Total max score: 30

For all multi choice questions in this exam: You may check as many answers you like, but each checked answer that is wrong will be punished with 1 point reduction for that question (no question can give negative points in total).

Which 2 of the following elements do not exist in HTML5?

- [ ] <li>
- [ ] <meta>
- [ ] <trow>
- [ ] <javascript>
- [ ] <a>
- [ ] <title>

Max score: 2

Which 2 of the following statements correctly explain what the src attribute refers to in HTML5?

- [ ] The src attribute refers to a file containing an image for the <body> element.
- [ ] The src attribute refers to a file (usually containing HTML code) for the <a> element.
- [ ] The src attribute refers to a file containing JavaScript code for the <script> element.
- [ ] The src attribute refers to a file containing an image for the <img> element.
- [ ] The src attribute refers to a file containing CSS code for the <link> element.

Max score: 2

Which one of the selectors listed below can be used to **only** select the <span> elements containing the numbers 1, 4 and 7 in the HTML code below?

```
<div class=”a”><span class=”a”>1</span> <span class=”b”>2</span></div>
<div id=”a”>   <span class=”a”>3</span> <span class=”b”>4</span></div>
<div>          <span class=”a”>5</span> <span class=”b”>6</span></div>
<div class=”a”><span id=”b”   >7</span> <span class=”b”>8</span></div>
```

- ○ div > span.b, span.a
- ○ .a.a, #a.b, #b
- ○ .a span.b, .a .b
- ○ .a .a, #a .b, .#b,
- ○ None.
- ○ .a (.a, #b), #a .b

Max score: 1

The id selector has higher specificity than the class selector. Given the code below, which one of the following statements is true?

```
<!DOCTYPE html>
<html>
  <head>
    <title>Written exams rule!</title>
    <style>
      .p{ color: blue; }
      #p{ background-color: red; }
    </style>
  </head>
  <body>
    <p id="p" class="p">Some text.</p>
  </body>
</html>
```

- Since the id selector has higher specificity, the background in the paragraph will be red, but the text will not be blue.
- The background in the paragraph will be red, and the text will be blue.
- It is invalid CSS code since two different rules are selecting the same element.
- The background in the paragraph will be red, but since the id selector has higher specificity, it will ”shine through” the text, so the text will be rendered as purple (a mix of being blue and red).
- Since the class selector has lower specificity, the text in the paragraph will be blue, but the background will not be red.

Max score: 1

When submitting the form below (without altering it), which one of the following HTTP requests will the browser send away to the server (we only show the headers relevant to the form; the browser will send many more headers in addition to these)?

```
<form method="GET" action="http://somesite.com/login">
  Username: <input name="username" value="Alice">
  Password: <input name="password" value="Abernathy">
  <input type="submit" value="Login">
</form>
```

-
  ```
  POST /login?username=Alice&password=Abernathy HTTP/1.1
  Host: somesite.com
  ```

-
  ```
  POST /login HTTP/1.1
  Host: somesite.com
  Content-Type: application-x-www-form-urlencoded
  Content-Length: 33

  username=Alice&password=*********
  ```

○

```
GET /login HTTP/1.1
Host: somesite.com
Content-Type: application-x-www-form-urlencoded
Content-Length: 33

username=Alice&password=Abernathy
```

○

```
GET /login HTTP/1.1
Host: somesite.com
Content-Type: application-x-www-form-urlencoded
Content-Length: 33

username=Alice&password=*********
```

○

```
GET /login?username=Alice&password=Abernathy HTTP/1.1
Host: somesite.com
```

○

```
POST /login HTTP/1.1
Host: somesite.com
Content-Type: application-x-www-form-urlencoded
Content-Length: 33

username=Alice&password=Abernathy
```

Max score: 1

Explain how the code below works and write which value the variable `sum` stores after the code has been executed.

```
var functions = []
for(var i=0; i<3; i++){
  functions.push(function(){ return i })
}

var sum = 0
for(var j=0; j<functions.length; j++){
  sum += functions[j]()
}
```

Max score: 2

When browsers started to support JavaScript it was popular to write JavaScript code in different event attriutes (such as `onclick`) in elements. With time we have started to use the method `addEventListener` in JavaScript instead. List all the benefits you can think of using this newer approach compared to the old one.

Max score: 4

Name three different methods you can use to retrieve HTML elements from the `document` variable.

Max score: 3

When is the `DOMContentLoaded` event fired?

Max score: 1

What is *the same-origin policy*?

Max score: 1

Servers that support JSONP typically allow clients to give a value for a parameter in the query string named `callback`. How is the value for this parameter used? Give an example of how you would use it in a request sent to a server and what the server would send back to you (you are not expected to write any JavaScript code sending a request to a server).

Max score: 1

Functions are objects in JavaScript, and they have both a property named `__proto__` and a property named `prototype`. Explain the difference between these two.

Max score: 2

Write the code needed for the following code to work as intended (make use of inheritance):

```
var lisa = new Human({name: "Lisa", age: 23})
var lisasName = lisa.getName()
```

Max score: 2

Implement the function count according to the comments below:

```
var one = count() // The first call should return 1.
var two = count() // The second call should return 2.
var three = count() // The third call should return 3.
// And so on...
```

Max score: 1

Given the code below, write the code which computes the id of the longest leg the horse has. Your code should still work 5 years from now when the horse has grown, and even if someone has given the horse a fifth leg. Call the function in the property `getLegs` as few times as possible.

```
var animal = {
  type: "Horse",
  getLegs: function(){
    return [{
      id: 0,
      length: 1341
    }, {
      id: 1,
      length: 1337
    }, {
      id: 2,
      length: 1339
    }, {
      id: 3,
      length: 1339
    }]
  }
}
```

Max score: 3

When the code below has been executed, which objects have been created? What do they look like? What is stored in the variable `sum`, and how come it's that value?

```
var x = {v: 1}
var a = Object.create(x)
var b = Object.create(x)
a.v = a.v + 1
var sum = x.v + a.v + b.v
```

Max score: 3