(function () {

var questions = [{

question: "What does HTML stand for?",

choices: ["Hyper Trainer Marking Language", "Hyper Text Marketing Language", "Hyper Text Markup Language", "Hyper Text Markup Leveler"],

correctAnswer: 2,

isbalnk: false,

explain: "HTML stands for Hyper Text Markup Language"

}, {

question: "Choose the correct HTML element for the largest heading:",

choices: ["head", "heading", "h1", "h6"],

correctAnswer: 2,

isbalnk: false,

explain: "The order of headings are h1 > h2 > h3 > h4 > h5 > h6"

}, {

question: "\"hr\" tag is used for line break (True or Flase)?",

choices: ["True", "Flase"],

correctAnswer: 1,

isbalnk: false,

explain: "\"br\" tag is used for line break, hr tag represents a horizontal rule"

}, {

question: "We use \"a\" tag to bound links(True or Flase)?",

choices: ["True", "False"],

correctAnswer: 0,

isbalnk: false,

explain: "We use a tag for links and href contains destination link."

}, {

question: "HTML tag used to make text area?",

choices: [],

isbalnk: true,

correctAnswer: "textarea",

explain: "textarea is used to enter some text"

}, {

question: "HTML tag used to make display images?",

choices: [],

isbalnk: true,

correctAnswer: "img",

explain: "We use img tag to display image."

}];

var questionCounter = 0;

var selections = [];

var lockSelection = [false, false, false, false, false, false];

var markReview = [false, false, false, false, false, false];

var quiz = $('#quiz');

displayNext();

$('#showanswer').on('click', function (e) {

e.preventDefault();

showAnswers();

})

$('#exit').on('click', function (e) {

e.preventDefault();

endQuiz();

})

$('#mark').on('click', function (e) {

e.preventDefault();

mark();

if (quiz.is(':animated')) {

return false;

}

choose();

if ((selections[questionCounter] == -3)) {

alert('Please Answer the question.');

} else {

questionCounter++;

showHideAnswer();

displayNext();

}

})

$('#next').on('click', function (e) {

e.preventDefault();

if (quiz.is(':animated')) {

return false;

}

unmark();

choose();

if ((selections[questionCounter] == -3)) {

alert('Please Answer the question.');

} else {

questionCounter++;

showHideAnswer();

displayNext();

}

});

$('#prev').on('click', function (e) {

e.preventDefault();

if (quiz.is(':animated')) {

return false;

}

choose();

questionCounter--;

showHideAnswer();

displayNext();

});

$('#start').on('click', function (e) {

e.preventDefault();

if (quiz.is(':animated')) {

return false;

}

questionCounter = 0;

selections = [];

var lockSelection = [false, false, false, false, false, false];

var markReview = [false, false, false, false, false, false];

displayNext();

$('#start').hide();

});

function createQuestionElement(index) {

var qElement = $('<div>', {

id: 'question'

});

var header = $('<h2>Question ' + (index + 1) + ':</h2>');

qElement.append(header);

var question = $('<p>').append(questions[index].question);

qElement.append(question);

var radioButtons = createOptions(index);

qElement.append(radioButtons);

return qElement;

}

function createOptions(index) {

var radioList = $('<ul>');

var item;

var input = '';

if(questions[index].isbalnk) {

item = $('<li>');

if (lockSelection[index]) {

input = '<input disabled type="text" name="answer" />';

} else {

input = '<input type="text" name="answer" />';

}

item.append(input);

radioList.append(item);

} else {

for (var i = 0; i < questions[index].choices.length; i++) {

item = $('<li>');

if (lockSelection[index]) {

input = '<input disabled type="radio" name="answer" value=' + i + ' />';

} else {

input = '<input type="radio" name="answer" value=' + i + ' />';

}

input += questions[index].choices[i];

item.append(input);

radioList.append(item);

}

}

return radioList;

}

function showHideAnswer() {

if (lockSelection[questionCounter]) {

removeAnswer();

showAnswers();

} else {

removeAnswer()

}

}

function showAnswers() {

var aElement = $('#answerdisplay');

if(questions[questionCounter].isbalnk) {

var answer = $('<div id="inneranswer"><h3>Answer: ' + questions[questionCounter].correctAnswer+ '<br>Explaination: ' + questions[questionCounter].explain + '</h2></div>');

} else {

var answer = $('<div id="inneranswer"><h3>Answer: ' + questions[questionCounter].choices[questions[questionCounter].correctAnswer] + '<br>Explaination: ' + questions[questionCounter].explain + '</h2></div>');

}

aElement.append(answer);

lock();

}

function removeAnswer() {

var aElement = $('#inneranswer');

aElement.remove();

}

function choose() {

if(questions[questionCounter].isbalnk) {

if(markReview[questionCounter]) {

selections[questionCounter] = $('input[name="answer"]').val();

if (selections[questionCounter] == "" || selections[questionCounter] == undefined) {

selections[questionCounter] = -2;

}

} else if (lockSelection[questionCounter]) {

selections[questionCounter] = $('input[name="answer"]').val();

if (selections[questionCounter] == "" || selections[questionCounter] == undefined) {

selections[questionCounter] = -1;

}

} else {

selections[questionCounter] = $('input[name="answer"]').val();

}

if(selections[questionCounter] == "" || selections[questionCounter] == undefined) {

selections[questionCounter] = -3

}

} else {

if(markReview[questionCounter]) {

selections[questionCounter] = +$('input[name="answer"]:checked').val();

if (isNaN(selections[questionCounter])) {

selections[questionCounter] = -2;

}

} else if (lockSelection[questionCounter]) {

selections[questionCounter] = +$('input[name="answer"]:checked').val();

if (isNaN(selections[questionCounter])) {

selections[questionCounter] = -1;

}

} else {

selections[questionCounter] = +$('input[name="answer"]:checked').val();

}

if(isNaN(selections[questionCounter])) {

selections[questionCounter] = -3;

}

}

}

function mark() {

markReview[questionCounter] = true;

}

function unmark() {

markReview[questionCounter] = false;

}

function lock() {

lockSelection[questionCounter] = true;

var input = $('input[type=radio]');

if(questions[questionCounter].isbalnk) {

input = $('input[type=text]');

}

var i = 0;

for (i = 0; i < input.length; i++) {

input[i].disabled = true

}

}

function displayNext() {

quiz.fadeOut(function () {

$('#question').remove();

if (questionCounter < questions.length) {

var nextQuestion = createQuestionElement(questionCounter);

quiz.append(nextQuestion).fadeIn();

if (!(isNaN(selections[questionCounter]))) {

$('input[value=' + selections[questionCounter] + ']').prop('checked', true);

}

if (questionCounter === 1) {

$('#prev').show();

} else if (questionCounter === 0) {

$('#prev').hide();

$('#next').show();

$('#exit').show();

$('#mark').show()

$('#showanswer').show();

}

} else {

var scoreElem = displayScore();

quiz.append(scoreElem).fadeIn();

$('#next').hide();

$('#prev').hide();

$('#exit').hide();

$('#mark').hide();

$('#showanswer').hide();

$('#start').show();

}

});

}

function endQuiz() {

questionCounter = questions.length;

displayNext();

}

function displayScore() {

console.log(selections);

var score = $('<p>', { id: 'question' });

var questonWiseMarks = '';

var numCorrect = 0;

for(var i = 0; i < questions.length; i++) {

if(selections[i] != undefined) {

if (selections[i].toString().toLowerCase() === questions[i].correctAnswer.toString()) {

numCorrect++;

questonWiseMarks += '<br><h3>Question ' + (i+1) + ' you got 1 Point';

} else {

questonWiseMarks += '<br><h3>Question ' + (i+1) + ' you got 0 Point';

}

} else {

questonWiseMarks += '<br><h3>Question ' + (i+1) + ' you got 0 Point';

}

}

score.append(questonWiseMarks);

score.append('<br>Your Total Score is ' + numCorrect + '/' + questions.length);

return score;

}

})();