Announcements
Put the hardcopy of Project 1b in the box
* Please keep up on reading ...

Programming Basics
When it comes to being precise about an algorithm, a programming language is better than English

The Plan
We will learn JavaScript over the next few lectures
* JavaScript is used with HTML for Web pages
* JavaScript is a contemporary programming language -- we will learn only its basics
* You will program in NotePad and run your program with your browser

*Programming is understandable, but most people don’t understand it the first time... so don’t give up by other examples & reread the textbook*

Using JavaScript in HTML
JavaScript must be surrounded by <script> tags ... build a test page

```html
<body>
  <script language="JavaScript">
    var textout, numout;
    textout = "two plus two equals ";
    numout = 2 + 2;
    alert(textout + numout);
  </script>
</body>
```

First JS Program
Write and run a program to figure 2+2

```html
<body>
  <script language="JavaScript">
    var textout, numout;
    textout = "two plus two equals ";
    numout = 2 + 2;
    alert(textout + numout);
  </script>
</body>
```

Names In Programming
In normal language, names, and the things they name -- their values -- usually cannot be separated
* In programming most names change values ... a consequence of finite specification
* Titles (US_Open_Champ), Offices (Mayor), Roles (Juliet), etc. are familiar examples of names that change values
* Rules, Processes and Directions exploit the variable value: "Juliet moves to the window"
Variables

- Names in programming are identifiers
- The things they name are their values

The package -- identifier & value -- is a variable, implying a possible change

- Identifiers have a specific structure in every programming language
- JS: letters, digits, _ start with letter, case sens.

To declare variables is to state what variables will be used

- Required … put declarations first in program
- Use the word: var
- Follow with a list of variables separated by ,
- Terminate all statements with a semicolon ;

\[ \text{var } x, \text{ input1, input2, rate;} \]

\[ \text{var interestRate } = 4, \pi = 3.14159; \]

Values

Programming languages allow several types of values: numeric, strings of letters, Boolean

- numbers: 1 0 -433 6.022e+23 .01
- not numbers: 1,000 10% 5% 7k2
- strings: "abc" 'efg' " "B&B's" ""
- not strings: ' ' '<tab>" " "
- Boolean: true false
- not Boolean: T F yes no

Assignment

The universal form of assignment:

\[ \text{<variable>} \text{<assignment symbol>} \text{<expression>} \]

For example …

\[ \text{day = hours/24;} \]

- value of the variable on the left is changed to have the new value of expression on right
- read “=” as “is assigned” “becomes” “gets”
- right-to-left value flow

Expressions

Expressions are like “formulas” saying how to manipulate existing values to compute new values, e.g. hours/24

- Operators: + - * / % produce numbers
- Operators: < <= == != > >= on numbers (or strings for === and !==) produce Booleans
- Operators: & | ! on Booleans produce Booleans
- Grouping by parentheses is OK and smart

\[ \text{seconds } = \left( \left( \text{days} \times 24 + \text{hours} \right) \times 60 + \text{min} \right) \times 60 \]

Overloading Plus

The + can be used to add numbers or join strings (concatenate)

- Operators: + - * / % produce numbers
- Operators: < <= == != > >= on numbers (or strings for === and !==) produce Booleans
- Operators: & | ! on Booleans produce Booleans
- Grouping by parentheses is OK and smart

\[ \text{seconds } = \left( \left( \text{days} \times 24 + \text{hours} \right) \times 60 + \text{min} \right) \times 60 \]
First JS Program, Revisited

Write and run a program to figure 2+2

```javascript
var textout, numout;
var textout = "two plus two equals ";
numout = 2 + 2;
alert(textout + numout);
</script>
</head>
</body>
</html>
```

Conditional

Conditionals test if an expression is true or not

- General form ...

  ```javascript
  if (<Boolean expression>)
  <Then statement>;
  ```

- For example

  ```javascript
  if (day == "Friday")
  evening_plan = "party";
  ```

If-Then-Else

Branch both ways with If-Then-Else

```javascript
if (<Boolean expression>)
<Then statement>;
else
<Else Statement>;
```

- Example ...

  ```javascript
  if ((year%4) == 0) {
  leapYear = true;
  febDays = febDays+1;
  }
else
  leapYear = false;
  ```

Summary

Programming is the exact specification of an algorithm

JavaScript is typical … with many rules

- Learning strategy

  - Do the reading first
  - Practicing is better than memorizing for learning the rules
  - Use the program-save-reload-check plan
  - Precision is your best friend