**Lab 4a The Phrase-O-Matic**

Below is the full code for the mad libs program. Please note that in the code, there are comments that describe why I wrote it that way. Sorry if there are a lot of comments but I was just trying to follow the instructions.

You will find there is just minor styling so when I take a screen shot of the output, it is clear to you sir.

Github repository for a cleaner code view available in this link: <https://github.com/AhmedAbdelRazak/RCC/tree/master/Chapter4>

<!DOCTYPE html>

<html lang="en">

    <head>

        <meta charset="UTF-8" />

        <title>Phrase-o-matic</title>

    </head>

    <body

        style="

            padding-right: 1000px;

            padding-left: 50px;

            margin-top: 30px;

            font-size: 1.7rem;

        "

    >

        <script>

            function makePhrases() {

                var adjective = [

                    "brave",

                    "famous",

                    "better",

                    "delightful",

                    "victorious",

                    "sticky",

                    "clever",

                    "bewildered",

                    "fast",

                    "powerful",

                    "adorable",

                ];

                var noun = [

                    "hotdog",

                    "vacation",

                    "basketball",

                    "dog",

                    "playground",

                    "bathtub",

                    "goose",

                    "shoe",

                    "beach",

                    "honey",

                ];

                var pluralNoun = [

                    "beaches",

                    "lunches",

                    "vacations",

                    "towels",

                    "shoes",

                    "trees",

                    "bottles",

                    "houses",

                    "baskets",

                    "lambs",

                ];

                var adverb = [

                    "fast",

                    "slow",

                    "brightly",

                    "softly",

                    "furiously",

                    "joyfully",

                    "mostly",

                    "cheerfully",

                    "calmly",

                    "freely",

                ];

                var verb = [

                    "run",

                    "swim",

                    "tickle",

                    "talk",

                    "slap",

                    "open",

                    "mix",

                    "walk",

                    "close",

                    "look",

                ];

                //I commented the below variables out just to follow the lecture instructions and I renamed them accordingly as advised by the professor.

                //

                // var rand1Adjustective = Math.floor(Math.random() \* adjective.length);

                // var rand2Noun = Math.floor(Math.random() \* noun.length);

                // var rand3PluralNoun = Math.floor(Math.random() \* pluralNoun.length);

                // var rand4Adverb = Math.floor(Math.random() \* adverb.length);

                // var rand5Verb = Math.floor(Math.random() \* verb.length);

                //

                //

                // In the phrase, I noticed that there are 6 adjectives and according to the lecture, I understood that it is required to generate different adjectives, thus I will create an array of random and unique indexes for array adjective. The code block below should give me another array of random unique numbers from 0 to 9 and the length of this array will be 6 since I need 6 unique adjectives

                var AdjectiveIndexes = [];

                for (var i = 0; i <= 5; i = i + 1) {

                    //x is a helper variable to store individual value in each iteration.

                    //I did adjective.length - 1 because I don't want the index to exceed the 9 so it renders real values not undefined when using AdjectiveIndexes in the Phrase Variable

                    //Please note that the objective.length is 11 not 10.

                    var x = Math.floor(Math.random() \* (adjective.length - 1));

                    //here I was just testing the values of x in the console.

                    console.log(x, "this is x");

                    //I know indexOf function was not mentioned in the lecture, however, it has to be used in this scenario to know whether the value of x was pushed to the AdjectiveIndexes before or not.

                    //IndexOf return the index/ position of the value passed in a specific array.

                    // if indexOf returned an index of -1, therefore, the value passed is not in the array.

                    if (AdjectiveIndexes.indexOf(x) === -1) {

                        AdjectiveIndexes.push(x);

                    } else {

                        //Here, I'm validating if I added 1 to x, will the value after adding 1 be available in the array or not.

                        if (AdjectiveIndexes.indexOf(x + 1) === -1) {

                            AdjectiveIndexes.push(x + 1);

                        } else {

                            AdjectiveIndexes.push(x + 2);

                        }

                    }

                }

                //

                //I have coded the phrase variable as the lecture however, it kept generating duplicates when I use it that way thus I re-defined it

                //

                // var phrase =

                //  adjective[Math.floor(Math.random() \* adjective.length)] +

                //  " teachers always give out " +

                //  adjective[Math.floor(Math.random() \* adjective.length)] +

                //  " assignments. But as everyone knows, if you want to pass all your classes so you can to a/an " +

                //  noun[Math.floor(Math.random() \* noun.length)] +

                //  " and become president of a big international " +

                //  noun[Math.floor(Math.random() \* noun.length)] +

                //  " and have millions of " +

                //  pluralNoun[Math.floor(Math.random() \* pluralNoun.length)] +

                //  " in the bank, you must do your homework and study " +

                //  adverb[Math.floor(Math.random() \* adverb.length)] +

                //  ". if you just sit around and " +

                //  verb[Math.floor(Math.random() \* verb.length)] +

                //  ", you won't get ahead in life. you must learn to pay attention to every " +

                //  adjective[Math.floor(Math.random() \* adjective.length)] +

                //  " thing your teacher says. Do not interrupt or whisper to other " +

                //  pluralNoun[Math.floor(Math.random() \* pluralNoun.length)] +

                //  " during class. Be sure to have a nice, " +

                //  adjective[Math.floor(Math.random() \* adjective.length)] +

                //  " notebook in which you can write down anything the teacher sayd that seems " +

                //  adjective[Math.floor(Math.random() \* adjective.length)] +

                //  ". Then go home and memorize all of those " +

                //  adjective[Math.floor(Math.random() \* adjective.length)] +

                //  " notes. When your teacher gives a surprise quiz, you will know all of the " +

                //  pluralNoun[Math.floor(Math.random() \* pluralNoun.length)] +

                //  ".";

                //Now, after testing, it looks like I will always have unique adjectives

                //I did not create unique indexes array for the other variables because they are not being used in the phrase as much as the adjectives so I noticed that there more likely won't be duplicated.

                // var vowels = ['a', 'e', 'i', 'o', 'u', 'y']

                var phrase =

                    adjective[AdjectiveIndexes[0]] +

                    " teachers always give out " +

                    adjective[AdjectiveIndexes[1]] +

                    // I really wanted to impress you when you mentioned in the lecture that determining it is "a" or "an" could be done :)

                    // For this, we will have to implement conditional rendering such that if the word after starts with a vowel (A, E, I, O, U, Y) ? use an : use a

                    // you will notice that I ~ only wrote the syntax `vowels.indexOf(noun[Math.floor(Math.random() \* noun.length)][0])===-1 ? a : an`

                    // I haven't wrote it the actual code because this code "noun[Math.floor(Math.random() \* noun.length)][0]" will be different since it is not stored a dedicative variable so I decided not to embed it in this program but I just wanted to present my idea, hopefully it is correct.

                    " assignments. But as everyone knows, if you want to pass all your classes so you can to " +

                    `a/ an ` +

                    noun[Math.floor(Math.random() \* noun.length)] +

                    " and become president of a big international " +

                    noun[Math.floor(Math.random() \* noun.length)] +

                    " and have millions of " +

                    pluralNoun[Math.floor(Math.random() \* pluralNoun.length)] +

                    " in the bank, you must do your homework and study " +

                    adverb[Math.floor(Math.random() \* adverb.length)] +

                    ". if you just sit around and " +

                    verb[Math.floor(Math.random() \* verb.length)] +

                    ", you won't get ahead in life. you must learn to pay attention to every " +

                    adjective[AdjectiveIndexes[2]] +

                    " thing your teacher says. Do not interrupt or whisper to other " +

                    pluralNoun[Math.floor(Math.random() \* pluralNoun.length)] +

                    " during class. Be sure to have a nice, " +

                    adjective[AdjectiveIndexes[3]] +

                    " notebook in which you can write down anything the teacher sayd that seems " +

                    adjective[AdjectiveIndexes[4]] +

                    ". Then go home and memorize all of those " +

                    adjective[AdjectiveIndexes[5]] +

                    " notes. When your teacher gives a surprise quiz, you will know all of the " +

                    pluralNoun[Math.floor(Math.random() \* pluralNoun.length)] +

                    ".";

                document.write(phrase);

                console.log(AdjectiveIndexes);

            }

            makePhrases();

        </script>

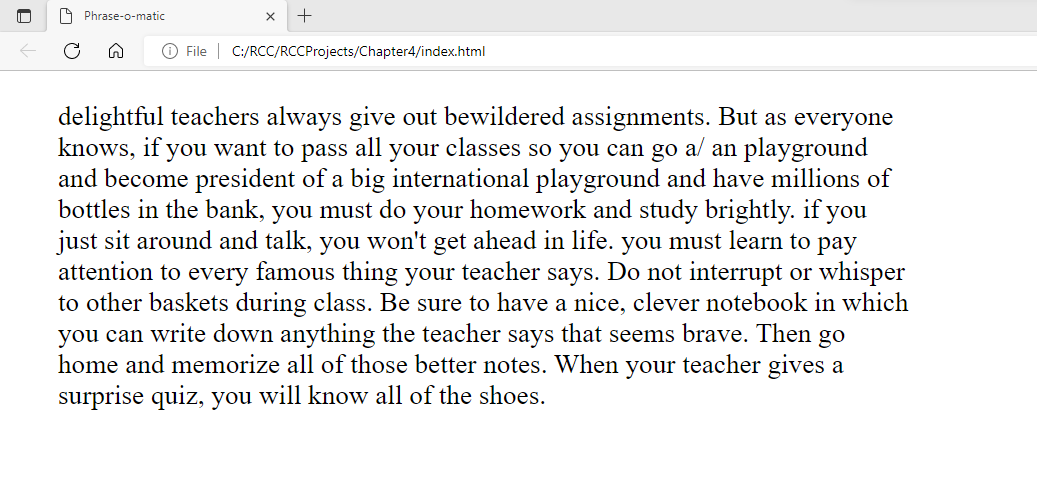
    </body>

</html>

Below is the final output of the program above:

I will paste 3 different outputs to ensure that the blanks are not generating duplicated words specially in the adjective array.

Output 1:



Output 2:

Text

Description automatically generated

Output 3:

A screenshot of a computer

Description automatically generated with medium confidence