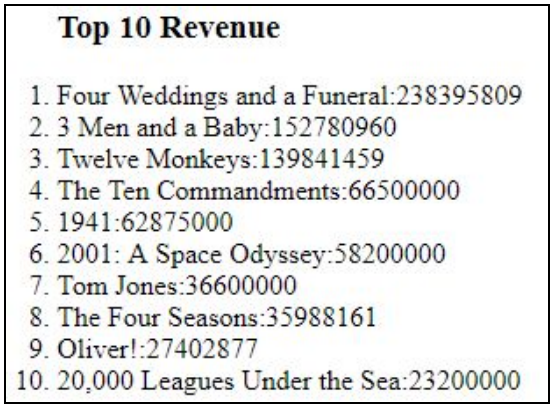


Figure 14

Question 3.5 Count the number of movies within each genre

We want to count the number of movies in each genre so that we will be able to display this information in unordered list (which we will do in the question after this one). There are multiple ways to get the count using arrays, objects, and perhaps other methods. You can use the method of your choice. **For movies without a genre, you can put them in the "null" genre.**

One approach that we recommend is to go through the following steps:

1. Create a new object and assign it to a variable (e.g. genreCt). You will use genre names as keys in this object, and the values will be the number of movies in the corresponding genre.
2. Write a for loop to iterate through all movies.
3. For each movie, increment the value in genreCt corresponding to that movie's genre.
   Note: you will need to handle two special cases:
   a. Where you encounter a movie in a genre that has not been seen before. In that case, you will need to add its key to genreCt.
   b. Where you encounter a movie without a genre (its genre is an empty string), you should increment a "null" genre counter (e.g. genreCt.null).

If you are using the method that we suggested, your new object will look similar to Figure 15:

Figure 15

Source Version : https://goo.gl/tsLwmc

Question 3.6: Display the genres and their counts

Create a <ul> (“unordered list”) element containing the genres and their counts. The debug view should look like Figure 16, and your webpage should look like Figure 17.

```
<ul id="genreCt">
  <h3>Genre Count</h3>
  <li>null: 15</li>
  <li>Action: 2</li>
  <li>Drama: 8</li>
  <li>Comedy: 8</li>
  <li>Thriller/Suspense: 1</li>
  <li>Musical: 2</li>
  <li>Adventure: 2</li>
  <li>Romantic Comedy: 1</li>
</ul>
```

Figure 16

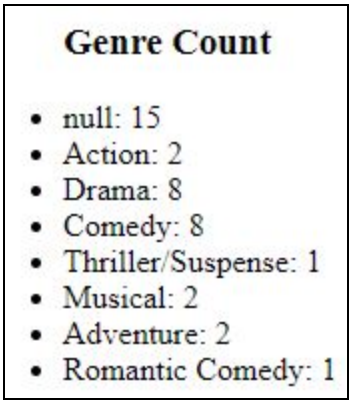


Figure 17

Section 4: table, editing table, and button

In this section, we are going to create a <table> element. We are going to write functions to modify this table, and add a button to call this function.
Tutorials:

- (Table) https://www.w3schools.com/html/html_tables.asp
- (Function)https://www.w3schools.com/js/js_functions.asp
  - Definition https://www.w3schools.com/js/js_function_definition.asp
  - Parameters https://www.w3schools.com/js/js_function_parameters.asp
  - Invocation https://www.w3schools.com/js/js_function_invocation.asp
  - Call https://www.w3schools.com/js/js_function_call.asp
- (Event) https://www.w3schools.com/js/js_events.asp
- (Button)https://www.w3schools.com/tags/tag_button.asp

Question 4.1 Create a table to display the first movie

Create a <table> element and append first movie’s title, worldwide gross, production budget, revenue and genre. Your table should look like Figure 18:

Source Version : https://goo.gl/tsLwmc

Movie	Worldwide Gross	Production Budget	Revenue	Genre
Pirates	6341825	40000000	-33658175	

Figure 18

Depending on how your sorted your dataset, the particular movie displayed here might be different. The debug view of your table element should look similar to Figure 19:

```
<table id="mvTable">
  <tbody>
    <tr>
      <th>Movie</th>
      <th>Worldwide Gross</th>
      <th>Production Budget</th>
      <th>Revenue</th>
      <th>Genre</th>
    </tr>
  </tbody>
  <tbody>
    <tr>
      <td id="mvTitle">Pirates</td>
      <td id="wwGross">6341825</td>
      <td id="pb">40000000</td>
      <td id="rv">-33658175</td>
      <td id="ge"></td>
    </tr>
  </tbody>
</table>
```

Figure 19

Hint: we are going to edit content of this table soon, so you might want to give each cell an id so it is easier to access them later.

Question 4.2 Create a random number generator

Instead of always displaying the first movie, we want our table to display a random movie from our dataset. Therefore, we need to first create a random number generator that gives us a random index ranging from 0 to mvData.length - 1.

Take a look at https://www.w3schools.com/js/js_random.asp. You can copy one of the getRndInteger(min, max) functions into your code (make sure you understand if your function **excludes** the max or **includes** the max).

Source Version : https://goo.gl/tsLwmc

Question 4.3 Write a function to display a random movie

Write a function that updates the table to display a random movie. When you call this function, your table should change.

Call your function in the debug console multiple times (e.g. I called my function placeRandomMovie):

```
> placeRandomMovie()
```

Verifying that each time you do this, your table updates to display a random movie:

Movie	Worldwide Gross	Production Budget	Revenue	Genre
The Four Seasons	42488161	6500000	35988161	Comedy

Figure 20

Question 4.4 Add a button to display a random movie.

Append a button to your page. Make the button to call the function you just wrote. When you click the button, your table should display a random movie.

(Hint: change the “onclick” attribute of the button to call the placeRandomMovie function.)

Movie	Worldwide Gross	Production Budget	Revenue	Genre
Pirates	6341825	40000000	-33658175	

Random Movie

Figure 21

Section 5 Selection (dropdown list), accessing form value, and sort by input

See these tutorials:

- (input methods in general) https://www.w3schools.com/html/html_form_elements.asp
- (select/drop down list) https://www.w3schools.com/tags/tag_select.asp

Source Version : https://goo.gl/tsLwmc

    - Example of using javascript to access the value created in dropdown lists: https://stackoverflow.com/questions/1085801/get-selected-value-in-dropdown-list-using-javascript
- Change event: https://developer.mozilla.org/en-US/docs/Web/Events/change

Question 5.1 Create data field dropdown

Create a <select> element that includes “worldwide gross”, “production budget”, “revenue”, and “major genre” as options.

Hint: you can place the <select> element in a new so that it is not directly next to your button.

After finishing the step, you should have a drop down list like shown in Figure 22, and the <select> element in the debug view should look like Figure 23:

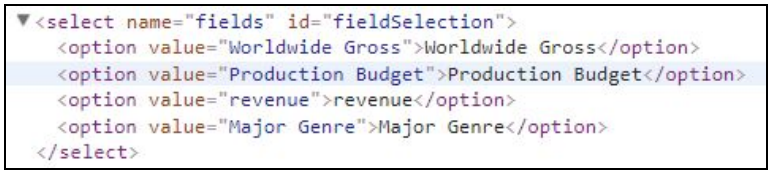


Figure 22 *Figure 23*

Question 5.2 Create a table to display the selected field

Create a table with 2 columns and 6 rows. The first row will be the column headers. The first column will display the titles of the first 5 movies, and the second column will display the field from your data that is selected by the drop down list. Initially, set this to be the first field in the dropdown (“worldwide gross”).

E.g. The table should initially look like Figure 23, and the debug view should look like Figure 24:

Source Version : https://goo.gl/tsLwmc

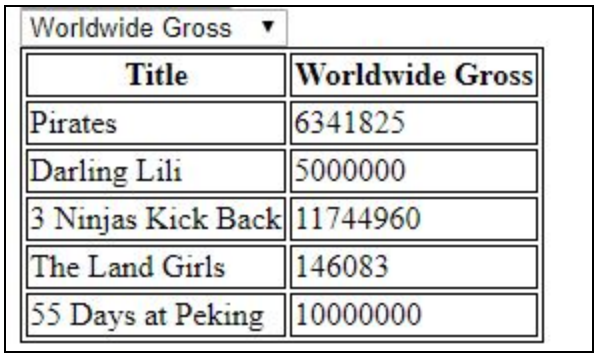


Figure 23

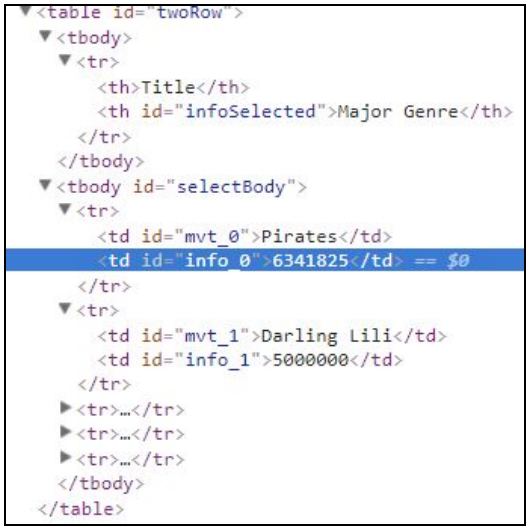


Figure 24

Notice that if you change the selection (e.g. from worldwide gross to revenue), nothing should happen yet., That's because you haven't written the function to update the table whenever your selection changes. Time for the next question!

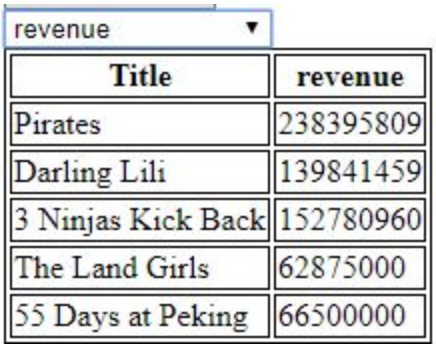
Question 5.3 Update the table when the selection changes.

To update the table when the value of the <select> element changes, you need to do two steps:

1) Write a function to update the table when a new field is selected.
    a) E.g. when your selection changes from Worldwide gross to Revenue, you need to first change the title of the second column. Then, for every movie listed in your table, you want to update the value of the second column to display revenue.
    b) In order to do that, you need to first get the value selected (see the second tutorial link, or google it if you want to use jquery)
    c) It might be useful to add a custom id for each cell so that you can access them easily. e.g. In Figure 24, you can see I named the cells I am changing as "info_0", "info_1".... So that I can find them easily.
    d) Once you've written this function, test that it works in the console before doing the next step.
2) After writing the function, you need to make sure it is called whenever the selection value is changed. This is done through a change event. You can update the <select>'s "onchange" element to point to the function you just wrote.

When you are done, test whether changing options will update your table (Figure 25)

Source Version : https://goo.gl/tsLwmc



Title	revenue
Pirates	238395809
Darling Lili	139841459
3 Ninjas Kick Back	152780960
The Land Girls	62875000
55 Days at Peking	66500000

Figure 25

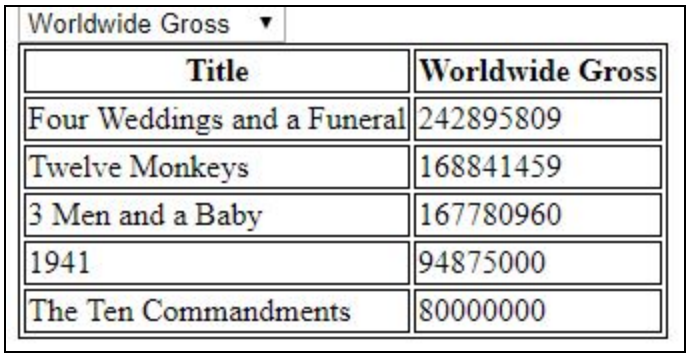
Question 5.4 (BONUS, optional) Sorted Table

It's great that we can display the first five movies with the field that we care about. However, what we really want is a sorted table instead of the first five movies in our dataset. Edit the update function so that when a new field is selected, the output is also sorted by that field **in descending order**.

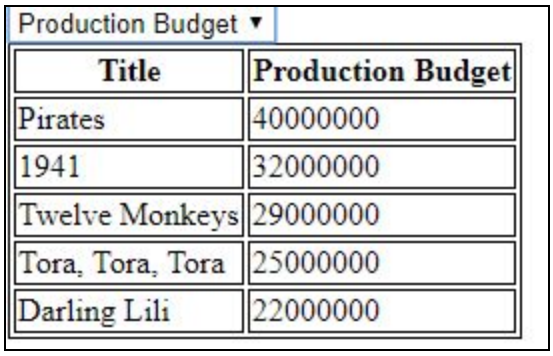
You will probably go through the following steps:

- To display the sorted table, you will need to write a generic comparison function that would take any fieldname (e.g. revenue, major genre) as a parameter.
- Then, you need to sort the data before you write it into the table.
- In your update function, whenever your selection changes, you need to sort the data by the new selection, and update both the movie title column and the column containing the data from the selected field.

Figure 26 gives example screenshots of different versions of the sorted table, depending on the column selected:



Title	Worldwide Gross
Four Weddings and a Funeral	242895809
Twelve Monkeys	168841459
3 Men and a Baby	167780960
1941	94875000
The Ten Commandments	80000000



Title	Production Budget
Pirates	40000000
1941	32000000
Twelve Monkeys	29000000
Tora, Tora, Tora	25000000
Darling Lili	22000000

Source Version : https://goo.gl/tsLwmc

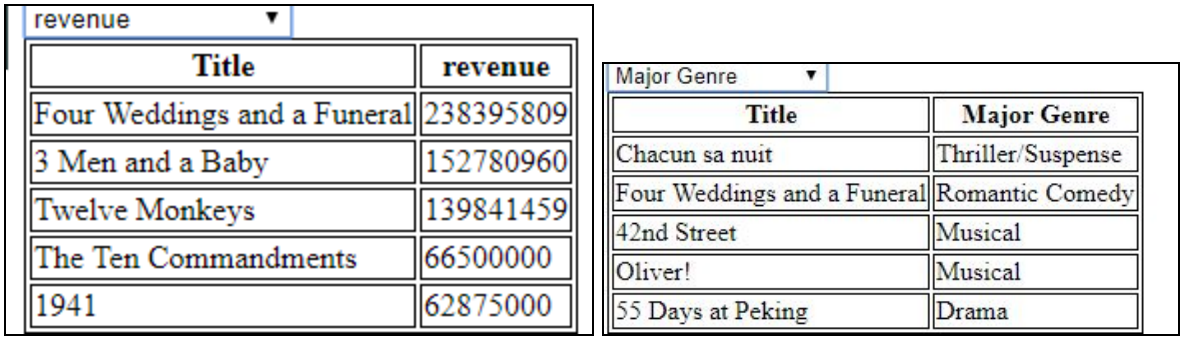


Figure 26