COM S/SE 319 : Software Construction and User Interfaces

Spring 2019

HW 3

[Total Points: 50]

Assignment Due: Sunday, February 24, 2019, 11:59 PM

[N.B.:5% penalty per day up to a maximum of 7 days after **February 24, 2019**]

This assignment is focused on node.js and Event Handling

Task 1: Event Handling (20 points)

Write a Javascript and HTML code (named **snake.html** and **snake.js**) to implement the functionality shown in '**Task1Output.mp4**' included in the zip file. Some example codes are given as well.

Note:

1. **The line you create can go over any previous paths. [5 points]**
2. **The line will bend left when the left button is clicked. [5 points]**
3. **The line will bend right when the right button is clicked. [5 points]**
4. **The line should stop if it touches any boundary. [5 points]**

Hints:

1. **Use HTML5 Canvas (see http://www.w3schools.com/graphics/canvas_intro.asp)**
2. **Make sure to use a timer (see example below) to update the canvas (so that the snake keeps moving). A Timer has two main functionalities that can be used in the project.**
    a. **The *setInterval(function, delay)* schedules the "code" after every "delay" microseconds.**
    b. **The *clearInterval* removes the timer**

Here is an example of timer code. This will countdown from 100 until you press stop!

```
<html>
<body>


<h1><p id="header">COUNTDOWN IN SECONDS</p>
<p id="here"></p> </h1>
<input type="button" onclick="clearInterval(timer)" value="STOP COUNTDOWN
">


<script>
  var i = 100;
  var timer = setInterval(function() {
                  document.getElementById("here").innerHTML = i--;
                }, 1000)
</script>

</body>
</html>
```

Task 2: (25 points)

Objectives:

Learn to use node.js programming.

Warm-up:

NOTE 1: Play with the given "**example.js**". Open using a text editor of your choice and modify to learn how the different instructions work.

Task:

**It will be a console based application*

Your assignment is to **create a program in *node.js* named "hw3.js" that can do the following operations**. You can start with the given warm-up example "***example.js***" and follow lab activity 3. You need to install **'readline-sync'** like here.

1. **Take four integer numbers as input** from the console using 'readline-sync' like given example code. So, the user should be given a prompt for entering all 4 numbers one by one then need to press enter for getting the output. **(5 points)**
2. **Calculate the factorial of the first number**. In the console, the factorial of that first given number should be shown as an output. **(5 points)**
3. **Calculate the sum of all the digits of the second number**. For example, if we have the number 1234, the program will calculate 1+2+3+4 which is equal to 10. **(5 points)**
4. **For the third number given as an input, show the reversed number as an output.** For instance, if we give 12345 as input, it will show 54321 as output **(5 points)**
5. **For the fourth number given as an input, check whether that number is a Palindrome or not** and show the output as ***True*** in case it is a palindrome and ***False*** if it is not a palindrome. For example, if we give "12345" as an input it will return false but if we give 12321 as input, it will return true. **(5 points)**

   Sample Input and Output:

```
C:\Users\shibbir\Documents\TA\NodeJSPortable\Data>node hw3.js
1st Number: 5
2nd Number: 1234
3rd Number: 1234567
4th Number: 12321
Factorial of the 1st number is = 120
The Sum of all the digits of the 2nd number = 10
The Reverse of the 3rd number is = 7654321
Is the 4th Number a Palindrome(True/False)? = true
```

What to Submit:

Make sure your solutions work on Chrome as TAs will use it to grade the assignment.

Submit via Canvas a **compressed file (.zip)** containing the following:

- ***hw3.js***, for **Task 2** and ***snake.html*** , ***snake.js*** for **Task 1**.**[Task 1+Task 2 = 20+25 = 45 Points]**
- README file explaining how to compile and run your program & a **Report** (.docx or .pdf) describing your solution approach and screenshots of every required output. **[5 points]**.