The following gives some sample code from Applet web pages.
It is intended to give examples so that you can understand enough to copy paste and modify.

**********Applet code Name added to generated code

```
<applet code="geogebra.GeoGebraApplet" name="ggbApplet"
            codebase="./" archive="geogebra.jar" height="500" width="730">
      <param name="filename" value="FamilyOfFunctions_worksheet.ggb">
      <param name="framePossible" value="true">
      <param name="showResetIcon" value="true">
      <param name="enableRightClick" value="false">
      <param name="showMenuBar" value="false">
      <param name="showToolBar" value="true">
      <param name="showToolBarHelp" value="true">
      <param name="showAlgebraInput" value="false">
Sorry, the GeoGebra Applet could not be started. Please make sure that Java 1.4.2 (or later) is
installed and activated. (<a href="http://java.sun.com/getjava">click here to install Java
now</a>)</applet>
```

**********Comments

The block of code above is from the page for the families of functions applet.

1) In the first line, which begins with "applet", the string

```
        name="ggbApplet"
```

was added to the code generated by GeoGebra. This names the applet so that I can communicate with the applet using javascript.

2) In the second line

```
        codebase="./"
```

indicates where the geogebra.jar file is to be found. In this case, the file is to be found in the same directory as the html file.

3) The value of the filename parameter indicates the Geogebra file to be used.

4) The remaining lines show options that can be set for the web page. These can be set with advanced features when the web page is exported. They can also be changed by changing the values true and false.

From that Family of Functions Applet, we want to focus on two lines of the web page. There is a line to either get the function f from the applet or to send a new definition of f to the applet. There is also a line that sets the viewing window.

We will be looking at Javascript for two forms (FunctionForm and ViewForm) and three functions. (setFunction, getFunction, and setView)

```
************************************
                        Function sending Info to Applet
<script type="text/javascript">
function setFunction(objName) {
      var applet = document.ggbApplet;
      var f= document.functionForm.FunctionField.value;
      applet.evalCommand(f);
  }

                          Function getting info from applet
function getFunction(objName) {
      var applet = document.ggbApplet;
      var f= applet.getDefinitionString("f");
      document.functionForm.FunctionField.value = f
  }

                           Function to send view window info
function setView(objName) {
      var applet = document.ggbApplet;
      var LowX = document.ViewForm.LowXField.value;
      var HighX = document.ViewForm.HighXField.value;
      var LowY = document.ViewForm.LowYField.value;
      var HighY = document.ViewForm.HighYField.value;
      applet.setCoordSystem(LowX, HighX, LowY, HighY);
}
</script>

                       Form to show/get function
 <form name="functionForm"  onsubmit="setFunction('T'); return false;">
 <input value="get function" onclick="getFunction('T');" type="button"> 
 <input name="FunctionField" size="40"value="f(x) = a*x^2+b*x+c" type="text"> 
 <input value="set function" onclick="setFunction('T');" type="button"> </form>

                       Form to collect view window info
<form name="ViewForm" onsubmit="setView('T');  return false;">
Min X = <input name="LowXField" size="5" onchange="setView('T');" value="-10" type="text">
Max X = <input name="HighXField" size="5" onchange="setView('T');" value="10" type="text">
Min Y = <input name="LowYField" size="5" onchange="setView('T');" value="-10" type="text">
Max Y = <input name="HighYField" size="5" onchange="setView('T');" value="10" type="text">
<input value="Set Window" onclick="setView('T');" type="button">
</form>

**********Comments
```

1) In javascript, document refers to the current web page. As noted above, the applet is named ggbApplet and the forms are functionForm and viewForm.
2) The 3 functions provided above can be separated into separate scripts.

From that Family of Functions Applet, we now want to focus on line of the web page that provided a list of examples. Note that each menu choice has us executing several commands.

Function to load selected example

```
<script type="text/javascript">
function ExampleLoader(objName) {
      var applet = document.ggbApplet;
      var n= document.ExampleForm.ExampleSelect.value;
      if (n==0) {
            var f="f(x)=a*x^2+b*x+c";
            applet.setCoordSystem(-10, 10, -8, 8);
            applet.evalCommand(f);
            document.functionForm.FunctionField.value = f;}
      else if (n==1) {
            var f="f(x)=a*(x-b)*(x-c)";
            applet.setCoordSystem(-10, 10, -8, 8);
            applet.evalCommand(f);
            document.functionForm.FunctionField.value = f;}
      else if (n==2) {
            var f="f(x)=a*(x-b)^2+c";
            applet.setCoordSystem(-10, 10, -8, 8);
            applet.evalCommand(f);
            document.functionForm.FunctionField.value = f;}
      else if (n==3) {
            var f="f(x)=a*sin(b*(x-c))";
            applet.setCoordSystem(-7, 7, -4, 4);
            applet.evalCommand(f);
            document.functionForm.FunctionField.value = f;}
      else if (n==4) {
            var f="f(x)=a*sin(x)+b*cos(x)";
            applet.setCoordSystem(-7, 7, -4, 4);
            applet.evalCommand(f);
            document.functionForm.FunctionField.value = f;}
      else if (n==5) {
            var f="f(x)=a*ln(b*x)+c";
            applet.setCoordSystem(-1, 10, -8, 8);
            applet.evalCommand(f);
            document.functionForm.FunctionField.value = f;}
      else if (n==6) {
            var f="f(x)=a*b^x+c";
            applet.setCoordSystem(-10, 10, -8, 8);
            applet.evalCommand(f);
            document.functionForm.FunctionField.value = f;}
  }
</script>
```

Form with drop down menu of examples

```
<form name="ExampleForm">
      <select name="ExampleSelect" size="1">
            <option value ="">Example Families</option>
            <option value =0>General Quadratic: a*x^2+b*x+c</option>
            <option value =1>Factored Quadratic: a*(x-b)*(x-c)</option>
            <option value =2>Vertex Parabola: a*(x-b)^2+c</option>
            <option value =3>Sin Curve with Phases: a*sin(b*(x-c))</option>
            <option value =4>Linear Combination of sin and cos: a*sin(x)+b*cos(x)</option>
            <option value =5>Logrithmic Curve: a*ln(b*x)+c</option>
            <option value =6>Exponential Curve: a*b^x+c</option>
      </select>
      <input value="Load Example" onclick="ExampleLoader(this)" type="button">
</form>
```

This last page of code comes from the Taylor Polynomial Approximation applet page.
It is noteworthy since it gathers a lot of information together, both for input and output.

```
*********************************
<script type="text/javascript">
                  Setting functions and parameters
function setFunction(objName) {
  var applet = document.ggbApplet;
  var f= document.functionForm.setFunctionField.value;
  applet.evalCommand("f(x)="+f);
  var n= document.functionForm.setTaylorDegree.value;
  applet.evalCommand("n="+n);
  var x0= document.functionForm.setx0.value;
  applet.evalCommand("x_0="+x0);
 var x1= document.functionForm.setx1.value;
  applet.evalCommand("x_1="+x1);
  }

                gathering lots of output together
function setOutput(objName) {
  var applet = document.ggbApplet;
  var TaylorPoly= applet.getValueString("TaylorPoly");
  var TaylorDegree = applet.getValue("n")
  var x0 = applet.getValue("x_0")
  var x1 = applet.getValue("x_1")
  var T1y = applet.getYcoord("T_1")
  var P1y = applet.getYcoord("P_1")
  var E1y = applet.getYcoord("E_1")
  document.outputForm.TextAreaOutput.value =
  "We are looking at the Taylor polynomial of degree " + TaylorDegree +
  "\ncentered at x_0 = " + x0 + "\n" +
  "At x_1 = " + x1 + " the function has value " + P1y +
  "\nwhile the polynomial approximation is " + T1y +
  "\ngiving an error of " + E1y + "\n" +
  TaylorPoly
  }
</script>

                  Function data form
<form name="functionForm" onsubmit="setFunction('T');  return false;">
<input value="set values" onclick="setFunction('T');" type="button"> 
f(x) = <input name="setFunctionField" onchange="setFunction('T');" size="30"
 value="3*sin(x/2)+5*cos(x)" type="text"><br>
Degree of Approximation = <input name="setTaylorDegree" size="5"
 onchange="setFunction('T');" value="4" type="text">
, x_0 = <input name="setx0" size="10" onchange="setFunction('T');" value="1"
 type="text">
, x_1 = <input name="setx1" size="10" onchange="setFunction('T');" value="2" type="text">
</form>

                   Test area output form
<form name="outputForm" onsubmit="setOutput('T');  return false;">
<input value="Get output" onclick="setOutput('T');" type="button"><br>
<textarea name="TextAreaOutput" rows="10" cols="90" value=" "
onchange="setOutput('T');"></textarea>
</form>

*****************************Comments
1) The output to the textarea is made up of a collection of things concatenated together with
plus signs.  The character ”\n” gives a new line.
```