FRESNO STATE
Discovery. Diversity. Distinction.

Lab 4: Web Programming - CSci130

Department of Computer Science, College of Science and Mathematics, Fresno State

Goals: The syntax of JavaScript, functions with JavaScript.

Exercise 0. Basic syntax (20min)

Create a page that shows a message “It is my first example with JavaScript” by using the **alert** function, then prompt the user if he wants to consider a circle, rectangle, or square depending on the choice, you will **prompt** again the user to get the dimensions of the object:

- Radius for the circle
- Height and width for the rectangle
- Sizes of the square.

Then, you will display through the alert function the **perimeter** and **area** of the selected element.

Use functions to compute areas and perimeters.

Exercise 1. Concatenations (15min)

Evaluate each expression (alert) and give the type (typeof) of each output:

1. "" + 1 + 0
2. "" - 1 + 0
3. true + false
4. 6 / "3"
5. "2" * "3"
6. 4 + 5 + "px"
7. "$" + 4 + 5
8. "4" - 2
9. "4px" - 2
10. 7 / 0
11. " -9\n" + 5
12. " -9\n" - 5
13. null + 1
14. undefined + 1

Comment the differences obtained for expressions 11 and 12.

Exercise 2. Conditions (5min)

Rewrite the following if then else expression with the ? syntax. Test both versions of the code with a prompt and alert in separate webpages.

```
let msg;
if (login == 'student') {
  msg = 'Hi';
} else if (login == 'faculty') {
  msg = 'Greetings';
} else if (login == '') {
  msg = 'No login';
} else {
  msg = '';
}
```

Exercise 3. Basic array functions (30min)

We consider an array that can be defined as follows:

var x = new Array(12); // 12 being the size of the array

You can access the elements like in C ++: x[0] for the first position

- Create a function to get the average of the values in the array
- Create a function to get the standard deviation of the values in the array
- Create a function to display the average and standard deviation in a message box.

Exercise 4. Prime numbers (40min)

Create a webpage that will display the *n* first prime number, where *n* is given by the user in a textbox next an appropriate label. There is an ok button to generate the table. Each prime number will be placed into a cell in a table, with one number per row. The first column is the index of the prime number, the second column is the value of the prime number.

Remark: each time you click on the ok button, it should remove the previously written elements.

Example:

For n=6, you need to create such a table:

1	2
2	3
3	5
4	7
5	11
6	13