

## # Full Stack Cat App VII

### ## Adding functionality and style to Site

\* Since the focus of this exercise is the routing and the back-end in particular, I am including the code for a simple javascript game that uses the User's age. The game comes dynamically out of `#"gameDiv"` in `rest-of-site.ejs`.

```
....  
<div class="profile-bar-div">  
<%- include('./partials/profile-bar.ejs') %>  
</div>  
<div class="portal-container site-container" style="background-image:url(<%= fci %>);">  
<div class="portal-content-wrapper game-content-wrapper">  
<div class="portal-content game-content">  
<div hidden id="name" data-value="<%= name %>"></div>  
<div hidden id="age" data-value="<%= age %>"></div>  
<div hidden id="fci" data-value="<%= fci %>"></div>  
<div hidden id="id" data-value="<%= id %>"></div>  
<div id="gameDiv" class="game-div"></div>  
</div>  
</div>  
</div>
```

....

Here is the code:

### ### Cat Game

```
```javascript  
const nameDiv = document.getElementById("name");  
const name = nameDiv.getAttribute('data-value');  
console.log(name);  
const ageDiv = document.getElementById("age");  
const age = ageDiv.getAttribute('data-value');  
const randomNum = (max, min) => Math.floor(Math.random() * (max - min + 1)) + min;  
const makeRange = (age) => {  
  if (age < 10) {  
    return randomNum(10, 1);  
  } else if (age < 20) {  
    return randomNum(20, 1);  
  } else {  
    return randomNum(100, 1);  
  }  
}
```

```
    }  
};
```

```
let randomNumber = 0;  
let score = 0;  
let wins = 0;  
let losses = 0;  
let val1 = 0;  
let val2 = 0;  
let val3 = 0;
```

```
document.addEventListener('DOMContentLoaded', () => {  
    setUpDivs();  
    setUpGame();  
    gameNumbers();  
});
```

```
function setUpGame() {  
    targetNum = 0;  
    playerNum = 0;  
    randomNumber = randomNum(20,1);  
    targetNum = age * randomNumber;  
    val1 = makeRange(age);  
    val2 = makeRange(age);  
    val3 = makeRange(age);  
    scoreDiv.innerHTML = "Score: " + score;  
    winsDiv.innerHTML = "<p>Wins: " + wins + "</p><div class='wins-losses-img-div  
wins-img-div'></div>";  
    lossesDiv.innerHTML = "<p>Losses: " + losses + "</p><div class='wins-losses-img-div  
losses-img-div'></div>";  
    playerNumber.innerHTML = "Player Number: " + playerNum;  
    targetNumber.innerHTML = "Game Number: " + targetNum;  
    target1.innerHTML = val1;  
    target2.innerHTML = val2;  
    target3.innerHTML = val3;  
}
```

```
function gameNumbers() {  
    target1.addEventListener('click', () => {  
        play(val1);  
    });  
    target2.addEventListener('click', () => {  
        play(val2);  
    });  
}
```

```
});  
target3.addEventListener('click', () => {  
    play(val3);  
});  
}
```

```
function play(val) {  
    playerNum = playerNum + val;  
    playerNumber.innerText = "Player Number: " + playerNum;  
    match(playerNum, targetNum);  
}
```

```
function match(playerNum, targetNum) {  
    if (playerNum > targetNum) {  
        lose();  
    }  
    if (playerNum === targetNum) {  
        win();  
    }  
}
```

```
function lose() {  
    losses++;  
    lossesDiv.innerText = "Losses: " + losses;  
    matchScores();  
    setUpGame();  
}
```

```
function win() {  
    wins++;  
    winsDiv.innerText = "Wins: " + wins;  
    matchScores();  
    setUpGame();  
}
```

```
function matchScores() {  
    console.log('matchedScores');  
    if (losses === 3) {  
        score--;  
        wins = 0;  
        losses = 0;  
        scoreDiv.innerText = "Score: " + score;  
        winsDiv.innerText = "Wins: " + wins;  
    }  
}
```

```

        lossesDiv.innerText = "Losses: " + losses;
    } else if (wins === 3) {
        score++;
        wins = 0;
        losses = 0;
        scoreDiv.innerText = "Score: " + score;
        winsDiv.innerText = "Wins: " + wins;
        lossesDiv.innerText = "Losses: " + losses;
    }
}

```

```

function setUpDivs(){
    const scoreWrapper = document.createElement("div");
    scoreWrapper.setAttribute("id", "scoreWrapper");
    scoreWrapper.setAttribute("class", "score-wrapper");
    const scoreDiv = document.createElement("div");
    scoreDiv.setAttribute("id", "scoreDiv");
    scoreDiv.setAttribute("class", "score-div");
    scoreWrapper.appendChild(scoreDiv);

    const winsDiv = document.createElement("div");
    winsDiv.setAttribute("id", "winsDiv");
    winsDiv.setAttribute("class", "wins-and-losses-div wins-div");
    scoreWrapper.appendChild(winsDiv);

    const lossesDiv = document.createElement("div");
    lossesDiv.setAttribute("id", "lossesDiv");
    lossesDiv.setAttribute("class", "wins-and-losses-div losses-div");
    scoreWrapper.appendChild(lossesDiv);

    const targetWrapper = document.createElement("div");
    targetWrapper.setAttribute("id", "targetWrapper");
    targetWrapper.setAttribute("class", "target-wrapper");
    const targetNumber = document.createElement("div");
    targetNumber.setAttribute("id", "targetNumber");
    targetNumber.setAttribute("class", "target-number");
    targetWrapper.appendChild(targetNumber);
    const playerNumber = document.createElement("div");
    playerNumber.setAttribute("id", "playerNumber");
    playerNumber.setAttribute("class", "player-number");
    targetWrapper.appendChild(playerNumber);

    const numberWrapper = document.createElement("div");

```

```
numberWrapper.setAttribute("id", "numberWrapper");
numberWrapper.setAttribute("class", "number-wrapper");
```

```
const target1 = document.createElement("button");
target1.setAttribute("id", "target1");
target1.setAttribute("class", "game-btn target1");
numberWrapper.appendChild(target1);
```

```
const target2 = document.createElement("button");
target2.setAttribute("id", "target2");
target2.setAttribute("class", "game-btn target2");
numberWrapper.appendChild(target2);
```

```
const target3 = document.createElement("button");
target3.setAttribute("id", "target3");
target3.setAttribute("class", "game-btn target3");
numberWrapper.appendChild(target3);
```

```
const gameDiv = document.getElementById("gameDiv");
gameDiv.appendChild(scoreWrapper);
gameDiv.appendChild(targetWrapper);
gameDiv.appendChild(numberWrapper);
```

```
}
```

```
....
```

### ### Game Styles

\* Here is game.css, the styling is carried over from other parts of the site, just customized to the game. You should consider doing the same, in order to have a cohesive style over many views:

```
....
```

```
.game-content {
  background-color: rgba(232, 233, 227, 0.8);
  padding-top: 0px;
  margin-top: -50px;
}
```

```
.game-content-wrapper {
  height: 100%;
  width: 100%;
```

```
padding-top: 0;  
}
```

```
.game-div {  
  height: 100%;  
  width: 100%;  
  max-width: 100%;  
  max-height: 100%;  
  display: flex;  
  flex-direction: column;  
}
```

```
.score-wrapper {  
  width: 100%;  
  height: 100%;  
  display: flex;  
  justify-content: space-around;  
  font-size: 2em;  
  margin: 1em 0em 1em 0em;  
}
```

```
.age-wrapper {  
  width: 100%;  
  height: 100%;  
  display: flex;  
  justify-content: center;  
  font-size: 1.5em;  
  margin: 1em 0em 1em 0em;  
}
```

```
.target-wrapper {  
  width: 100%;  
  height: 100%;  
  display: flex;  
  justify-content: space-around;  
  font-size: 2em;  
  margin: 1em 0em 1em 0em;  
}
```

```
.number-wrapper {  
  width: 100%;  
  height: 100%;  
  display: flex;
```

```
    justify-content: center;
}
```

```
.wins-and-losses-div {
    display: flex;
    flex-direction: column;
}
```

```
.wins-losses-img-div {
    height: 100px;
    width: 100px;
    border-radius: 50%;
    background-size: cover;
    background-size: cover;
    background-position: center;
}
```

```
.wins-img-div {
    background-image: url('../images/whosahappykitty.gif');
}
```

```
.losses-img-div {
    background-image: url('../images/angrycat.gif');
}
```

```
.game-btn {
    margin: 10px;
    font-size: 8em;
    height: 200px;
    width: 200px;
    border-radius: 50%;
    background-size: cover;
    background-size: cover;
    background-position: center;
    outline: none;
    border: none;
}
```

```
.target1 {
    background-image: url('../images/catdance.gif');
}
```

```
.target2 {
```

```
    background-image: url('../images/mew2.gif');  
}
```

```
.target3 {  
    background-image: url('../images/spinnercat.gif');  
}
```

....

### Additionally there is now a how-to page, which loads from a button on the game view "nav bar".

### This project is now functional, and should be styled to stand as it is.

### What's next?

\* Finally, let's see if we can save the player's score and how many times the player has played the game.