Back to Resume Page

Blackjack (21) Programmed By: Christopher Singleton

Date: 03/09/2016

```
/*======== Initialize all variables and Set Arrays ==========
Suits have 4 of each cards 0=spades, 1=hearts, 2=clubs, 3=diamonds.
Number starts at 1 for aces.
pointValue array is set up to keep the values of the cards.
Note: This allows to handle the Ace as a 1 or 11.
Card deck is cut in half (0-11 are for the dealer, 12-23 for the player).
============================================================*/
var deck = [52];
var topCard;
var suit = ["s", "h", "c", "d"];
var pointValue = [0, 11, 2, 3, 4, 5, 6, 7, 8, 9, 10, 10, 10, 10];
/* entries 0-11 are for the dealer, 12-23 are for the player */
var dealerScore = Math.floor(Math.random()*(21 - 17 + 1)+17);
var hand = [24];
var cardValue = [24];
var nCards = [2];
var DEALER = 0;
var PLAYER = 1;
var losses = 0;
var wins = 0;
var playDelay = 10;        // Dealer play's his cards slower(adjustable).
"use strict";
/*======== Pass the image to the document layer =============
 Returns the item within a given layer; if the itemName is the
 null string, then this function returns the document.
============================================================*/
        function getItem( layerName, itemName ) {
                var       obj;
                if(itemName != ""|| itemName == isDOM)
                {
                        obj = document.getElementById( itemName );
                }
                else
                {
                        obj = document.getElementById( layerName );
                }
                return obj;
        }
```

```
/*============ Get the Style Items by divName ===============
 Returns a style item (used for setting visibility
        and movement of the picture objects).
=============================================================*/
        function getStyleItem( divName ) {
        var     obj;
        obj = document.all[divName].style;
        return obj; }
/*================= Get the card image =====================
 Returns the image object for card "n" in "player" hand.
=============================================================*/
        function getCardImage( player, n ) {
        var     v;
        n = player * 12 + n;
        v = getItem( "card" + n, "pic" + n );
        return v; }
/*========== Allow visibility of the Style Object ============
Show or hide an object using the "vis" parameter.
=============================================================*/
        function show( object, vis ) {
                var     obj;
                obj = getStyleItem( object );
                obj.visibility = vis; }
/*=============== Dealer shuffles the Deck ==================
        Shuffle the deck and get ready to play.
=============================================================*/
        function shuffle( n ) {
                var i, j, temp;
                for (i=0; i<n; i++)
                {
                        j = Math.floor( Math.random() * 52 );
                        temp = deck[i];
                        deck[i] = deck[j];
                        deck[j] = temp;
                }
        }
```

```
/*========= Don't show any cards, clear values ==============
 Don't show any cards on the table nor have any loaded as of yet.
============================================================*/
	function clearCards()
	{
	var	i,;
	var	v;
	// clear in reverse order so cards unstack properly
	for (i=11; i>=0; i--)
		{
			if (hand[i] != 0)
		{
			show("card" + i, "hidden");
			v = getItem( "card" + i, "pic" + i );
			v.src = "images/null.png";
			}
		hand[i] = cardValue[i] = 0;
		}
		for (i=23; i>=12; i--)
			{
				if (hand[i] != 0)
			{
				show("card" + i, "hidden");
				v = getItem( "card" + i, "pic" + i );
				v.src = "images/null.png";
		}
		hand[i] = cardValue[i] = 0;
		}
	// tell player & dealer that neither has any cards in hand
	nCards[DEALER] = nCards[PLAYER] = 0;
	}
```

```
/*=============================================================
 Displays card "n" in "dealerCard"'s hand to the player.
 A dealer card is placed face down and one right side up.
 Cards are deducted from part of the deck, so that it is not
 played again.
=============================================================*/
function showCard( dealerCard, n, faceUp )
{
	var theCard, theSuit, thePips;
	var img;

	theCard = hand[dealerCard*12 + n];
	theSuit = Math.floor((theCard - 1) / 13);
	thePips = ((theCard - 1) % 13) + 1;

	/* should it be face-up or face down? */
	img = getCardImage( dealerCard, n );

	if (!faceUp)
	{
		img.src = "images/facedown.png";
	}
	else
	{
		img.src = "images/" + suit[theSuit] + thePips + ".png";
	}
	show( "card" + (dealerCard*12 + n), "visible" );
}
```

```
/*============================================================
        Take a card off the deck and give it to the Dealer,
        then give cards to the player.
============================================================*/
        function giveCard( dealerCard, winLosses )
        {
                var n, theCard, thePips;
                var i;
                var temp = [52];
                var index12 = cardValue[12];
                var index13 = cardValue[13];
                var playerInitial = index12 + index13;
                //var wins = 0;
                //var losses = 0;
                n = nCards[dealerCard];
                hand[dealerCard*12 + n] = theCard = deck[topCard++];
                thePips = ((theCard - 1) % 13) + 1;
                cardValue[dealerCard*12 + n] = pointValue[thePips]
                nCards[dealerCard]++;

                /*Show the value for one of the dealers cards and show a question mark for the other one face down.
                  Show the two players cards (total amount) and update the score results.*/
                document.getElementById("dealerscore").innerHTML = "<span style=\"color:red\">"+"Dealer's Score:"+"</span>" +
                "<span style=\"color:#F16D17\">"+cardValue[0] +"</span>"+"<span style=\"color:red\">"+"  ?"+"</span>";

                document.getElementById("playerscore").innerHTML = "Player's Score: " +
                "<span style=\"color:green\">"+ playerInitial + "</span>";

                scores(wins,losses);

                var img = showCard( dealerCard, n, (n != 1 || dealerCard == PLAYER) );
                /* check to see if we're out of cards. */
                if (topCard == 52)
                {
                        window.status = "Reshuffling...";
```

```
                    /* Count how many cards are on the table right now */
                    n = 0;
                    for (i=0; i<24; i++)
                    {
                            if (hand[i] != 0)
                            {
                                    n++;
                            }
                    }
                    /* Shuffle the first 52-n cards that has already been played. */
                    shuffle( 52 - n );
                    /* move the newly shuffled discards down to the bottom of the deck */
                    for (i=51-n; i>=0; i--)
                            { deck[i+n] = deck[i]; }
                            /* and the cards currently on the table to the top of the deck
                            (they will get shuffled the next time we run out of cards) */
                    topCard = 0;
                    for (i=0; i<24; i++)
                            { if (hand[i] != 0)

                            deck[topCard++] = hand[i];
                            }
                }
        }
/*========= Pass cards to the Dealer and Player ==============
 This calls the function to deal the cards to the player, the
 dealer and hides the play and makes the play button visible.
============================================================*/
function dealHand( ) {
            giveCard( PLAYER );
            giveCard( DEALER );
            giveCard( PLAYER );
            giveCard( DEALER );
            show( "dealButton", "hidden" )
            show( "playButtons", "visible" );
          }
```

```
/*=============== Return the sum of the cards ================
 Return the sum of the cards from the dealer to the player.
============================================================*/
function sumHand( player ) {
        var sum, i;
        sum = 0;
        for (i=0; i<12; i++) { sum += cardValue[player*12+i]; }
        return sum;
}
/*============================================================
 Passing cards to the player (Hit Me). If the Ace is over
 21, try to go below 21 by valuing aces as 1 rather than 11.
 Dealer aces will be always 11. This code is basically Checking
 the cards in play for aces. If an ace is found and the player
 holds less than ten points, the value stays at 11, over ten
 points the value turns aces to ones and breaks the loop. If
 no aces are found or the player busts, break the loop in order
 to get out of the loop.
============================================================*/
        function hit( addCards ) {
        var sum,  i, hasAces;
        giveCard( addCards );
        sum = sumHand( addCards );
        if (addCards == PLAYER && sum > 21)
        { while (true)
                { hasAces = false;
                  for (i=0; i<12; i++)
                        { if (cardValue[addCards*12 + i] == 11)
                                { cardValue[addCards*12 + i] = 1;
                                        sum -= 10;
                                        hasAces = true;
                                        break; }
                        }
                        if (sum > 21 || (!hasAces))  { break; }
                }
        }
```

```
/* Add the losses for the player when the total is over 21.
            Update the score values to relect the changes while displaying
            a message.*/
          if ((sum > 21) && (addCards == PLAYER))
          {
                    losses++;
                    handFinished("<span style=\"color:red\">Sorry, your total is over 21! (You Bust!)</span>");
                    scores(wins,losses);
          }
          document.getElementById("playerscore").innerHTML = "Player's Score: " +
          "<span style=\"color:green\">"+ sum + "</span>";
          }

/*================== Dealer plays their cards ================
  Play the Dealer's cards, compare hands to see  wins, update the
  score and display a message in showing a result.
============================================================*/
          function dealerPlays()
          {
          var       sum, playerSum;
          sum = sumHand( DEALER );
          if (sum <= 16)
                    {
                              hit( DEALER );
                              playDelay = window.setTimeout("dealerPlays();", 10);
                              return;
                    }
          window.clearTimeout( playDelay );
          playerSum = sumHand( PLAYER );

          //When the Dealer play's out, show the dealer score.
          var ds = document.getElementById("dealerscore");
          ds.innerHTML = "Dealer's Score: "+"<span style=\"color:#F16D17\">" + sum + "</span>";

          //Show the updated Player's Score.
          var ps = document.getElementById("playerscore").innerHTML = "Player's Score: " +
          "<span style=\"color:green\">"+ playerSum + "</span>";
```

```
	/*Run a chain of if statements with operators in order to reflect the course of action
	  Display a message with the score values changing (add to wins or losses based on the
	  argument.*/
	if (sum > 21 && playerSum <= 21)
		{
			handFinished( "<span style=\"color:green\">Dealer Busts! (You Win!)</span>");
			wins++;
			scores(wins,losses);
		}
	else if (sum == playerSum )
		{
			handFinished( "<span style=\"color:black\">It's a Draw! (Tied)</span>" );
			scores(wins,losses);
		}
	else if (sum < playerSum && playerSum <= 21)
		{
			handFinished( "<span style=\"color:green\">You cleaned the table! (You Win!)</span> ");
			wins++;
			scores(wins,losses);
		}
	else
		{
			losses++;
			handFinished( "<span style=\"color:red\">Dealer takes all! (Dealer Wins!)</span>");
			scores(wins,losses);
		}
	}
/*=============== Dealer Cards Motion Timer ================
  The face down card is turned over on the dealer's turn and
  the dealer plays out in slower motion using a timer.
==========================================================*/
	function stand()
	{
		/* Facedown card is turned over. */
		showCard( DEALER, 1, true );
		playDelay = window.setTimeout( "dealerPlays();", 10);
	}
```

```
/*============ Button handling (Next Game) ==================
 The play is finished. Display the "next game button" area,
 hide the deal and play buttons. Then give a message.
============================================================*/
        function handFinished( msg )
        {
                var       doc;
                show("dealButton", "hidden");
                show("playButtons", "hidden");
                doc = getItem( "appendMsg", "" );
                str = msg + "<p><form>";
                str += "<div id='appendMsg'>";
                str += "</form></p>";
                doc.innerHTML = str;
                show( "nextGame", "visible" );
        }

/*=============== Reset to play again ======================
        Clear the cards allow to play again and the scores are
        reset to 0 on both the dealer and the player. Show the
        deal button and hide the nextGame button.
============================================================*/
        function newDeal()
        {
                clearCards();
                show( "nextGame", "hidden" );
                show( "dealButton", "visible" );
                beginDealersScore (dealerscore);
                beginPlayersScore (playerscore);
                document.getElementById("appendMsg").innerHTML = " ";
        }
```

```
/*============ Shuffle the cards and set up ==================
        Shuffle the deck, clear the cards then hide the next game
        button. Set all Scores and update win/losses.
============================================================*/
        window.onload=function setupGame() {
        var wins = 0;
        var losses = 0;
        var       i;
        isDOM = true;
        for (i=0; i<52; i++)
                {
                        deck[i] = i + 1;
                }
        shuffle( 52 );
        topCard = 0;
        clearCards();
        show( "nextGame", "hidden" );
        /*Show the heading strings on the Dealer's table (form).
          Show the Player's strings,Dealer's strings with Dealer scores
          Player's scores and Game Score Results.*/
        document.getElementById("h1Java").innerHTML = "JAVA";
        document.getElementById("h1Script").innerHTML = "SCRIPT";
        document.getElementById("h1Blackjack").innerHTML = "BLACKJACK";
        document.getElementById("dealerHand").innerHTML = "Dealer's Hand: ";
        document.getElementById("playerhand").innerHTML = "Player's Hand: ";
        beginDealersScore (dealerscore);
        beginPlayersScore (playerscore);
        scores(wins,losses);
        }
/*====================== Dealer's Score ======================
  This function returns the Score Result values (wins/losses).
  Clears the Dealer's score to 0 in the beginning.
============================================================*/
        function beginDealersScore (dealerscore)
        {
                document.getElementById("dealerscore").innerHTML = "Dealer's Score: "+
                "<span style=\"color:#F16D17\">0</span>";
        }
```

```
/*====================== Player's Score ========================
  This function returns the Score Result values (wins/losses).
  Clears the players score to 0 in the beginning.
==================================================================*/
        function beginPlayersScore (playerscore)
        {
                document.getElementById("playerscore").innerHTML = "Player's Score: "+
                "<span style=\"color:green\">0</span>";

        }

/*==================== Score Win / Losses ======================
  This function returns the Score Result values (wins/losses).
  Keeps track of wins/losses in displaying.
==================================================================*/
        function scores(wins, losses)
        {
                var wins;
                var losses;
                //Show the Game Results.
                document.getElementById("losses", "wins").innerHTML = "<br>"+"<span style=\"color:red\">Game Results: "+
                "<span style=\"color:green\">"+wins+"</span>"+"<span style=\"color:black\">/</span>"+
                "<span style=\"color:#F16D17\">"+losses+"</span>";
        }
```