Form based input

# Lesson Notes

## What about input!

* So far the behavior of our programs could not be changed by the user
  + Most applications have some way for the user to interact with them
* One style of user input is “form input”
* Similar to filling out a paper form (like a quiz)
  + Write or make a mark in each “field”
    - Type in text or select an option
  + Hand in the form
    - Click OK
  + Show hand-drawn example of a form

## Form input

* Forms appear in many user interfaces
  + A simple one appears in web search
* Common concepts in form-based input:
  + OK button (means “I’m done”)
    - Can have any name (OK, Apply, Save, Search, …)
    - You can press [Enter] as well
  + Input fields (lots of types of these)
  + Focus (where the user is)
  + Tab order (when you press tab, where does the focus go?)
  + Field validation (did I type a valid value?)

## Using HTML forms

* A form element
  + Container for all the fields
* Each field is an element in javascript
  + Fields are either <input> or <textarea> elements
  + Find the object using document.getElementByID()
  + Each field has a value property
  + Each field has a name property
* A <label> binds the description of the field with the field
  + Clicking on the label will set focus to the element
* Submitting the form:
  + Submitting means: “I’m done”
    - Press a submit button
    - Press [enter]
  + Form action attribute controls what happens when form is submitted
  + We can also use it to call a javascript function:
    - action=”javascript:myFunction();”
* Many kinds of inputs
  + Link to different kinds of inputs (page)

## A simple example

* Show example form:
  + Logic to put a variable into the form when page loads
  + Logic to read form values when form is submitted

## Converting a number to a string

* Convert a number to a string using the toString property of a string.
  + var x = 1;
  + var y = x.toString();

## Converting a string to a number

* Convert a string to a number using the global parseInt function:
  + var x = ”1”;
  + var y = parseInt(x, 10);
* The parseInt function supports multiple number bases, so we pass 10 for decimal.

## Exercise

* Add a form to one of your programs:
  + Start with an existing program that does not have user input
  + Add a form with fields that control its behavior
  + The program should take some action when you submit the form
  + Test the tab order to make sure it makes sense visually
  + Test your labels to make sure they set focus to the field when you click on them