Tree: 63e3f80bc4 ▾   **bowling-challenge** / css / **styles.css**   Find file   Copy path

44 lines (34 sloc) | 730 Bytes

```css
 1   body {
 2       background-image: url(../images/bowling.jpg);
 3   }
 4
 5   .header {
 6       font-family: Impact, Haettenschweiler, 'Arial Narrow Bold', sans-serif;
 7       font-size: 70px;
 8       color: purple;
 9       text-shadow: black;
10   }
11
12   select {
13       font-family: Impact, Haettenschweiler, 'Arial Narrow Bold', sans-serif;
14       font-size: 35px;
15       color: purple;
16       width: 70px
17   }
18
19   button {
20       font-family: Impact, Haettenschweiler, 'Arial Narrow Bold', sans-serif;
21       font-size: 40px;
22       color: greenyellow;
23
24   }
25
26   #scoretable {
27       width: 40%;
28   }
29
30   label {
31       font-family: Impact, Haettenschweiler, 'Arial Narrow Bold', sans-serif;
32       font-size: 25px;
33       color: hotpink;
34
35   }
36
37   #ball {
38       position: relative;
39
40   }
41
42   #bonus-roll {
43       display: none;
44   }
```

Fetching contributors…

⬤ Cannot retrieve contributors at this time

45 lines (38 sloc) | 1.05 KB

```html
1   <html>
2
3   <head>
4       <link rel="stylesheet" href="css/bootstrap.css" />
5       <link rel="stylesheet" href="css/styles.css" />
6   </head>
7
8   <body>
9       <span class="header"></span>
10      <img src="images/bowlingkata.png" height="90px" alt="">
11      <span id='score'>Score = </span>
12      <div></div>
13
14      <!-- buttons -->
15      <label for="roll1">
16          Roll 1
17      </label>
18      <select id=roll1>
19      </select>
20      <label for="roll2">
21          Roll 2
22      </label>
23      <select id=roll2>
24      </select>
25      <button type="submit" id="roll">Roll!!</button>
26      <button type="submit" id="bonus-roll">Bonus Roll!</button>
27
28      <img id="ball" src="images/ball.png" alt="" width="100" height="100">
29
30      <!-- table -->
31      <table class="table table-striped table-hover" id="scoretable">
32          <thead>
33              <td>Frame</td>
34              <td>Roll 1</td>
35              <td>Roll 2</td>
36          </thead>
37      </table>
38
39      <script src="js/jquery.js"></script>
40      <script src="js/jquery-rotate.js"></script>
41      <script src="src/interface.js"></script>
42      <script src="src/game.js"></script>
43  </body>
44
45  </html>
```

33 lines (27 sloc) | 687 Bytes

```javascript
function rollSpare(game) {
    game.roll(6);
    game.roll(4);
};

function rollStrike(game) {
    game.roll(10);
    rollHelper(18, 1, game);
};

function rollStandardFrame(game) {
    game.roll(4);
    game.roll(2);
    rollHelper(18, 0, game);
};

function standardGame(game) {
    rollHelper(20, 3, game);
};

function perfectGame(game) {
    rollHelper(12, 10, game);
};

function gutterGame(game) {
    rollHelper(20, 0, game);
};

function rollHelper(numberOfRolls, numberOfPins, game) {
    for (i = 0; i < numberOfRolls; i++) {
        game.roll(numberOfPins);
    };
};
```

Fetching contributors…

Cannot retrieve contributors at this time

30 lines (29 sloc) │ 1011 Bytes

```javascript
 1   describe('Bowling Game', function () {
 2       var game;
 3       beforeEach(function () {
 4           game = new Game();
 5       });
 6
 7       describe('Game', function () {
 8           it('can roll standard game and keep score', function () {
 9               standardGame(game);
10               expect(game.score()).toEqual(60);
11           });
12           it('can roll a gutter game and keep score', function () {
13               gutterGame(game);
14               expect(game.score()).toEqual(0);
15           });
16           it('can roll a spare and add a bonus from the next roll', function () {
17               rollSpare(game);
18               rollStandardFrame(game);
19               expect(game.score()).toEqual(20);
20           });
21           it('can roll a strike, skipping second roll and add a bonus from next two rolls', function () {
22               rollStrike(game);
23               expect(game.score()).toEqual(30);
24           })
25           it('can handle a perfect game', function () {
26               perfectGame(game);
27               expect(game.score()).toEqual(300)
28           });
29       });
30   });
```

36 lines (30 sloc) | 1004 Bytes

```javascript
function Game() {
    this.scoreCard = [];
};

Game.prototype.roll = function (numberOfPinsDown) {
    this.scoreCard.push(numberOfPinsDown);
};

Game.prototype.score = function () {
    var overallScore = 0;
    var rollNumber = 0

    for (frame = 0; frame < 10; frame++) {

        if (this.isStrike(rollNumber)) {
            overallScore += 10 + this.scoreCard[rollNumber + 1] + this.scoreCard[rollNumber + 2];
            rollNumber++;
        } else {
            if (this.isSpare(rollNumber)) {
                overallScore += 10 + this.scoreCard[rollNumber + 2];
            } else {
                overallScore += this.scoreCard[rollNumber] + this.scoreCard[rollNumber + 1];
            }
            rollNumber += 2;
        }
    }
    return overallScore;
};

Game.prototype.isStrike = function (rollNumber) {
    return this.scoreCard[rollNumber] === 10;
};

Game.prototype.isSpare = function (rollNumber) {
    return this.scoreCard[rollNumber] + this.scoreCard[rollNumber + 1] === 10;
};
```

Fetching contributors…

⊘ Cannot retrieve contributors at this time

82 lines (70 sloc) | 2.15 KB

```javascript
$(function () {
    var game = new Game();
    var frame = 1;
    var lastFrame = 10;
    //Setup roll 1 drop down list
    $('#bonus-roll').hide();
    for (i = 0; i < 11; i++) {
        $("#roll1").append($("<option>", {
            value: i,
            text: i
        }));
        $("#roll2").append($("<option>", {
            value: i,
            text: i
        }));
    }


    $('#bonus-roll').click(function (event) {

        var firstBonusRoll = parseInt($("#roll1").val());
        var secondBonusRoll = parseInt($("#roll2").val());
    });

    $("#roll").click(function (event) {
        var firstRoll = parseInt($("#roll1").val());
        var secondRoll = parseInt($("#roll2").val());

        if (frame === 10 && firstRoll !== 10 && firstRoll + secondRoll === 10) {
            $('#bonus-roll').show();

            // lastFrame = 11;
            //if roll spare in frame 10 1 bonus ball
        }
        if (frame === 10 && firstRoll === 10) {
            $('#bonus-roll').show();

            // lastFrame = 11;
            //if roll strike in frame 10 2 bonus balls
        }
        if (frame <= lastFrame) {
            game.roll(firstRoll);
            game.roll(secondRoll);
            $("#scoretable").append($("<tr><td>" + frame + "</td><td>" + firstRoll + "</td><td>" + secondRoll + "</td></tr>"))

        } else {
            $("#score").text(game.score());
        }
        frame++;
    });

    // when roll 1 is 10 lock roll 2 to 0
    $("#roll1").change(function () {
        console.log($("#roll1").val());
        if ($("#roll1").val() == 10) {
            $("#roll2").val(0);
            $("#roll2").prop("disabled", true);
        } else {
            $("#roll2").prop("disabled", false);
        };
    });
    $("#roll").click(function () {
        $("#ball").animate({
            //   opacity: 0.25,
            left: "+=75"
            //   height: "toggle"
        }, 300, function () {
            // Animation complete.
        });
    })

    var rotation = function () {
        $("#ball").rotate({
            angle: 0,
            animateTo: 360,
            callback: rotation
        });
    }
    rotation();


});
```