Mark Gutierrez 02/26/2020

Homework 4

1. What are for loops good for?

   a. They are good to run a block of code a number of times.
   b. They are very good to use alongside arrays

2. For loop syntax

   a. Basic for loop syntax:

```
for ([initialization]; [condition]; [final-expression])

statement
```

   b. Complex syntax for loop:

```
for (i = 0; i < 8; i++) {
  text += "Number count: " + i + "<br>";
}
```

3. What does for loops do?

It loops through properties of objects.

4. for loop code example breakdown:

```
let weaponList = ["Sword", "Lance", "Axe", "Bow"];

let i = 0;

let weaponType = weaponList.length;

let text = "";

for (; i < weaponType;) {

        text += weaponList[i] + "<br>";

        i++;

}

document.querySelector(".output_text").innerHTML = text;
```

In a for loop only the condition is required. The initialization and final-expression are optional inside the for lop. However, a semicolon is still required before the condition. In this code example I initialized the variable “i” in a global scope for the sake of showing another way; it’s better not to declare variables in the global scope unless is absolutely necessary, because it can slow the browser. The variables are initialized and assigned, “let weaponList” is an array, “let weaponType” is getting the length of the “WeaponList” array; which is 4. The for loop has a condition that is checking if the variable “i” is less than the length of “weaponList” which is assigned to “weaponType.” If the condition is true, which it is, then run the statement ‘text += weaponList[i] + “<br>”;’ Text is a variable with an empty value. “weaponType[i]” means that its going through the array using I. so if “i” equals 0 then get the first array value aka “Sword.” You can write the syntax as follow to get the same array result/ first array value:

```
let playerClass = ["Warrior", "Paladin", "Pirate"];

console.log(playerClass[0]);
```

By using the array followed by brackets and a number inside the brackets, “playerClass[0]”, you can get a specific value. In JavaScript we count from 0 and on. 0 in this case is the first value. For this example, we want to get all the array’s value one by one via the for loop using ‘i’. As mentioned above, for loops are very good alongside arrays. Once the variable “i” is not less than “<” the length of weaponList then the loop stops. We inserted a “<br>” element after so we get a list instead of a line of text in the html. Again, for this particular code example I decided to have the final-expression “i++;” after the statement to show that is an option. You can just write it after the condition if you so choose to. The final-expression “i++;” increases the variable “i” every time the condition “for (;i < weaponType;)” is true. The variable “i” ends up being 4 after the for loop is over and so we get all the array values one after the other.

5. What are two characters represent the “for loop” code block?

Curly braces {}

6. After the for loop code block has been executed, which step does the for loop return to?

The for loop returns to step 2, the condition.

7. For loop if statement and break statement inside if statement

Code example:

```
let swordLevel = "";
for (let swordAttack = 1; swordAttack < 100; swordAttack++) {
   if (swordAttack > 20) {
document.querySelector(".damage_cal").innerHTML = swordLevel;
   break;
   }
   swordLevel +=  `Sword has reached ${swordAttack} Attack!` + `<br>`;
}
document.querySelector(".damage_cal").innerHTML = swordLevel;
```

8. For loop if statement and continue statement inside if statement

Code example:

```
        let field = "";
        for (apples = 1; apples < 10 ; apples++) {
           if (apples === 3) { continue; }
           field +=`${apples} apple collected! <br>`;
        }
        const appleCollected = document.querySelector(".field");
        appleCollected.innerHTML = field;
```

9. Array example string

Code example:

```
        let promoteClass = ["Mage", "Thief", "cleric"];
        console.log(promoteClass);
```

10. Array example numbers

Code example:

```
    let countNum = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];
    console.log(countNum);
```

```
11. let loopArrayDataType = ["string", true, 0, false];
let allDataType = loopArrayDataType.length;
let arrayResult = "";

for (i = 0; i < allDataType; i++) {
arrayResult += loopArrayDataType[i] + "<br>";
}
let loopArrayShow = document.querySelector(".data_type");
loopArrayShow.innerHTML = arrayResult;
```

12. For-in loop explained

Code example:

```
    let person = {
    name: "Katharina",
    haircolor: "Blonde",
    eyeColors: "Blue"
    };
    personInfo = "";
    for (let x in person){
    personInfo += person[x] + "<br>";
    }
    document.querySelector(".Person").innerHTML = personInfo;
```

The for-in loop can go through objects and get the properties. I have a for-in loop “person” and it has the properties “name, hairColor, eyeColors.” With a for-in loop the block of code inside the loop will be executed once for each property. So “Katharina”, “Blonde”, and “Blue” will be returned.

13. For of loop example

Code example:

```
const cardType = ["Common", "Rare", "Legendary", "Exclusive"];

for (const cardTypeValue of cardType) {

console.log(cardTypeValue);

}
```

For of loop can go to through the values of an iterable object. In this case I am looping “cardType’ with a for of loop to get the array values.

14. For in vs for of

The main difference is that for of can only loop through iterable objects.

15. Do while loop

```
do {

i_var = i_var + 1;

doResult = doResult + i_var;

} while (i_var < 5);

console.log(doResult);
```

16. Why does a loop terminate

A loop terminates because the condition is no longer true.