# Week 2 – What do you know about debugging??

Some of the student web sites I share, are from an older version of the class. Look at the approach they took to their project.

I have attached the associated Word doc. It is much more colorful. Probably better when looking at code. But start here then download and read at your leisure.

# Table of Contents

* What a first week!!!!
* Teamwork
* Look at the rubric
* Reminder on final individual project
* Office Hours for the week
* Templates
* Debugging
* Events

# What a first week!!!!

I know y’all are still dealing with some challenges with MS Teams. I try several times a day to day the unlisted students. I will keep doing that. Until then I have posted in each team the official list of all team members. Please contact those that are still not in the MS Teams group and include them in your team meetings.

# Teamwork

How team interactions are supposed to work

**Team card:**

**Study before meeting and collectively complete in the meeting**

The Team Trello card assignments are to be done DURING THE TEAM meeting. Study the problem and come to the meeting with your ideas for the solution. Then one person in the team acts as the typist and the team members contribute to the final solution. The one typing the code then submits the solution to the main branch of the repo and everyone synchs their local copies of the repo.

**Assigned Individual card:**

**Everyone will complete the card before team meeting. Chose the best version to add to repo**

The assigned individual cards are supposed to be coded by everyone in the team BEFORE the team meeting. Then in the team meeting everyone shares their ideas, and one member's solution is pushed to the main repo and everyone re-synchs their local copy.

**Self-chosen Individual card:**

**Everyone will choose a different card. Share with the team what you did.**

Then there is the individual card you chose. If the team is thinking ahead, hopefully everyone chooses a different card so you can learn from each other. This gets tricky. You will merge you code to the main branch and so will everyone else. Just be careful not to mess up the main branch.

I hope that helps with how the assignments are done. I know some of you are very competent is JS. The finish line is not the goal. Everyone is learning by doing, not seeing the solution after the fact. If some of you are too quick to solve the problem, you can inhibit others learning. This is a great time to take a back seat so everyone in the team can learn.

# Rubric

Each assignment has a Rubric. If you missed some points look at the rubric and the graders comments to see why you may have lost any points. The assignments are graded by a team of graders not me. If you have a question about a grade, put a comment in the assignment and they will respond.

# Final Individual Project

Be thinking, even now, about the final individual project. **A third-party API is required.** If you do not know what this is, please come to the Office Hours. This assignment is worth 60% of your grade. No you may not use a local JSON file.

* JavaScript: 25 points. you may **not** use Vue, React, Angular, jQuery, Bootstrap or any other framework. Just straight forward JavaScript
* Third-party API: 25 points. there are many to choose from. Search for ‘cool APIs’ or just go to RapidAPI to find something interesting. **Two** different end points are required. And come to my Office Hours. I will show MANY examples
* JSON: 15 points. Finding a good API will almost guarantee full points. What to look for: something that returns an array of data (list of movies, weather 10-day forecast, news stories) and each array element should have 8-10 data items.
* CSS: 10 points. be creative; card-flips, shadows, hover effects, the options are endless.
* Events: 10 points. Include 3-5 events. Click, hover, documentLoad. See examples below
* LocalStorage: 5 points. Save: options, favorites, searches. Restore on load

# Office Hours – Let’s talk

Office Hours this week

[https://byupw.zoom.us/my/**gtjames**](https://byupw.zoom.us/my/gtjames)

**Use this link to make sure you have the correct time for you time zone**

[**https://www.timeanddate.com/worldclock/converter.html?iso=20250202T030000&p1=tz\_ct&p2=tz\_mt&p3=1440**](https://www.timeanddate.com/worldclock/converter.html?iso=20250202T030000&p1=tz_ct&p2=tz_mt&p3=1440)

**Tuesday** 11:30 AM **CENTRAL**

**Tuesday** 9:00 PM **CENTRAL**

Please come I share many many examples and answer all the questions you have.

This week’s topics will be Debugging, Templates and JavaScript High Order Functions (HOF: filter, map, reduce).

If you need a different time, please contact me and we will arrange something.

# Templates – this will look better in the attached Word file

This week you will learn about template. Let me share a minimal code example.

Here is a github ‘gist’

<https://gist.github.com/gtjames/674c796b2c2440ee4ab463b51eaa277c>

Three files – 1 CSS style sheet and 2 HTML files, with imbedded JS

Both HTML files do the same thing: Use a Disney API to load all Disney characters

**Disney.html**  uses JS to create HTML text to be inserted into the page

**DisneyWithTemplate.html** uses an HTML <template> tag to create the very same look and feel for the page but using the template approach and DOM functions.

Both pieces of code receive an array of JSON data that looks like this. There are thousands of Disney characters. Download the files and test. Then get in the debugger to see how they do the same thing but with different code.

This is the JSON from the API request:

{

\_id": 6,

"name": "'Olu Mel",

"imageUrl": <https://static.wikia.nocookie.net/disney/images/6/61/Olu_main.png>,

"url": <https://api.disneyapi.dev/character/6>

}

**Disney.html** uses a JavaScript string literal to create the HTML to generate a ‘card’ to display on the web page for each character.

card.innerHTML += `  
 <div class="w3-col m4 l3 w3-theme-d${(row%5)+1} disney-card">  
 <a href="${character.sourceUrl}">${character.name}</a>  
 <br>  
 <img src="${character.imageUrl}" alt="">  
 <ul class='w3-theme-l${(row%5)+1}' id="movies">  
 ${li}  
 </ul>  
 </div>`;

(To me this is much clearer to show what is happening. Just my opinion)

**DisneyWithTemplate.html** uses a template tag and DOM to do the exact same thing.

Both blocks below are needed to accomplish the same task as the above block

**This is the temple tag inside the HTML page**

**<template id="cardTemplate">**

**<div id="div" class="w3-col m4 l3 disney-card">**

**<a id="a" href="">character.name</a>**

**<br>**

**<img id="img" src="" alt="">**

**<ul id="ul" class='w3-theme-l${(row%5)+1}' id="movies">**

**</ul>**

**</div>**

**</template>**

This is the JS which clones the template and uses DOM to create the ‘card’

**card.appendChild(*createCard*(character, li, ++row));**

**function *createCard*(character, li, row) {  
 const node = cardTemplate.content.cloneNode(true); //clone the HTML  
 const div = node.querySelector('#div')  
 const link = node.querySelector('#a')  
 const img = node.querySelector('#img')  
 const ul = node.querySelector('#ul')  
  
 div.class += ` w3-theme-d${(row%5)+1}`; //adds style   
  
 link.href = character.sourceUrl; // set link for the <a> tag  
 link.innerHTML = character.name; // set the text to click on  
  
 img.src = character.imageUrl; // show the image  
  
 ul.innerHTML = li; // add li to the <ul> tag  
 ul.class = `w3-theme-l${(row%5)+1}`; // add a little style  
  
 return node; // node is now compete!  
}**

# Debugging

This week you will spend time on debugging. There ought to be a Bachelor of Science in Debugging. You could write your own ticket for a job. It is about the most important skill you will develop. It is easy to write code. It is not easy to write, safe, defensive, supportable code without errors! You need to be methodical and analytical when designing and developing code. You need that and more to get your code clean, tested, and hardened. If you do not know how to use a debugger, then spend time on it this week and in the months and years to come. I can’t emphasize enough how critical this skill is. I will spend some time on debugging in this week’s Office Hours. Please come on by. It will be worth your time.

# JavaScript Events

We use a few events in the SleepOutside app. If you want to do more with them here are some resources for you. Events are required for the Final Project. Do not get trapped in the most basic superficial events (click and mouse over). Explore what is possible. Here is a link to a web page with an extensive list of events. <https://www.w3schools.com/jsref/dom_obj_event.asp>. Now go explore the possibilities.

Here are some examples from previous classes.

|  |  |
| --- | --- |
| <https://baldwin-casey.github.io/cit261/week9/javascript_events.html> | Casey Baldwin |
| <https://cboshy.github.io/JavaScript%20Events/JavaScript%20Events.html> | Chris Bollschweiler |
| <https://rikkor.github.io/CIT261/assignment6.html> | Eric Birch |
| <https://allen-zac.github.io/Allen-Zac-Fluency/fluency-code/js-events/javascript-events.html> | Zac Allen |
| <https://domenickcasper.github.io/CIT-261-BYUI/Week%205/JSEvents.html> | Domenick Casper |
| <https://kentroper.com/cit261/javascript_events.html> | Kent Roper |
| <https://codeunifier.github.io/schoolProject/topics/events/events.html> | Brian Evans |
| <https://hurst-justin.github.io/09-JavascriptEvents/index.html> | Dalan Ienatsch |
| <https://olive-spencer.github.io/CIT-261/a9.html> | Olive Spence |
| <https://mooremelissa.github.io/JavaScriptEvents.html> | Melissa Moore |