***Documentation Packet [ 06 07 10 22 ] Oct. 7th, 22***

|  |  |
| --- | --- |
| Student Name: |  |
| Goals:  1. Review Merging and PR DocPacs 2. Develop a daily routine 3. Contribute to the Formbar Project 4. Practice handling div layers | Events:  1. Weekly Review, Oct. 3rd 2. Print DocPacs, Oct. 6th 3. DocPac Due Oct. 7th |
| Included Documentation  1. [S] Weekly Contribution 2. [J] Daily Routine 3. [J] JSFighter 4. Routines Presentation 5. Reflection | Required Documentation:  1. [S] Weekly Contribution 2. [J] Daily Routine 3. [J] JSFighter 4. Reflection |
| Changes/Notes: none | |

# [J] Daily Routine

1. Create a flow chart that outlines your weekly routine.
2. The flow chart must cover every day of the week, and account for variations depending on what day it is.
3. You should also include subroutines to handle the specifics of frequent actions.
4. Make sure you account for any good habits you are trying to build, including allotting time to study/do homework.
5. Save this routine flowchart as “FirstnameLastname.vsdx” in the “DailyRoutine” in this DocPac folder.
6. Commit your changes and open a pull request from your fork/branch to the DocPac Repo’s main branch

# [J] Blackjack HTML

1. Create a simple Blackjack game in an HTML file
   1. If you do not know how blackjack is played, search for information online.
   2. For this game, you only need to worry about your hand (no dealer)
2. Each card in your pile must also be tracked in an array as **objects**.
   1. Each card object will have a *number, value, suit,* and *face* property
   2. Cards are numbered 1-13 (Ace through King)
   3. Cards with a number greater than 11 have a value of 11
   4. If the number is 1, the face is Ace. If it is 11, 12, 13, the face is Jack, Queen, King
3. If the total value of all cards you’ve drawn is less than 21, you may click a button to draw a new card. You may always click a button to start a new game.
4. **For each** card in the card pile array add a div to the page showing the card suit and value
5. Code you may need: getElementById(),createElement(), innerHTML, onclick, for..of
6. Save this HTML file as “FirstnameLastname.html” in the “Blackjack” folder in this DocPac folder.

# [J] JSFighter

1. In an HTML file, create a Fighter class with the following properties and methods
   1. name: The fighter’s name (like “Bob”)
   2. hp : A number tracking how many “health points” the fighter has
   3. atk : A number tracking how much extra damage a Fighter can do
   4. attack( target ) : A method that can reduce a target Fighter’s hp by a random number from 1 – 5 plus this fighter’s attack power. If the hp falls below 0, make it equal 0. If the target’s hp is already 0 or less, do not reduce the hp at all.
2. Create two new objects from the Fighter class (player1 and player 2)
3. Create a div layer containing an “Attack” button for each player object, and that player’s current ‘hp’ property.
   1. Each button will have on onclick event that calls its players “attack()” method, with the other player as the argument, and then updates that player’s hp on the page.
4. Save this HTML file as “FirstnameLastname.html” in a folder called “JSFighter” in this DocPac folder.
5. Commit and add to your PR for the week.

In short, you will have two boxes. Each will have a player’s fighter’s name and hp, and an Attack button. If you click on one of the attack buttons, it will reduce the other player’s fighter’s hp with an attack formula, and update the page.

# [S] Weekly Contribution

You must contribute for the csmith1188/formbar and document your contribution. A contribution is considered to be:

* Serious contribution to a issue, discussions, or documentation
* Advanced an issue that was stalled
* Fixed an issue (by solving)
* Completed a "section" or feature in documentation
* Detailed and ACCURATE comments on one module / file
* Expanded details on a "section" or feature
* Provide an idea in a discussion that is accepted as part of the project
* Performing an action that allows you to complete a card in a Project

In the box below, write the Issue #, Discussion #, PR #, or any relevant information that can easily point me to your work for grading

|  |
| --- |
|  |

# Reflection

**How do you feel about the independent work format of this class? In what ways does it help you and in what ways does it not?**

Okay great question! Honestly, I don’t think it’s the best format in the world but it works well enough. It does help that basically if you can manage to get it done you have a study hall essentially but there are some ways it doesn’t help. One, it’s pretty Confusing in general and quite a process half of the time to even start making the day’s current game, and two it’s just sorts of difficult if you don’t know exactly what you’re doing 100 percent of the time. I think you could fix this by providing a starting point, a reference, or a better guide as to what you’re doing. The RPG maker stuff had clear and concise directions and possible ways of doing things which laid out or gave you clear implications as what you needed to use. Technically the directions do the same thing but it’s not as understandable because there are multiple ways to write JavaScript so how to do it and what exactly you should do without any help or pointers unless I ask you. Other than that, it works perfectly fine it could just be better that’s all. Compared to something like Mr. Saia or Mr. Merkert’s it doesn’t work as well but it’s perfectly acceptable. And also, I get it’s not required but if you don’t know what you’re doing or you’re having trouble with something you better bring it home or homework and figure it out because If you don’t you’ll be behind and that’s sort of a problem. Technically the fact that I see you during lunch too sometimes, nullifies the problems a bit because I can tell you trust me considering you gave admin fob with keys to me. But overall, it’s perfectly fine.

**What has been your favorite project in class so far this year, and why?**

The Button Masher & JS Fighter. They we’re a little troublesome at times, but they we’re perfect. I only really struggled with Id’s being switched and how exactly you could make a turbo masher or how to make it so inputs are only accepted. That’s how you do a JavaScript game when it’s easier to figure out. I’ll say though Blackjack was a little easier to get working than Fighter because I am unfamiliar with basic RPGs.

# A picture containing text, monitor, screen, clipart Description automatically generatedGrading

|  |  |
| --- | --- |
| 100% | You went above and beyond expectations. You applied knowledge that was not taught in this class in addition to what was taught. |
| 100% | You performed as well as can be expected for this class. You show a complete understanding and made no mistakes. You have mastered the subject. |
| 90% | Assignment is complete. You show a good understanding of the subject, but there are mistakes or minor incorrect details. You are ready to move to new subjects. |
| 80% | You show and understanding of the subject, but there are serious errors, or there are pieces you can practically use without understanding them. Remediation needed. |
| 70% | Assignment is incomplete but/or you showed that you understand at least the fundamentals of the subject. Assignment is low effort. Serious need of remediation. |
| 60% | You show minimum effort, assignment is incomplete, or have serious mistakes. You did not demonstrate that you understand the content or purpose of the submission. |
| 0% | The work was not submitted, damaged, seriously incorrect, or unprofessional. The submission is rejected. |

# [S] Weekly Contribution

# [J] Daily Routine

# [J] JSFighter

# Reflection