Assignment 1:

**Part A:**

**Question 1:**

Source Code:

<!DOCTYPE>

<html>

<head>

<title>Question 1</title>

<p>Please insert the number of row and columns for the table:</p>

</head>

<body>

<script type="text/javascript">

function createTable(){

var row = document.getElementById("rowField");

var col = document.getElementById("colField");

var rows = parseInt(row.value);

var cols = parseInt(col.value);

document.write("<table border = '1px'>");

for (var i=1;i<=cols;i++){ //coloum

document.write("<tr style='height:30px;'>");

for(var j = 1; j<= rows;j++){ //row

document.write("<td>" + i\*j + "</td>");

}

document.write("</tr>");

}

document.write("</table>");

}

</script>

Row<br><input type = "text" id ="rowField" size = "10"><br>

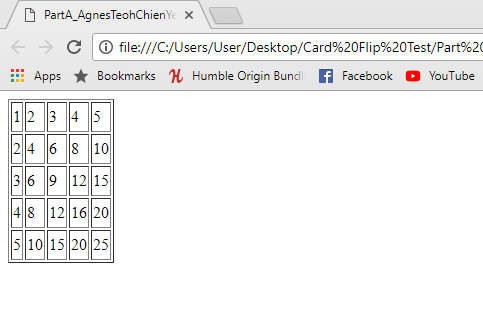
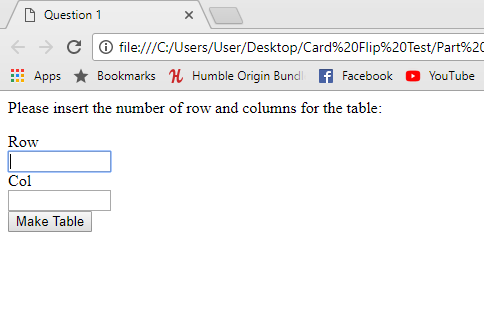
Col<br><input type = "text" id ="colField" size = "10"><br>

<input type = "button" onclick = "createTable()" value = "Make Table"><br>

</body>

</html>

Output / Screenshots:



**Question 2:**

Source Code:

<!DOCTYPE>

<html>

<head>

<title>Question 2</title>

<p>Select the rectangle to open Colour Picker and click Change to switch background colour:</p>

</head>

<body>

<input type = "button" onclick ="change()" value = "Change">

<input type = "color" id = "colour">

<script>

function change(){

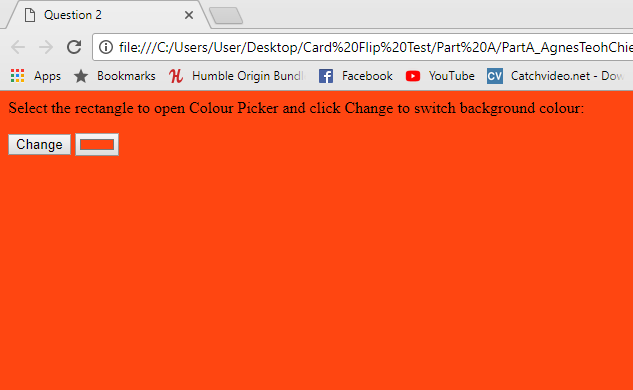
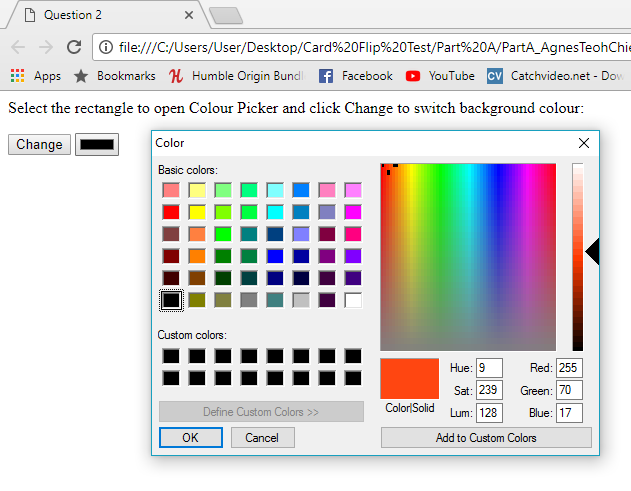
var color = document.getElementById("colour");

document.body.style.backgroundColor = color.value;}

</script>

</body>

</html>

Output / Screenshots: 

**Question 3:**

Source Code:

<!DOCTYPE>

<html>

<head>

<meta charset='utf-8'>

<title>Question 3</title>

<p>Pick an emoji and input the amount of times it should be displayed: </p>

</head>

<body>

<style>

select {

font-family: 'FontAwesome', 'Second Font name'

}

</style>

<link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.6.1/css/font-awesome.min.css" rel="stylesheet"/>

<select id="myEmoji">

<option>&#xf042;</option>

<option>&#xf043;</option>

<option>&#xf044;</option>

<option>&#xf045;</option>

<option>&#xf046;</option>

</select></br>

Number of prints: <br><input type = "text" id = "numberField" size = "10"><br>

<input type = "button" onclick ="show()" value = "Show">

<script type="text/javascript">

function show(){

var start = 0;

var num = document.getElementById("numberField");

var nump = parseInt(num.value);

var x = document.getElementById("myEmoji").value;

while(start < nump){

document.writeln(x);

start++;

}

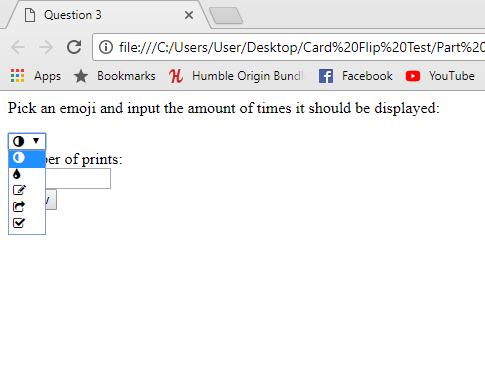
}

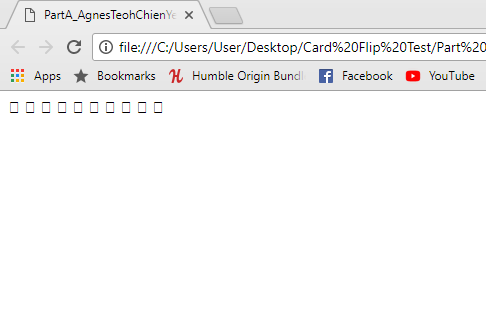
</script>

</body>

</html>

Output / Screenshots:





**Question 4:**

Source Code:

<!DOCTYPE>

<html>

<head>

<title>Question 4</title>

<p>Drag the slider for the Min Number and the Max Number. Click Print Prime Number after: </p>

</head>

<body>

<div id="slidecontainer">

Min Number<br><input type="range" min="1" max="1000" value="500" class="slider" id="minRange"><br>

<p>Value : <span id = "minValue"></span></p><br>

</div>

<div id="slidecontainer">

Max Number<br><input type="range" min="1" max="1000" value="500" class="slider" id="maxRange"><br>

<p>Value : <span id = "maxValue"></span></p><br>

</div>

<script>

var min = document.getElementById("minRange");

var minOutput = document.getElementById("minValue");

minOutput.innerHTML = min.value;

var max = document.getElementById("maxRange");

var maxOutput = document.getElementById("maxValue");

maxOutput.innerHTML = max.value;

min.oninput = function() {

minOutput.innerHTML = this.value;

}

max.oninput = function() {

maxOutput.innerHTML = this.value;

}

function getPrime(){

var mins = document.getElementById("minRange").value;

var maxs = document.getElementById("maxRange").value;

while(mins<maxs){

var i = 2;

var answer = 0;

do{

if (mins % i != 0){

answer = 1;}

else{

answer = 0;}

i++;

}while (answer == 1 && i<mins);

if(answer == 1){

document.write(mins + " ");}

mins++;

}

}

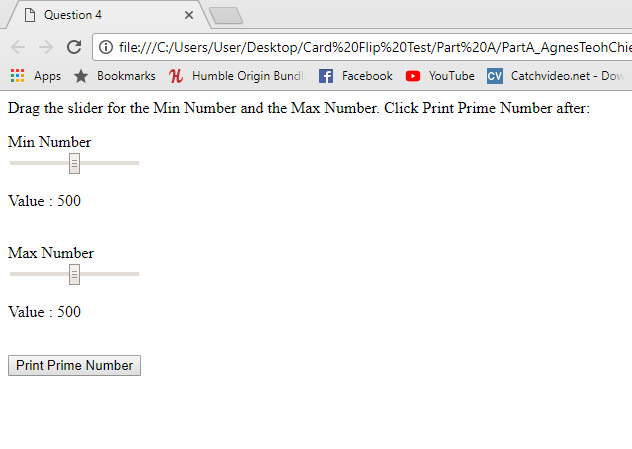
</script>

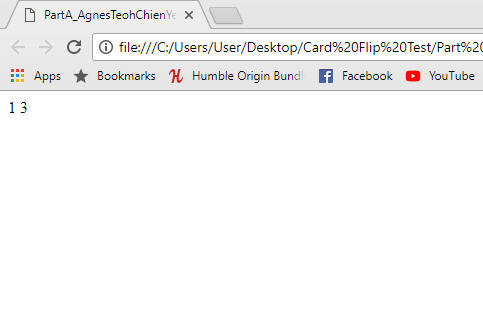
<input type = "button" onclick = "getPrime()" value = "Print Prime Number"><br>

</body>

</html>

Output / Screenshots:





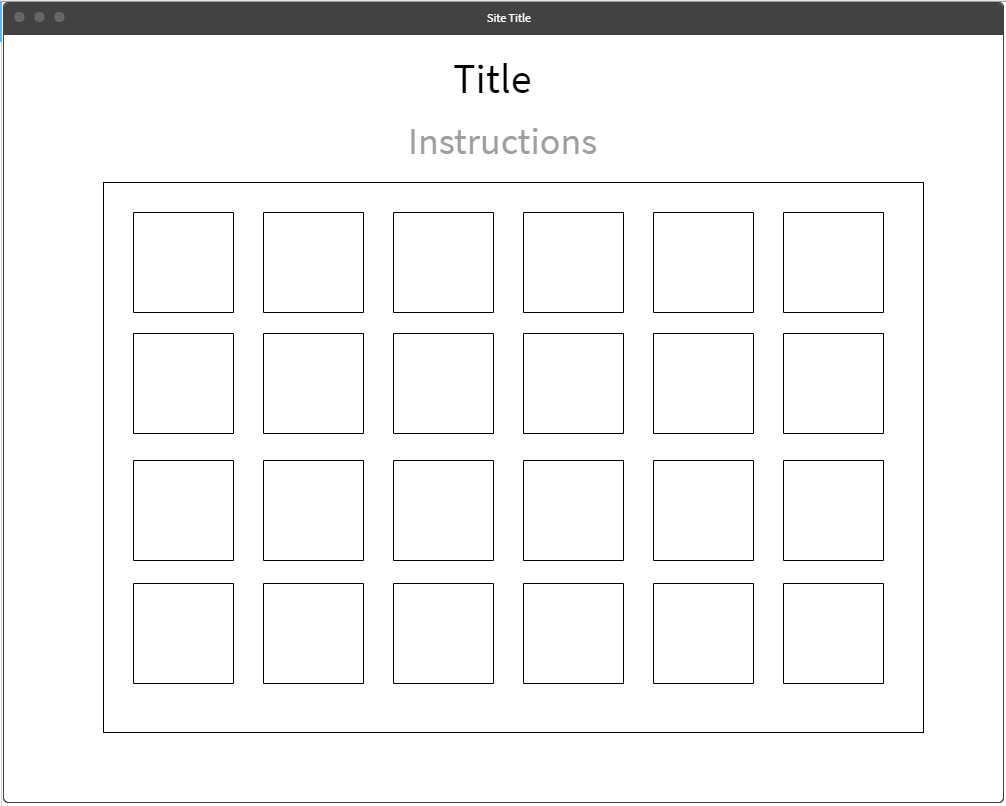
Part B:

1. Game Description

This game is a Memory Card game which allows players to flip the cards over to reveal the alphabet beneath, match them to the coresponding cards and finish the game. There will be 12 sets of letters and 24 cards with an unlimited amount of time and flips. After successfully matching all cards, the user can then click ok from the alert prompt to reset the game.

The goal of the game is to improve a person’s memory by making them remember the cards they have flipped and matching it together. This will urge them to memorize the location of cards in order to match it.

1. Layout Structure / Interface Design



1. Source Code

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>Memory Card Game</title>**

**<center>**

**<font size = "+5">Memory Card Game</font>**

**<p><font size = "+2" color = "darkgray">Click the tiles to see what alphabet they uncover and try to find the matching alphabet underneath the other tiles.</font></p>**

**<p><font size = "+2" color = "darkgray">Uncover two matching alphabets at once to eliminate them from the game.</font></p>**

**</center>**

**<style>**

**div#memory\_board{**

**background:#CCC;**

**border:#999 1px solid;**

**width:800px;**

**height:540px;**

**padding:24px;**

**margin:0px auto;**

**}**

**div#memory\_board > div{**

**background: url(tile\_bg.jpg) no-repeat;**

**border:#000 1px solid;**

**width:71px;**

**height:71px;**

**float:left;**

**margin:10px;**

**padding:20px;**

**font-size:64px;**

**cursor:pointer;**

**text-align:center;**

**}**

**</style>**

**</head>**

**<body>**

**<script>**

**var memory\_array = ['A','A','B','B','C','C','D','D','E','E','F','F','G','G','H','H','I','I','J','J','K','K','L','L'];**

**var memory\_values = [];**

**var memory\_tile\_ids = [];**

**var tiles\_flipped = 0;**

**Array.prototype.memory\_tile\_shuffle = function(){**

**var i = this.length, j, temp;**

**while(--i > 0){**

**j = Math.floor(Math.random() \* (i+1));**

**temp = this[j];**

**this[j] = this[i];**

**this[i] = temp;**

**}**

**}**

**function newBoard(){**

**tiles\_flipped = 0;**

**var output = '';**

**memory\_array.memory\_tile\_shuffle();**

**for(var i = 0; i < memory\_array.length; i++){**

**output += '<div id="tile\_'+i+'" onclick="memoryFlipTile(this,\''+memory\_array[i]+'\')"></div>';**

**}**

**document.getElementById('memory\_board').innerHTML = output;**

**}**

**function memoryFlipTile(tile,val){**

**if(tile.innerHTML == "" && memory\_values.length < 2){**

**tile.style.background = '#FFF';**

**tile.innerHTML = val;**

**if(memory\_values.length == 0){**

**memory\_values.push(val);**

**memory\_tile\_ids.push(tile.id);**

**} else if(memory\_values.length == 1){**

**memory\_values.push(val);**

**memory\_tile\_ids.push(tile.id);**

**if(memory\_values[0] == memory\_values[1]){**

**tiles\_flipped += 2;**

**// Clear both arrays**

**memory\_values = [];**

**memory\_tile\_ids = [];**

**// Check to see if the whole board is cleared**

**if(tiles\_flipped == memory\_array.length){**

**alert("Congratulations! Click ok to try again.");**

**document.getElementById('memory\_board').innerHTML = "";**

**newBoard();**

**}**

**} else {**

**function flip2Back(){**

**// Flip the 2 tiles back over**

**var tile\_1 = document.getElementById(memory\_tile\_ids[0]);**

**var tile\_2 = document.getElementById(memory\_tile\_ids[1]);**

**tile\_1.style.background = 'no-repeat';**

**tile\_1.innerHTML = "";**

**tile\_2.style.background = 'no-repeat';**

**tile\_2.innerHTML = "";**

**// Clear both arrays**

**memory\_values = [];**

**memory\_tile\_ids = [];**

**}**

**setTimeout(flip2Back, 700);**

**}**

**}**

**}**

**}**

**</script>**

**<div id="memory\_board"></div>**

**<script>newBoard();</script>**

**</body>**

**</html>**

1. Program Output / Screenshots

