Corrections Special School District
Career & Technical Education Center
Term: **April 3 to June 16, 2023**

Please provide the following information.

Room Number: [   ]

Instructor Name: [Choose an instructor ▾]

Time Slot:
- ◯ 7:10 AM to 8:30 AM
- ◯ 8:40 AM to 10:00 AM
- ◯ 10:10 AM to 11:30 AM
- ◯ 11:40 AM to 1:00 PM
- ◯ 1:10 PM to 2:30 PM

Course Title: [   ]

[Start Over]   [Submit Class Information]

Corrections Special School District
Career & Technical Education Center
Term: **April 3 to June 16, 2023**

Thank you. You entered the following information.

Room Number: **10**

Instructor Name: **Heines**

Course Title: **Website Development I**

Time Slot: **8:40 AM to 10:00 AM**

[Return to the Form Page]

1

Line wrap ☐

```
 1  <!DOCTYPE html>
 2  <html>
 3  <head>
 4    <meta charset="utf-8">
 5    <title>Basic Form</title>
 6
 7  <!--
 8    File:  http://cssdweb.edu/StudentResources/CodeSamples/BasicForm.html
 9    Copyright (c) 2022-2023 by Jesse Heines.  All rights reserved.  May be freely
10      copied or excerpted for educational purposes with credit to the author.
11    updated by JMH on June 12, 2022 at 10:39 PM
12    updated by JMH on June 6, 2023 at 10:58 AM
13  -->
14
15    <style>
16      h2#title {                  /* main page title */
17        color: blue ;
18        margin-bottom: 0.2em ;
19      }
20      h3#semester {               /* page subtitle (semester dates) */
21        color: blue ;
22        margin-top: 0 ;
23      }
24      .nobot {                    /* eliminate the bottom margin */
25        margin-bottom: 0 ;
26      }
27      .notop {                    /* eliminate the top margin */
28        margin-top: 0 ;
29      }
30    </style>
31  </head>
32
33  <body>
34    <!-- display the page title and term dates -->
35    <h2 id="title">Corrections Special School District<br>
36      Career &amp; Technical Education Center</h2>
37    <h3 id="semester"><em>Term:</em>  April 3 to June 16, 2023</h3>
38
39    <!-- instructions to the user -->
40    <p><br>
41      <em>Please provide the following information.</em></p>
42
43    <!-- form to create a new table entry -->
44    <form method="get" action="BasicForm_Handler_v2.html">
45      <!--
46        The value of the name attribute in each of the following form elements will be the
47        key passed when the form in submitted.  This key is what is used to identify each
48        data item.
49            Note that unlike most other attributes, the required attribute does not have a
50        corresponding value.  It is used simply on its own.
51      -->
52
53      <!-- spinner form element -->
54      <p>Room Number: 
55      <input type="number" name="room" min="1" max="10" step="1"
56            value="" required></p>
57      <!--
58        Input type number is a spinner.  A spinner has minimum, maximum, and step values.
59        Setting the initial value to blank ("") forces the user to enter a value within
60        the specified range.
61      -->
62
63      <!-- select form element (or dropdown list) -->
64      <p>Instructor Name: 
65        <select name="instructor" required>
66          <option value="" selected disabled hidden>Choose an instructor</option>
67          <option value="Burzycki">Burzycki</option>
68          <option value="Heines">Heines</option>
69          <option value="Latorella">Latorella</option>
70          <option value="Lees">Lees</option>
71          <option value="Mosher">Mosher</option>
72          <option value="O'Donnell">O'Donnell</option>
73          <option value="Piper">Piper</option>
74          <option value="Spires">Spires</option>
75        </select>
76      <!--
77        The first option above prompts the user with instructions.  However, this option
        cannot be selected.  The user is forced to select one of the other options.
            It is also possible to specify how many of the options should initially be
        visible, but this can sometimes confuse the user into thinking that those are the
        only options available.
```

2

```
82        -->
83
84        <!-- radio buttons group -->
85        <p class="nobot">Time Slot:</p>
86        <blockquote class="notop">
87          <input type="radio" name="timeslot" value="A" required>
88            <label for="A">7:10 AM to 8:30 AM</label><br>
89          <input type="radio" name="timeslot" value="B">
90            <label for="B">8:40 AM to 10:00 AM</label><br>
91          <input type="radio" name="timeslot" value="C">
92              <label for="C">10:10 AM to 11:30 AM</label><br>
93          <input type="radio" name="timeslot" value="D">
94            <label for="D">11:40 AM to 1:00 PM</label><br>
95          <input type="radio" name="timeslot" value="E">
96            <label for="E">1:10 PM to 2:30 PM</label>
97        </blockquote>
98        <!--
99          Note that here we display the actual times to the user via the label tag.  However,
100         the input tag's value attribute specifies only a single letter.  Therefore, it is
101         the letter that is passed to the handler page, not the full time string.
102        -->
103
104       <!-- text input form element -->
105       <p>Course Title: 
106         <input type="text" name="course" size="40" required></p>
107       <!--
108         This element allows the user to enter a single line of text.
109             Note the size attribute.  This value controls the width of the form, not the
110         number of characters that the user can enter.  To limit then number of characters,
111         add a maxlength attribute to the above code.
112        -->
113
114       <!-- the action buttons -->
115       <br>
116       <input type="Reset" value="Start Over">    
117       <input type="Submit" value="Submit Class Information">
118       <!--
119         Input type reset returns the form to its initial state.  Note that in these two
120         cases, the value attribute specifies the text to be displayed on the buttons.
121             Input type submit checks to make sure that all the required fields have legal
122         values, packages up those values into a query string, and then passes that query
123         string to the page specified in the form tag's action attribute.
124             To do more sophisticated error checking to ensure that all data entered by the
125         user is valid, one can write JavaScript code that will get executed before the data
126         is passed to the handler page.
127        -->
128    </form>
129
130 </body>
131
132 </html>
133
```

3

Line wrap ☐

```
1   <!DOCTYPE html>
2   <html>
3   <head>
4     <meta charset="utf-8">
5     <title>Basic Form Handler</title>
6
7   <!--
8     File:  http://cssdweb.edu/StudentResources/CodeSamples/BasicForm_Handler.html
9     Copyright (c) 2022-2023 by Jesse Heines.  All rights reserved.  May be freely
10      copied or excerpted for educational purposes with credit to the author.
11    updated by JMH on June 11, 2022 at 5:29 PM
12    updated by JMH on June 30, 2022 at 9:24 PM
13    updated by JMH on June 6, 2023 at 10:58 AM
14  -->
15
16    <script>
17      // define an array of the time slots
18      // Note that the array subscripts can be strings, in this case single letters.
19      arrTimeSlots = new Array() ;
20      arrTimeSlots["A"] = "7:10 AM to 8:30 AM" ;
21      arrTimeSlots["B"] = "8:40 AM to 10:00 AM" ;
22      arrTimeSlots["C"] = "10:10 AM to 11:30 AM" ;
23      arrTimeSlots["D"] = "11:40 AM to 1:00 PM" ;
24      arrTimeSlots["E"] = "1:10 PM to 2:30 PM" ;
25    </script>
26
27    <style>
28      h2#title {                /* main page title */
29        color: blue;
30        margin-bottom: 0.2em;
31      }
32      h3#semester {             /* page subtitle (semester dates) */
33        color: blue;
34        margin-top: 0;
35      }
36      .entry{                   /* for entries entered by the user */
37        color: blue;
38        font-weight: bold;
39      }
40      .error {
41        color: red;             /* errors are often displayed in red text */
42      }
43    </style>
44
45    <script>
46      //// CHECK FOR AND PARSE THE SEARCH STRING
47      var ProcessSearchString = function() {
48        let strSearch = window.location.search ;
49        if ( strSearch == "" ) {
50          document.getElementById( "ResultMessage" ).innerHTML = "<br>" +
51            "<span class='error'>" +
52            "<em>You didn&rsquo;t enter any information.  " +
53            "Please click the </em><strong>Return to the Form Page</strong><em> " +
54            "button below to try again.</em></span>" ;
55        } else {
56          // This would be a nice place to use a table, but here I have used regular
57          // paragraphs for simplicity.
58          document.getElementById( "ResultMessage" ).innerHTML = "<br>" +
59            "<em>Thank you.  You entered the following information.</em>" ;
60
61          // Note that URLSearchParams is a special type of function called a constructor.
62          // It takes the query string as its argument and constructs an object from which we
63          // can subsequently extract the values entered by the user in the various form fields
64          // on the main form page.
65          var urlParams = new URLSearchParams( strSearch ) ;
66          var strRoomNumber = urlParams.get( "room" ) ;
67          var strInstructorName = urlParams.get( "instructor" ) ;
68          var strCourseTitle = urlParams.get( "course" ) ;
69          var strTimeSlot = arrTimeSlots[ urlParams.get( "timeslot" ) ] ;
70
71          // display the data entered by the user
72          document.getElementById( "RoomNumber" ).innerHTML =
73            "Room Number:  <span class='entry'>" + strRoomNumber + "</span>" ;
74          document.getElementById( "InstructorName" ).innerHTML =
75            "Instructor Name:  <span class='entry'>" + strInstructorName + "</span>" ;
76          document.getElementById( "CourseTitle" ).innerHTML =
77            "Course Title:  <span class='entry'>" + strCourseTitle + "</span>" ;
          document.getElementById( "TimeSlot" ).innerHTML =
            "Time Slot:  <span class='entry'>" + strTimeSlot + "</span>" ;
        }
```

4

```
 81        }
 82      </script>
 83  </head>
 84
 85  <body onload="ProcessSearchString();">
 86      <!-- display the page title and term dates -->
 87      <h2 id="title">Corrections Special School District<br>
 88        Career &amp; Technical Education Center</h2>
 89      <h3 id="semester"><em>Term:</em>  April 3 to June 16, 2023</h3>
 90
 91      <p id="ResultMessage"></p>
 92      <p id="RoomNumber"></p>
 93      <p id="InstructorName"></p>
 94      <p id="CourseTitle"></p>
 95      <p id="TimeSlot"></p>
 96
 97      <p><br>
 98        <button onclick="window.location.replace('BasicForm_v2.html');">
 99          Return to the Form Page</button></p>
100  </body>
101
102  </html>
103
```

5