

## ANSWER KEY

### 1. CSS Coding

```
#menu p {
    color: white;
    background-color: black;
    float: left;
    padding-left: 20px;
    padding-right: 20px;
    margin-right: 3px;
}

#pic {
    clear: left;
}

#pic, #score, #rating {
    float: left;
}

#score {
    width: 50%;
}

#rating {
    background-color: green;
    color: white;
    font-size: 30pt;
    padding: 20px;
    margin: 5px;
}

body {
    background-color: grey;
}

h1 {
    text-align: center;
}

#all {
    background-color: white;
    margin-left: auto;
    margin-right: auto;
    overflow: hidden;
    padding: 20px;
    width: 80%;
}
```

## 2. NodeJS

```
app.get('/', function (req, res) {
  var letter = req.query.letter;
  var times = req.query.times;
  var json = {"names" : []};

  if (letter && times) {
    letter = letter.toLowerCase();
    var file = fs.readFileSync("peeps.txt", 'utf8').split("\n");
    for (var i = 0; i < file.length; i++) {
      var line = file[i].toLowerCase();
      var count = 0;
      for (var j = 0; j < line.length; j++) {
        var c = line[j];
        if (c == letter) {
          count++;
        }
      }
      if (count >= times) {
        json["names"].push({"name" : line, "count" : count});
      }
    }
    res.json(json);
  } else {
    res.status(400);
    res.send("oops");
  }
})
```

## 3. NodeJS/JSON

```
app.get('/', function (req, res) {
  var category = req.query.category;
  if (category) {
    var file = fs.readFileSync("books.txt", 'utf8').split("\n");
    var json = {"count" : file.length, {"books" : []}};
    for (var i = 0; i < file.length; i++) {
      var line = file[i].split("|");
      json["books"].push({"category" : category, "year" : line[3],
        "price" : line[4], "title" : line[0],
        "author" : line[1]});
    }
    res.json(json);
  } else {
    res.status(400);
    res.send("oops");
  }
})
```

## 4. JavaScript

```
window.onload = function() {
    makeRequest("begin");
};

function makeRequest(nodeid) {
    var url = "http://localhost:3000?nodeid=" + nodeid;
    fetch(url)
        .then(checkStatus)
        .then(function(responseText) {
            var json = JSON.parse(responseText);
            var situation = json.situation;
            if(situation) {
                ID("questionparagraph").innerHTML = situation;
                var answers = json.answers;
                ID("answer").innerHTML = "";
                for(var i = 0; i < answers.length; i++) {
                    var button = document.createElement("button");
                    button.innerHTML = answers[i]["answer"];
                    button.onclick = go;
                    ID("answer").appendChild(button);
                }
            } else {
                ID("questionparagraph").innerHTML = json.end;
                ID("answer").innerHTML = "";
            }
        })
        .catch(function(error) {
            console.log(error);
        });
}

function go() {
    makeRequest(this.innerHTML);
}
```

## 4. Regular Expressions

- a) `/^[0-9]{5}(-[0-9]{4})?$/`
- b) `^\$[0-9]*\.[0-9][0-9]$\`
- c) `<[a-zA-Z]+( [a-zA-Z]+="[a-zA-Z]" )*\>/`

## 5. JavaScript / DOM

```
window.onload = function() {
    document.getElementById("find").onclick = findClick;
};

function findClick() {
    document.getElementById("palindromes").innerHTML = "";
    var words = document.getElementById("phrase").value.split(" ");
    var count = 0;
    for (var i = 0; i < words.length; i++) {
        if (document.getElementById("min").value &&
            words[i].length < document.getElementById("min").value) { continue; }
        if (document.getElementById("max").value &&
            words[i].length > document.getElementById("max").value) { continue; }
        if (isPalindrome(words[i])) {
            var li = document.createElement("li");
            li.innerHTML = words[i];
            if (count % 2 == 0) {
                li.style.backgroundColor = "#cccccc";
            }
            document.getElementById("palindromes").appendChild(li);
            count++;
        }
    }
    document.getElementById("count").innerHTML = count + " total palindrome(s).";
}

function isPalindrome(s) {
    s = s.toLowerCase();
    for (var i = 0; i < s.length / 2; i++) {
        if (s[i] != s[s.length - 1 - i]) { return false; }
    }
    return true;
}
```

## 6. SQL

```
-- all the times a director has appeared in his own movie, twice

SELECT DISTINCT d.first_name, d.last_name
FROM directors d
JOIN movies_directors md ON md.director_id = d.id
JOIN roles r1 ON r1.movie_id = md.movie_id
JOIN roles r2 ON r2.movie_id = md.movie_id
JOIN actors a ON a.id = r1.actor_id AND a.id = r2.actor_id
WHERE a.first_name = d.first_name AND a.last_name = d.last_name
    AND r1.role < r2.role
ORDER BY d.last_name, d.first_name;
```