

```
1 import './style.css'
2
3 let rpcOptions = document.getElementsByClassName("option")
4 let stage = document.getElementById("location")
5 let desc = document.querySelector(".desc")
6 let maincontent = document.getElementById("maincontent")
7 let location = document.getElementById("location")
8
9 const stagesetchoices = {
10   cerbChoices : [
11     {
12       counter : rpcOptions.paper,
13       beats : rpcOptions.scissors,
14       ties : rpcOptions.rock,
15       image : "/carb_rock.png"
16     },
17     {
18       counter : rpcOptions.scissors,
19       beats : rpcOptions.rock,
20       ties : rpcOptions.paper,
21       image: "/cerb_paper.png"
22     },
23     {
24       counter: rpcOptions.rock,
25       beats: rpcOptions.paper,
26       ties: rpcOptions.scissors,
27       image: "carb_scissor.png"
28     },
29   ],
30   LuciferChoices_Norm : [
31     {
32       counter : rpcOptions.paper,
33       beats : rpcOptions.scissors,
34       ties : rpcOptions.rock,
35       image: "luc_rock.png"
36     },
37     {
38       counter : rpcOptions.scissors,
39       beats : rpcOptions.rock,
40       ties : rpcOptions.paper,
41       image: "luc_pap.png"
42     },
43     {
44       counter: rpcOptions.rock,
45       beats: rpcOptions.paper,
46       ties: rpcOptions.scissors,
47       image: "luc_scissors.png"
48     }
49   ]
50 }
```

```
9  const stagesetchoices = {
10    LuciferChoices_Norm : [
11      {
12        beats : rpcOptions.scissors,
13        ties : rpcOptions.rock,
14        image: "luc_rock.png"
15      },
16      {
17        counter : rpcOptions.scissors,
18        beats : rpcOptions.rock,
19        ties : rpcOptions.paper,
20        image: "luc_pap.png"
21      },
22      {
23        counter: rpcOptions.rock,
24        beats: rpcOptions.paper,
25        ties: rpcOptions.scissors,
26        image: "luc_scissors.png"
27      },
28      {
29        counter: null,
30        beats: [rpcOptions.paper, rpcOptions.rock, rpcOptions.scissors],
31        ties: null,
32      }
33    ],
34    LuciferChoices_Exp : [
35      {
36        counter : rpcOptions.paper,
37        beats : rpcOptions.scissors,
38        ties : rpcOptions.rock,
39        image: "luc_rock_exp.png"
40      },
41      {
42        counter : rpcOptions.scissors,
43        beats : rpcOptions.rock,
44        ties : rpcOptions.paper,
45        image: "luc_paper_exp.png"
46      },
47      {
48        counter: rpcOptions.rock,
49        beats: rpcOptions.paper,
50        ties: rpcOptions.scissors,
51        image: "luc_scissors_exp.png"
52      },
53      {
54        counter: null,
55        beats: [rpcOptions.paper, rpcOptions.rock, rpcOptions.scissors],
56        ties: null,
57        image: "luc_spellcast_exp.png"
58      }
59    ]
60  }
```

```

        image: "luc_spellcast_exp.png"
    }
}

if (rpcOptions.cross) {stagesetchoices.LuciferChoices_Exp[3].counter = cross}

function randombotChoice(min, max) {
    return Math.floor(Math.random() * (max - min + 1)) + min
}
function matchoptions(option, botset, botchoice, nextRoom, LoseRoom, tieRoom){
    console.log(botset[botchoice].counter.id)
    if (botset[botchoice].counter !== null){
        if (botset[botchoice].counter.length > 1){
            botset[botchoice].counter.forEach(counter => {
                if (option.target.id == counter.id) {

                    if (!nextRoom){
                        return "win"
                    }else{
                        console.log("WIN")
                        window.location.href = nextRoom;
                    }
                }
            })
        }
        else if (option.target.id == botset[botchoice].counter.id){
            console.log("WIN")

            if (!nextRoom){
                return "win"
            }
            window.location.href = nextRoom;
        }
    }
    if (botset[botchoice].beats !== null){
        if (botset[botchoice].beats.length > 1){
            botset[botchoice].beats.forEach(beats => {
                if (option.target.id == beats.id) {
                    console.log("LOSE")
                    if (!loseRoom){

```

```

120         botset[botchoice].beats.forEach(beats => {
121             console.log( "LOSE" )
122             if (!loseRoom){
123                 console.log("cancelled")
124                 return "lose"
125             }
126             window.location.href = loseRoom;
127         }
128     })
129 }
130 else if (option.target.id == botset[botchoice].beats.id){
131     console.log("LOSE")
132     if (!loseRoom){
133         console.log("cancelled")
134         return "lose"
135     }
136     window.location.href = loseRoom;
137 }
138 }
139 }
140 if (botset[botchoice].ties !== null){
141 if (botset[botchoice].ties.length > 1){
142     botset[botchoice].ties.forEach(tie => {
143         if (option.target.id == tie.id) {
144             console.log("TIE")
145             if (!tieRoom){
146                 return "tie"
147             }
148             window.location.href = tieRoom;
149         }
150     })
151 }
152 else if (option.target.id == botset[botchoice].ties.id){
153     console.log("TIE")
154     if (typeof tieRoom == undefined){
155         return "tie"
156     }
157     window.location.href = tieRoom;
158 }
159 }
160 }
161
162 function addSelection(options, botset, botchoice, nextRoom, loseRoom, tieRoom){
163     for (let i = 0; i < options.length; i++){
164         rpcOptions[i].addEventListener("click", (el)=>{
165             el.preventDefault()
166             return matchoptions(el, botset, botchoice, nextRoom, loseRoom, tieR
167         })
168     }

```

```

174 function battleSetup(min, max, botset, nextRoom, loseRoom, tieRoom){
175     let botchoice = randombotChoice(min,max)
176     console.log(botset[botchoice].image)
177     maincontent.src = botset[botchoice].image
178     return addSelection(rpcOptions, botset, botchoice, nextRoom, loseRoom, tieRoom)
179 }
180 }
181
182 if (stage.dataset.room == "cerb"){
183     battleSetup(0, 2, stagesetchoices.cerbChoices, "/start_win.html", "/start_lose.html", "/start_tie.html")
184 }
185 else if (stage.dataset.room == "lucifer"){
186     battleSetup(0, 2, stagesetchoices.luciferChoices_Norm, "/Lucifer_Inter", "/lucifer_lose.html", "lucifer_tie.html")
187 }
188 else if (stage.dataset.room == "lucifer2"){
189     maincontent.style.width = "62%"
190     location.style.marginBottom = "0"
191     battleSetup(0, 3,stagesetchoices.luciferChoices_Exp, "/lucifer_win", "/lucifer_lose.html", "lucifer_tie.html")
192 }
193
194
195
196
197 function changedesc(i, text, redirect){
198     if (i < text.length){
199         desc.textContent = text[i].text
200         if (text[i].image) {
201             maincontent.src = text[i].image
202         }
203     } else{
204         window.location.href = redirect
205     }
206 }
207
208 let luciferDialogue = [
209     {text:"I see that you have made it, damnedest.\\"},
210     {text:"My name and title is Lucifer, the bearer of this land.\\"},
211     {text:"Your will is doomed, but, there is one way to repent\\"},
212     {text:"Face me, and escape your fate... through the noble game of ドゥルケンシュート, 가위바위보, Forfices Saxorum et Chartarum or... ROCK, PAPER, SCISSORS.\\"},
213     {text:"\u00d7ESCAPE from the sin of... gambling...?"},
214     {text:"Ok yeah you've definitely skipped a step."}
215 ]
216
217 if (stage.dataset.room == "throne"){
218     let i = 0
219
220     document.querySelector('.option').addEventListener("click", (e)=>{

```

↳ YH4-ES (14 minutes ago) Ln 84, Col 78 Sp

```

220     document.querySelector('.option').addEventListener("click", (e)=>{
221         e.preventDefault()
222         changedesc(i, luciferDialogue, "/LuciferFight.html")
223         i++
224     })
225 }

```