HTML FORMS

"Good news has come to the citizens of Lothal. The Outer Rim planet has been granted the coveted opportunity to participate in the new Imperial Registration Program." —Alton Kastle

HTML FORMS: HOW, WHEN, AND WHY

HTML forms are the primary way of receiving information from the browser to a server. A form consists html

elements just like other parts of html, but unlike other html elements form and form controls provide extra and specific functionality.

```html
<form action="http://url_for_form_submit">
    <h3>Galactic Registry</h3>
    <fieldset>
        <legend>Citizen Information</legend>
        <div>
            <label for="first-name">First Name</label>

            <input name="first-name" id="first-name">
        </div>
        <div>
            <label for="last-name">Last Name</label>
            <input name="last-name" id="last-name" type="t
        </div>
        <div>
```

The form element can contain any type of element, but specific ones called control elements will actually respond with data when the form is submitted.

The form will 'submit' to a url that is the value of the action attribute. If there is no action it will submit to itself.

pressing a submit button or pressing enter will submit the form

Fieldset is a way to semantically separate sections in a form. It can also be helpful to

a screen reader for the visually impaired.

The legend tag inside of the fieldset is a great way to label the section.

The most basic controller is the input. The input controller has several different 'types'

that are specified in the type attribute.

The default attribute is just a text input.

All control items can use the name and the id attribute (all elements can actually accept the id attribute).

The name element will be the name of the field that is submitted. So it needs to be unique.

The label doesn't do anything when by itself, but when the for attribute is applied it will give the label to the element with the id in the for attribute.

If you click on a label that the for attribute and there is an control element that has the same id. It will make that control element active.

The text input and textarea have an attribute called "placeholder" It is test written inside the field.

You can also have value already placed in the text field by using the value attribute. This can be overwritten.

Radio elements are unique in the fact that they do not "toggle" by clicking or un-clicking. Instead the need multiple elements with the same name to be able to be

toggled.

If the radio elements do not have a value attribute, then they will always submit " [name attribute]=on"

Checkboxes can be toggled by clicking the box.

Javascript allows for more advanced usages of grouping checkboxes.

selects allow for a single option to be selected

A preselected value is allowed by using the boolean attribute 'selected' in the option that you want selected.

The textarea element allows for multiple lines.

Control Elements can be disabled with the disable bool attribute.