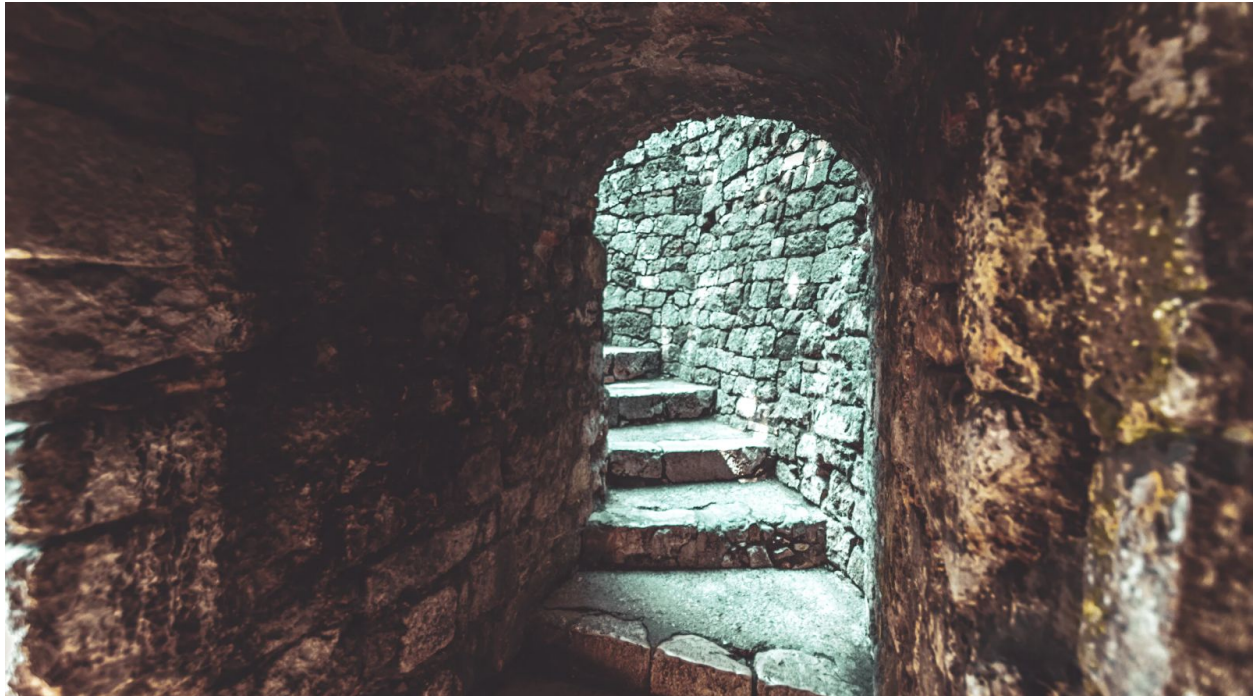# Web Development Course

Project 5: Choose-your-own Adventure Game

Big idea: Interaction and animations can make your site engaging and fun

Project 1: https://story-game-project.glitch.me/
Code: https://glitch.com/edit/#!/story-game-project

Project 2: https://adventure-tim.glitch.me/
Code: https://glitch.com/edit/#!/adventure-tim?path=README.md%3A1%3A0

Time: 35-45 minutes

5 minutes	Introduce project, More advanced CSS, layouts
5-10 minutes	Write/plan your game
20-25 minutes	Code the choose-your-own adventure game
5 minutes	Sharing and Discussion

Project Checklist

1. Include at least 5 pages that the player could find
2. Each web page should include images, hover-states, custom CSS for fonts and colors, and at least 2 links (unless it's the ending page)

Overview: About choose your own adventure games

Choose your own adventure games are a classic type of game where the player controls what they do and where they go. Their decisions affect the outcome of the story. There can be hidden messages and mini-games throughout to make the game even more fun!

Planning

Start by planning out the story on paper

Before any coding starts, have your students make a map of their stories. What pages will they need, what will they say, and what are your different endings?

Coding and Creating Files

Adding and linking internal pages

You can link to pages you create on Glitch the same way you link to external pages!

To make a new page, click "New file," and call it the name of the new file **.html.** For example, you can call it **page-2.html**, or the name of the puzzle, like **quiz.html** or **dragon.html**.

Make sure it has a .html at the end, that's how your browser knows to load it as an HTML page.

What you make a new page, make sure you **copy** the head over from index.html, and include a body tag.

```
New File ⌄

☐ assets
⚲ .env
README.md
index.html
page-2.html
page-3.html
puzzle-1.html        ⋮
script.js
style.css
```

```
1   <!DOCTYPE html>
2 ⌄ <html lang="en">
3 ⌄   <head>
4       <meta charset="utf-8">
5       <meta http-equiv="X-UA-Compatible" content="IE=edge">
6       <meta name="viewport" content="width=device-width, initial-scale=1">
7
8 ⌄     <title>The first puzzle</title>
9
10⌄     <!-- import the webpage's stylesheet -->
11      <link rel="stylesheet" href="/style.css">
12
13⌄     <!-- import the webpage's javascript file -->
14      <script src="/script.js" defer></script>
15    </head>
16⌄   <body>
17
18    </body>
19
20  </html>
```

To link to an **internal page**, you don't have to put in the whole URL. You can just type:
/name-of-file.html
Or in the example above
/puzzle-1.html

The whole link then, would be:
****

If you want a button to be clickable, **nest it inside the link.** For example in **HTML** write:
****
        <button>Start Here</button>
****

You can do this with anything! If you want images to be links, you can do the same:
****
        <img src="door.png">
****

Hover states

One of the tools you have during this project is that you can use **hover states** to hide and change elements on your website. **Hover states** are when an element **changes** when your user hovers their mouse over it. This is done in CSS.

One fun thing to do is add a hover state to a button. Set the base style in CSS, and then change it on hover. For example:

```
/* base style - applied to every button */
button {
      background-color: red;
      font-weight: bold;
}


/* these states are only visible if someone hovers over the button with their
cursor */
button:hover {
      background-color: purple;
      color: white;
      padding: 20px;
}
```

Bring the code above into your project! How can you customize the buttons in your work. What about links, images, or other elements? Can you hover over an image to make it bigger? How about making links italic on hover? Hover states can make your website feel more professional and dynamic. In the context of this game, they can make it feel more playful, personalized and immersive.

You can also use hover states to **hide objects in your game.**

If you wanted to hide an image for example, except when someone **hovers** over it, you could write in **HTML**:

```
<img class="hidden" src="my-image.jpg">
```

The hidden class won't do anything on its own, but in CSS you can add

```
.hidden {
      opacity: 0;
}

.hidden:hover {
      opacity: 1;
}
```

You can use this to hide things like buttons and secret doors from players, unless they can be sneaky and find them.

Transitions

With transitions, you can get hover events to happen in an animated way. This can make the interactions feel less abrupt and more game-like.

```
div {
  width: 100px;
  height: 100px;
  background: red;
  transition: width 2s;
}

div:hover {
  width: 300px;
}


.hidden {
    opacity: 0;
    Transition: opacity, 3s;
}

.hidden:hover {
    opacity: 1;
}
```

Animations

You're going to use a third-party library to add animations.

Here's the library we're going to use: https://www.w3schools.com/w3css/w3css_animate.asp
A library is a collection of someone else's code that your can link to and use in your project. There are a lot of animation libraries, but this one is lightweight (not a lot of code) and easy to use. Open up the link and read through it yourself. This is called documentation, and you'll want to get familiar with reading it when you're making your own decisions about what external libraries to use in your future websites.

Link to the library in your page's header. Add the following code inside the <head> and </head> tags on the webpage you want to add animations to:

<link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">

Then, all you have to do is add a **class** specified by this library to an element, and it will animate in.

For example:
```
<img class="w3-animate-top" src="img.jpg>
```

Will make your image animate in from the top of the screen when the page first loads.

Here is the list of possible animations:

```
w3-animate-top    Slides in an element from the top (-300px to 0)
w3-animate-bottom Slides in an element from the bottom (-300px to 0)
w3-animate-left   Slides in an element from the left (-300px to 0)
w3-animate-right  Slides in an element from the right (-300px to 0)
w3-animate-opacity    Animates an element's opacity from 0 to 1 in 1.5
seconds
w3-animate-zoom   Animates an element from 0 to 100% in size
w3-animate-fading Animates an element's opacity from 0 to 1 and 1 to 0 (fades
in + fade out)
w3-spin           Spins an element 360 degrees
```

Try it out! What animation matches the theme of your game?

Putting it all together

You and your students have all the pieces to make an awesome Choose your own adventure game. Check out the examples, plan your own, and then build it! Include at least 5 pages, including winning and losing pages that your players can find.

Sharing and Discussion
- What are some of the strategies you used to make your story more interactive?
- Was it hard to organize your story? What did you do to make it easier?
- What were some of the creative ways you hid things from the player?