

De broncode van een Hot-Potatoes oefening. De belangrijkste wijzigingen en aanpassingen.

Alhoewel er complexe methodes zijn om een hot-potatoes oefening helemaal naar je hand te zetten (zie *hacking in hot-potatoes, a little knowledge brings a lot of power*, http://hotpot.uvic.ca/howto/hacking_hotpot.htm) proberen wij een kort overzicht te geven van onze belangrijkste wijzigingen. Onze aanpassingen situeren zich min of meer op het vlak van de lay-out, voor meer complexe aanpassingen (vb in de score) verwijzen we door naar eerder vermelde website.

```
<?xml version="1.0"?>
  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "xhtml11.dtd">
  <html xmlns="http://www.w3.org/1999/xhtml"
    xml:lang="en"><head><meta name="author" content="Created with Hot Potatoes by Half-Baked Software,
registered to kristof dhaenens.">
</meta><meta name="keywords" content="Hot Potatoes, Hot Potatoes, Half-Baked Software, Windows, University of
Victoria"></meta>
```

```
<link rel="schema.DC" href="http://purl.org/dc/elements/1.1/" />
<meta name="DC:Creator" content="kristof dhaenens" />
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
```

```
<title>
```

De titel van je oefening verschijnt sowieso in het blauwe balkje bovenaan je scherm. Indien je iets anders wil zien verschijnen (bijvoorbeeld 'welkom') moet je dit hier toevoegen.

```
</title>
```

```
<!-- Made with executable version 6.0 Release 4 Build 22 -->
```

```
<!-- The following insertion allows you to add your own code directly to this head tag from the configuration screen -->
```

```
<style type="text/css">
```

```
/* This is the CSS stylesheet used in the exercise. */
```

```
/* Elements in square brackets are replaced by data based on configuration settings when the exercise is built. */
```

```
/* BeginCorePageCSS */
```

```
/* Made with executable version 6.0 Release 4 Build 22 */
```

```
body{
```

```
font-family: Geneva,Arial,sans-serif;
```

```
background-color: #2d8b93;
```

```
color: #000000;
```

```
background:
```

```
margin-right: 5%; Hoe kleiner je dit percentage instelt, hoe dichter de oefening bij de rechterkant komt.
```

```
Margin-left: 5%; Hoe kleiner het percentage, hoe dichter de oefening bij de linkerkant komt. Het aanpassen van deze beide percentages kan extra ruimte opleveren om eventueel het moeten scrollen te voorkomen.
```

```
font-size: small
```

```
}
```

```
p{
```

```
text-align: left;
```

```
margin: 0px;
```

```
font-size: 100%;
```

```
}
```

```
table,div,span,td{
```

```
font-size: 100%;
```

```
color: #000000;
```

```
}
```

```

div.Titles{
    padding: 0.5em;;
    text-align: center;
    color: #000000;
}
button{
    font-family: Geneva,Arial,sans-serif;
    font-size: 100%;
    display: inline;
}

.ExerciseTitle{
    font-size: 140%;    Pas dit percentage aan als je de titel groter of kleiner wil.
    color: #000000;
}
.ExerciseSubtitle{
    font-size: 120%;    Pas dit percentage aan als je de ondertitel groter of kleiner wil.
    color: #000000;
}
div.StdDiv{
    background-color: #2d8b93;
    text-align: center;
    font-size: 100%;
    color: #000000;
    padding: 0.5em;
    border-style: solid;
    border-width: 1px 1px 1px 1px;    Door de vier elementen in border-width telkens op Opx te definiëren
    verdwijnen de standaard lijnen boven, links en rechts rond de
    oefening. Door de pixels >1 te definiëren kan je de lijnen 'dikker'
    maken.
    border-color: #000000;    Als je toch een kader rond de oefening wil kan je hier de kleur
    van de lijnen definiëren.
    margin-bottom: 1px;
}
/* EndCorePageCSS */

.RTLText{
    text-align: right;
    font-size: 150%;
    direction: rtl;
    font-family: "Simplified Arabic", "Traditional Arabic", "Times New Roman", Geneva,Arial,sans-serif;
}
.CentredRTLText{
    text-align: center;
    font-size: 150%;
    direction: rtl;
    font-family: "Simplified Arabic", "Traditional Arabic", "Times New Roman", Geneva,Arial,sans-serif;
}
button p.RTLText{
    text-align: center;
}
.RTLGapBox{
    text-align: right;
    font-size: 150%;
    direction: rtl;
    font-family: "Times New Roman", Geneva,Arial,sans-serif;
}
.Guess{
    font-weight: bold;
}
.CorrectAnswer{
    font-weight: bold;
}
div#Timer{

```

```

padding: 0.25em;
margin-left: auto;
margin-right: auto;
text-align: center;
color: #000000;
}
span#TimerText{
padding: 0.25em;
border-width: 1px;
border-style: solid;
font-weight: bold;
display: none;
color: #000000;
}
span.Instructions{
}
div.ExerciseText{
}
.FeedbackText, .FeedbackText span.CorrectAnswer, .FeedbackText span.Guess, .FeedbackText span.Answer{
color: #000000;
}

.LeftItem{
font-size: 100%;
color: #000000;
text-align: left;
}
.RightItem{
font-weight: bold;
font-size: 100%;
color: #000000;
}

span.CorrectMark{
}
input, textarea{
font-family: Geneva,Arial,sans-serif;
font-size: 120%;
}
select{
font-size: 100%;
}
Met div.Feedback kan je exact bepalen hoe je feedbackscherf eruit ziet. Je bepaalt de achtergrondkleur, de afstand tot de linkerkant van je scherm, de breedte van je feedbackscherf, het kader rond de feedback,...

div.Feedback {
background-color: #2d8b93; achtergrondkleur
left: 33%; afstand tot de linkerkant van het scherm
width: 34%; breedte van je feedbackscherf (in percentage tov het computerscherf)
top: 33%;
z-index: 1;
border-style: solid;
border-width: 1px; breedte van de lijnen rond het feedbackscherf
padding: 5px;
text-align: center; uitlijnen van de tekst in het feedbackscherf (left, center of right)
color: #000000; tekstkleur
position: absolute;
display: none;
font-size: 100%;
}
div.ExerciseDiv{
color: #000000;
}
/* JMatch flashcard styles */

```

```

table.FlashcardTable{
    background-color: transparent;
    color: #000000;
    border-color: #000000;
    margin-left: 5%;
    margin-right: 5%;
    margin-top: 2em;
    margin-bottom: 2em;
    width: 90%;
    position: relative;
    text-align: center;
    padding: 0px;
}
table.FlashcardTable tr{
    border-style: none;
    margin: 0px;
    padding: 0px;
    background-color: #2d8b93;
}
table.FlashcardTable td.Showing{
    font-size: 140%;
    text-align: center;
    width: 50%;
    display: table-cell;
    padding: 2em;
    margin: 0px;
    border-style: solid;
    border-width: 1px;
    color: #000000;
    background-color: #2d8b93;
}
table.FlashcardTable td.Hidden{
    display: none;
}
/* JMix styles */
div#SegmentDiv{
    margin-top: 2em;
    margin-bottom: 2em;
    text-align: center;
}
a.ExSegment{
    font-size: 120%;
    font-weight: bold;
    text-decoration: none;
    color: #000000;
}
span.RemainingWordList{
    font-style: italic;
}
div.DropLine {
    position: absolute;
    text-align: center;
    border-bottom-style: solid;
    border-bottom-width: 1px;
    border-bottom-color: #000000;
    width: 80%;
}
/* JCloze styles */
.ClozeWordList{
    text-align: center;
    font-weight: bold;
}
div.ClozeBody{
    text-align: left;
    margin-top: 2em;
    margin-bottom: 2em;
    line-height: 2.0
}

```

```

span.GapSpan{
    font-weight: bold;
}
/* JCross styles */

table.CrosswordGrid{
    margin: auto auto 1em auto;
    border-collapse: collapse;
    padding: 0px;
    background-color: #000000;
}
table.CrosswordGrid tbody tr td{
    width: 1.5em;
    height: 1.5em;
    text-align: center;
    vertical-align: middle;
    font-size: 140%;
    padding: 1px;
    margin: 0px;
    border-style: solid;
    border-width: 1px;
    border-color: #000000;
    color: #000000;
}
table.CrosswordGrid span{
    color: #000000;
}
table.CrosswordGrid td.BlankCell{
    background-color: #000000;
    color: #000000;
}
table.CrosswordGrid td.LetterOnlyCell{
    text-align: center;
    vertical-align: middle;
    background-color: #ffffff;
    color: #000000;
    font-weight: bold;
}
table.CrosswordGrid td.NumLetterCell{
    text-align: left;
    vertical-align: top;
    background-color: #ffffff;
    color: #000000;
    padding: 1px;
    font-weight: bold;
}
.NumLetterCellText{
    cursor: pointer;
    color: #000000;
}
.GridNum{
    vertical-align: super;
    font-size: 66%;
    font-weight: bold;
    text-decoration: none;
    color: #000000;
}
.GridNum:hover, .GridNum:visited{
    color: #000000;
}
table#Clues{
    margin: auto;
    vertical-align: top;
}
table#Clues td{
    vertical-align: top;
}
table.ClueList{
    margin: auto;
}

```

```

td.ClueNum{
    text-align: right;
    font-weight: bold;
    vertical-align: top;
}
td.Clue{
    text-align: left;
}
div#ClueEntry{
    text-align: left;
    margin-bottom: 1em;
}
/* Keypad styles */
div.Keypad{
    text-align: center;
    display: none; /* initially hidden, shown if needed */
}
div.Keypad button{
    font-family: Geneva,Arial,sans-serif;
    font-size: 120%;
    background-color: #ffffff;
    color: #000000;
    width: 2em;
}
/* JQuery styles */
div.QuestionNavigation{
    text-align: center;
}
.QNum{
    margin: 0em 1em 0.5em 1em;
    font-weight: bold;
    vertical-align: middle;
}
textarea{
    font-family: Geneva,Arial,sans-serif;
}
.QuestionText{
}
.Answer{
    font-size: 120%;
    letter-spacing: 0.1em;
}
.PartialAnswer{
    font-size: 120%;
    letter-spacing: 0.1em;
    color: #000000;
}
.Highlight{
    color: #000000;
    background-color: #ffff00;
    font-weight: bold;
    font-size: 120%;
}
ol.QuizQuestions{
    text-align: left;
    list-style-type: none;
}
li.QuizQuestion{
    padding: 1em;
    border-style: solid;
    border-width: 0px 0px 1px 0px; Hier kan je de lijn onder de quiz vragen bewerken. Opx is geen lijn.
}

ol.MCAnswers{

```

```
text-align: left;
list-style-type: upper-alpha;
```

Bij een normale quiz worden de vragen aangeduid met A, B of C. Wil je geen aanduidingen zet dan *none* ipv upper-alpha.
Andere mogelijkheden: *circle* (open rondje), *square* (massief vierkantje), *decimal* (decimaal getal, beginnend bij 1), *lower-alpha* (a,b,c)...

```
padding: 1em;
}
ol.MCAnswers li{
margin-bottom: 1em;
}
ol.MSelAnswers{
text-align: left;
list-style-type: lower-alpha;
padding: 1em;
}
div.ShortAnswer{
padding: 1em;
}
```

Bij J-Quiz kan je via de Function Button hier de knoppen (vb knoppen naast de antwoorden in een kwisvraag) aanpassen. Dit kan zorgen voor dynamische effecten. Het 'mouse over' effect zit standaard in de oefening, maar hier kan je de kleuren onbeperkt aanpassen.

```
.FuncButton { = de knop die je ziet op je scherm.
text-align: center; de positie van de tekst of afbeelding in de knop (left, center of right)
border-style: solid;
border-left-color: #96c5c9; kleur van de linkerlijn rond de knop
border-top-color: #96c5c9; kleur van de bovenste lijn rond de knop
border-right-color: #164549; kleur van de rechterlijn rond de knop
border-bottom-color: #164549; kleur van de onderste lijn
color: #000000; de tekskleur
background-color: #2d8b93; de achtergrondkleur
border-width: 2px; de breedte van de lijnen rond de knop.
padding: 3px 6px 3px 6px; de breedte van het kader (0,0,0,0=geen kader)
cursor: pointer;
}
.FuncButtonUp { = de knop als je er met de muis boven gaat.
color: #2d8b93;
text-align: center;
border-style: solid;
border-left-color: #96c5c9;
border-top-color: #96c5c9;
border-right-color: #164549;
border-bottom-color: #164549;

background-color: #000000;
color: #2d8b93;
border-width: 2px;
padding: 3px 6px 3px 6px;
cursor: pointer;
}
.FuncButtonDown { = de knop die je ziet als de muis niet meer op de knop staat.

color: #2d8b93;
text-align: center;
border-style: solid;
border-left-color: #164549;
border-top-color: #164549;
border-right-color: #96c5c9;
border-bottom-color: #96c5c9;
background-color: #000000;
color: #2d8b93;

border-width: 2px;
padding: 3px 6px 3px 6px;
cursor: pointer;
}
```

```
/*BeginNavBarStyle*/
```

```
div.NavButtonBar{  
    background-color: #2d8b93;  
    text-align: center;  
    margin: 2px 0px 2px 0px;  
    clear: both;  
    font-size: 100%;  
}
```

Bij alle oefeningen kan je ook de navigatieknop (volgende vraag of oefening) aanpassen. Ook dit is automatisch ingebouwd in Hot-Potatoes, maar hier kan je de kleur, breedte en achtergrondkleur ook weer volledig naar je hand zetten.

```
.NavButton { = de navigatieknop die je ziet op je scherm
```

```
    border-style: solid;  
    border-left-color: #96c5c9;  
    border-top-color: #96c5c9;  
    border-right-color: #164549;  
    border-bottom-color: #164549;  
    background-color: #2d8b93;  
    color: #000000;
```

```
    border-width: 2px;  
    cursor: pointer;
```

```
}
```

```
.NavButtonUp { = de navigatieknop als je er met de muis boven gaat.
```

```
    border-style: solid;  
    border-left-color: #96c5c9;  
    border-top-color: #96c5c9;  
    border-right-color: #164549;  
    border-bottom-color: #164549;  
    color: #2d8b93;  
    background-color: #000000;  
    border-width: 2px;  
    cursor: pointer;
```

```
}
```

```
.NavButtonDown { = de knop die je ziet als de muis niet meer op de knop staat.
```

```
    border-style: solid;  
    border-left-color: #164549;  
    border-top-color: #164549;  
    border-right-color: #96c5c9;  
    border-bottom-color: #96c5c9;  
    color: #2d8b93;  
    background-color: #000000;  
    border-width: 2px;  
    cursor: pointer;
```

```
}
```

```
/*EndNavBarStyle*/
```

```
a{  
    color: #0000FF;  
}  
a:visited{  
    color: #0000CC;  
}
```

```
a:hover{  
    color: #0000FF;  
}
```

```
div.CardStyle {  
    position: absolute;  
    font-family: Geneva,Arial,sans-serif;  
    font-size: 100%;  
    padding: 5px;  
    border-style: solid;  
    border-width: 1px;  
    color: #000000;  
    background-color: #2d8b93;  
    left: -50px;
```



```

        top: -50px;
        overflow: visible;
    }
    .rtl{
        text-align: right;
        font-size: 140%;
    }
</style>
<script type="text/javascript">
//
&lt;!--
function Client(){
//if not a DOM browser, hopeless
    this.min = false; if (document.getElementById){this.min = true;};
    this.ua = navigator.userAgent;
    this.name = navigator.appName;
    this.ver = navigator.appVersion;

//Get data about the browser
    this.mac = (this.ver.indexOf('Mac') != -1);
    this.win = (this.ver.indexOf('Windows') != -1);

//Look for Gecko
    this.gecko = (this.ua.indexOf('Gecko') &gt; 1);
    if (this.gecko){
        this.geckoVer = parseInt(this.ua.substring(this.ua.indexOf('Gecko')+6, this.ua.length));
        if (this.geckoVer &lt; 20020000){this.min = false;}
    }
//Look for Firebird
    this.firebird = (this.ua.indexOf('Firebird') &gt; 1);

//Look for Safari
    this.safari = (this.ua.indexOf('Safari') &gt; 1);
    if (this.safari){
        this.gecko = false;
    }
//Look for IE
    this.ie = (this.ua.indexOf('MSIE') &gt; 0);
    if (this.ie){
        this.ieVer = parseFloat(this.ua.substring(this.ua.indexOf('MSIE')+5, this.ua.length));
        if (this.ieVer &lt; 5.5){this.min = false;}
    }
//Look for Opera
    this.opera = (this.ua.indexOf('Opera') &gt; 0);
    if (this.opera){
        this.operaVer = parseFloat(this.ua.substring(this.ua.indexOf('Opera')+6, this.ua.length));
        if (this.operaVer &lt; 7.04){this.min = false;}
    }
    if (this.min == false){
        alert('Your browser may not be able to handle this page.');</pre>
</div>
```

```

function NavBtnDown(Btn){
    Btn.className = 'NavButtonDown';
}
//[strNavBarJS]

function FuncBtnOver(Btn){
    if (Btn.className != 'FuncButtonDown'){Btn.className = 'FuncButtonUp';}
}
function FuncBtnOut(Btn){
    Btn.className = 'FuncButton';
}
function FuncBtnDown(Btn){
    Btn.className = 'FuncButtonDown';
}
function FocusAButton(){
    if (document.getElementById('CheckButton1') != null){
        document.getElementById('CheckButton1').focus();
    }
    else{
        if (document.getElementById('CheckButton2') != null){
            document.getElementById('CheckButton2').focus();
        }
        else{
            document.getElementsByTagName('button')[0].focus();
        }
    }
}
//CODE FOR HANDLING DISPLAY OF POPUP FEEDBACK BOX

var topZ = 1000;
function ShowMessage(Feedback){
    var Output = Feedback + '<br /><br />';
    document.getElementById('FeedbackContent').innerHTML = Output;
    var FDiv = document.getElementById('FeedbackDiv');
    topZ++;
    FDiv.style.zIndex = topZ;
    FDiv.style.top = TopSettingWithScrollOffset(30) + 'px';
}

```

Hier kan je de verticale positie van het feedbackscherm exact bepalen. Als je de feedback naar beneden wil moet je een getal hoger dan 30 intikken. Wil je het scherm naar boven, dan moet je een kleiner getal invullen.

```

    FDiv.style.display = 'block';
    ShowElements(false, 'input');
    ShowElements(false, 'select');
    ShowElements(false, 'object');

//Focus the OK button
    setTimeout("document.getElementById('FeedbackOKButton').focus()", 50);
//
//    RefreshImages();
//
}
function ShowElements(Show, TagName){
//Special for IE bug -- hide all the form elements that will show through the popup
    if (C.ie){
        var Els = document.getElementsByTagName(TagName);
        for (var i=0; i<Els.length; i++){
            if (Show == true){
                Els[i].style.display = 'inline';
            }
            else{
                Els[i].style.display = 'none';
            }
        }
    }
}
function HideFeedback(){
}

```

```

        document.getElementById("FeedbackDiv").style.display = 'none';
        ShowElements(true, 'input');
        ShowElements(true, 'select');
        ShowElements(true, 'object');
        if (Finished == true){
            Finish();
        }
    }
}

//GENERAL UTILITY FUNCTIONS AND VARIABLES

//PAGE DIMENSION FUNCTIONS
function PageDim(){
//Get the page width and height
    this.W = 600;
    this.H = 400;
    this.W = document.getElementsByTagName('body')[0].clientWidth;
    this.H = document.getElementsByTagName('body')[0].clientHeight;
}
var pg = null;

function GetPageXY(EI) {
    var XY = {x: 0, y: 0};
    while(EI){
        XY.x += EI.offsetLeft;
        XY.y += EI.offsetTop;
        EI = EI.offsetParent;
    }
    return XY;
}
function GetScrollTop(){
    if (document.documentElement && document.documentElement.scrollTop){
        return document.documentElement.scrollTop;
    }
    else{
        if (document.body){
            return document.body.scrollTop;
        }
        else{
            return window.pageYOffset;
        }
    }
}
function GetViewportHeight(){
    if (window.innerHeight){
        return window.innerHeight;
    }
    else{
        return document.getElementsByTagName('body')[0].clientHeight;
    }
}
function TopSettingWithScrollOffset(TopPercent){
    var T = Math.floor(GetViewportHeight() * (TopPercent/100));
    return GetScrollTop() + T;
}

//CODE FOR AVOIDING LOSS OF DATA WHEN BACKSPACE KEY INVOKES history.back()
var InTextBox = false;

function SuppressBackspace(e){
    if (InTextBox == true){return;}
    if (C.ie) {
        thisKey = window.event.keyCode;
    }
    else {
        thisKey = e.keyCode;
    }
}
var Suppress = false;

```

```

    if (thisKey == 8) {
        Suppress = true;
    }

    if (Suppress == true){
        if (C.ie){
            window.event.returnValue = false;
            window.event.cancelBubble = true;
        }
        else{
            e.preventDefault();
        }
    }
}
if (C.ie){
    document.attachEvent('onkeydown', SuppressBackspace);
    window.attachEvent('onkeydown', SuppressBackspace);
}
else{
    if (window.addEventListener){
        window.addEventListener('keypress', SuppressBackspace, false);
    }
}

function ReduceItems(InArray, ReduceToSize){
    var ItemToDump=0;
    var j=0;
    while (InArray.length > ReduceToSize){
        ItemToDump = Math.floor(InArray.length*Math.random());
        InArray.splice(ItemToDump, 1);
    }
}

function Shuffle(InArray){
    var Num;
    var Temp = new Array();
    var Len = InArray.length;

    var j = Len;

    for (var i=0; i<Len; i++){
        Temp[i] = InArray[i];
    }

    for (i=0; i<Len; i++){
        Num = Math.floor(j * Math.random());
        InArray[i] = Temp[Num];

        for (var k=Num; k < (j-1); k++) {
            Temp[k] = Temp[k+1];
        }
        j--;
    }
    return InArray;
}

function WriteToInstructions(Feedback) {
    document.getElementById('InstructionsDiv').innerHTML = Feedback;

    RefreshImages();
}

}
imgs = new Array();

function PreloadImages(){
    var a = PreloadImages.arguments;
    for (var i=0; i<a.length; i++){
        imgs[i] = new Image();
        imgs[i].src = a[i];
    }
}
}

```

```

function RefreshImages(){
    for (var i=0; i<document.images.length; i++){
        if (document.images[i].name.substring(0,6) != 'NavBar'){
            document.images[i].src = document.images[i].src;
        }
    }
}
function EscapeDoubleQuotes(InString){
    return InString.replace(/"/g, '&quot;');
}
function TrimString(InString){
    var x = 0;

    if (InString.length != 0) {
        while ((InString.charAt(InString.length - 1) == '\u0020') || (InString.charAt(InString.length - 1) == '\u000A') ||
        (InString.charAt(InString.length - 1) == '\u000D')){
            InString = InString.substring(0, InString.length - 1)
        }

        while ((InString.charAt(0) == '\u0020') || (InString.charAt(0) == '\u000A') || (InString.charAt(0) == '\u000D')){
            InString = InString.substring(1, InString.length)
        }

        while (InString.indexOf(' ') != -1) {
            x = InString.indexOf(' ')
            InString = InString.substring(0, x) + InString.substring(x+1, InString.length)
        }

        return InString;
    }

    else {
        return "";
    }
}
function FindLongest(InArray){
    if (InArray.length < 1){return -1;}

    var Longest = 0;
    for (var i=1; i<InArray.length; i++){
        if (InArray[i].length > InArray[Longest].length){
            Longest = i;
        }
    }
    return Longest;
}
//UNICODE CHARACTER FUNCTIONS
function IsCombiningDiacritic(CharNum){
    var Result = (((CharNum >= 0x0300)&&(CharNum <= 0x370))||((CharNum >= 0x20d0)&&(CharNum <=
0x20ff)));
    Result = Result || (((CharNum >= 0x3099)&&(CharNum <= 0x309a))||((CharNum >= 0xfe20)&&(CharNum <=
0xfe23)));
    return Result;
}
function IsCJK(CharNum){
    return ((CharNum >= 0x3000)&&(CharNum < 0xd800));
}
//SETUP FUNCTIONS
//BROWSER WILL REFILL TEXT BOXES FROM CACHE IF NOT PREVENTED
function ClearTextBoxes(){
    var NList = document.getElementsByTagName('input');
    for (var i=0; i<NList.length; i++){
        if ((NList[i].id.indexOf('Guess') > -1)||((NList[i].id.indexOf('Gap') > -1)){
            NList[i].value = "";
        }
        if (NList[i].id.indexOf('Chk') > -1){
            NList[i].checked = "";
        }
    }
}
}

```

```

//EXTENSION TO ARRAY OBJECT
function Array_IndexOf(Input){
    var Result = -1;
    for (var i=0; i<this.length; i++){
        if (this[i] == Input){
            Result = i;
        }
    }
    return Result;
}
Array.prototype.indexOf = Array_IndexOf;

//IE HAS RENDERING BUG WITH BOTTOM NAVBAR
function RemoveBottomNavBarForIE(){
    if ((C.ie)&&(document.getElementById('Reading') != null)){
        if (document.getElementById('BottomNavBar') != null){

            document.getElementById('TheBody').removeChild(document.getElementById('BottomNavBar'));
        }
    }
}
//HOTPOTNET-RELATED CODE

var HPNStartTime = (new Date()).getTime();
var SubmissionTimeout = 30000;
var Detail = ""; //Global that is used to submit tracking data

function Finish(){
//If there's a form, fill it out and submit it
    if (document.store != null){
        Frm = document.store;
        Frm.starttime.value = HPNStartTime;
        Frm.endtime.value = (new Date()).getTime();
        Frm.mark.value = Score;
        Frm.detail.value = Detail;
        Frm.submit();
    }
}
//QUIZ CORE JAVASCRIPT CODE

```

De feedback vul je het best in bij het maken van de oefening, maar het kan hier ook!

```

var CurrQNum = 0;
var CorrectIndicator = 'ja !';
var IncorrectIndicator = 'fout !';
var YourScoreIs = 'Je resultaat is: ';
var ContinuousScoring = true;
var CorrectFirstTime = "";
var ShowCorrectFirstTime = true;
var ShuffleQs = false;
var ShuffleAs = false;
var DefaultRight = 'goed zo !';
var DefaultWrong = 'Sorry ! Probeer opnieuw !';
var QsToShow = 0;
var Score = 0;
var Finished = false;
var Qs = null;
var QArray = new Array();
var ShowingAllQuestions = false; Wil je toch alle vragen zien, zet dan true ipv false
var ShowAllQuestionsCaption = "";
var ShowOneByOneCaption = 'Show questions one by one!';
var State = new Array();
var Feedback = "";
var TimeOver = false;
var strInstructions = "";
var QsToShow = 7;          Indien je bij een match-oefening toch een vraag bijmaakt in de broncode moet
                           je de variabele 'aantal vragen tonen' aanpassen.

```

```

//The following variable can be used to add a message explaining that
//the question is finished, so no further marking will take place.

```

```

var strQuestionFinished = "";

function CompleteEmptyFeedback(){
    var QNum, ANum;
    for (QNum=0; QNum<I.length; QNum++){
//Only do this if not multi-select
        if (I[QNum][2] != '3'){
            for (ANum = 0; ANum<I[QNum][3].length; ANum++){
                if (I[QNum][3][ANum][1].length < 1){
                    if (I[QNum][3][ANum][2] > 0){
                        I[QNum][3][ANum][1] = DefaultRight;
                    }
                    else{
                        I[QNum][3][ANum][1] = DefaultWrong;
                    }
                }
            }
        }
    }
}

function SetUpQuestions(){
    var AList = new Array();
    var QList = new Array();
    var i, j;
    Qs = document.getElementById('Questions');
    while (Qs.getElementsByTagName('li').length > 0){
        QList.push(Qs.removeChild(Qs.getElementsByTagName('li')[0]));
    }
    var Dumpltem = 0;
    if (QsToShow > QList.length){
        QsToShow = QList.length;
    }
    while (QsToShow < QList.length){
        Dumpltem = Math.floor(QList.length*Math.random());
        for (j=Dumpltem; j<(QList.length-1); j++){
            QList[j] = QList[j+1];
        }
        QList.length = QList.length-1;
    }
    if (ShuffleQs == true){
        QList = Shuffle(QList);
    }
    if (ShuffleAs == true){
        var As;
        for (var i=0; i<QList.length; i++){
            As = QList[i].getElementsByTagName('ol')[0];
            if (As != null){
                AList.length = 0;
                while (As.getElementsByTagName('li').length > 0){
                    AList.push(As.removeChild(As.getElementsByTagName('li')[0]));
                }
                AList = Shuffle(AList);
                for (j=0; j<AList.length; j++){
                    As.appendChild(AList[j]);
                }
            }
        }
    }

    for (i=0; i<QList.length; i++){
        Qs.appendChild(QList[i]);
        QArray[QArray.length] = QList[i];
    }

//Show the first item
    QArray[0].style.display = "";

//Now hide all except the first item
    for (i=1; i<QArray.length; i++){
        QArray[i].style.display = 'none';
    }
}

```

```

    }
    SetQNumReadout();

    SetFocusToTextbox();
}

function SetFocusToTextbox(){
//if there's a textbox, set the focus in it
    if (QArray[CurrQNum].getElementsByTagName('input')[0] != null){
        QArray[CurrQNum].getElementsByTagName('input')[0].focus();
//and show a keypad if there is one
        if (document.getElementById('CharacterKeypad') != null){
            document.getElementById('CharacterKeypad').style.display = 'block';
        }
    }
    else{
        if (QArray[CurrQNum].getElementsByTagName('textarea')[0] != null){
            QArray[CurrQNum].getElementsByTagName('textarea')[0].focus();
//and show a keypad if there is one
            if (document.getElementById('CharacterKeypad') != null){
                document.getElementById('CharacterKeypad').style.display = 'block';
            }
        }
//This added for 6.0.4.11: hide accented character buttons if no textbox
        else{
            if (document.getElementById('CharacterKeypad') != null){
                document.getElementById('CharacterKeypad').style.display = 'none';
            }
        }
    }
}

function ChangeQ(ChangeBy){
//The following line prevents moving to another question until the current
//question is answered correctly. Uncomment it to enable this behaviour.
//    if (State[CurrQNum][0] == -1){return;}
    if (((CurrQNum + ChangeBy) < 0)||((CurrQNum + ChangeBy) >= QArray.length)){return;}
    QArray[CurrQNum].style.display = 'none';
    CurrQNum += ChangeBy;
    QArray[CurrQNum].style.display = "";
//Undocumented function added 10/12/2004
    ShowSpecialReadingForQuestion();
    SetQNumReadout();
    SetFocusToTextbox();
}

var HiddenReadingShown = false;
function ShowSpecialReadingForQuestion(){
//Undocumented function for showing specific reading text elements which change with each question
//Added on 10/12/2004
    if (document.getElementById('ReadingDiv') != null){
        if (HiddenReadingShown == true){
            document.getElementById('ReadingDiv').innerHTML = "";
        }
        if (QArray[CurrQNum] != null){
            var Children = QArray[CurrQNum].childNodes;
            for (var i=0; i<Children.length; i++){
                if (Children[i].className=="HiddenReading"){
                    document.getElementById('ReadingDiv').innerHTML =
Children[i].innerHTML;
                    HiddenReadingShown = true;
//Hide the ShowAllQuestions button to avoid confusion
                    if (document.getElementById('ShowMethodButton') != null){
                        document.getElementById('ShowMethodButton').style.display =
'none';
                    }
                }
            }
        }
    }
}

function SetQNumReadout(){

```



```

document.getElementById('QNumReadout').innerHTML = (CurrQNum+1) + ' / ' + QArray.length;
if ((CurrQNum+1) >= QArray.length){
    if (document.getElementById('NextQButton') != null){
        document.getElementById('NextQButton').style.visibility = 'hidden';
    }
}
else{
    if (document.getElementById('NextQButton') != null){
        document.getElementById('NextQButton').style.visibility = 'visible';
    }
}
if (CurrQNum <= 0){
    if (document.getElementById('PrevQButton') != null){
        document.getElementById('PrevQButton').style.visibility = 'hidden';
    }
}
else{
    if (document.getElementById('PrevQButton') != null){
        document.getElementById('PrevQButton').style.visibility = 'visible';
    }
}
}
}

```

Stel dat je al heel wat aanpassingen hebt aangebracht in de broncode.

Je oefening is klaar maar dan merk je dat er een taalfout of andere fout in één van je vragen staat. Je kan gemakkelijk de vraag corrigeren in het oorspronkelijk Hot-Potatoes aanmaakblad. Maar als je die opslaat gaan de wijzigingen die je aanbracht in de broncode onherroepelijk verloren.

Geen probleem, kleine aanpassingen in de vragen en de feedback (individuele feedback, per vraag of antwoord) kan je hier aanbrengen. Indien je de logische opbouw volgt kan je zelfs vragen toevoegen. Dit is al moeilijker. Denk dus eerst goed na over de vragen voor je prutst in de broncode van de oefening. Als je toch een vraag bijmaakt in een match oefening, vergeet dan niet de variable VarQsToShow aan te passen (zie eerder)

!! De feedback staat tussen aanhalingstekens. Gebruik NOOIT een apostrof in je feedback (vb auto's) want dan krijg je gegarandeerd een foutmelding en werkt de oefening niet. Het kan heel lang duren voor je deze fout gevonden hebt. Je kan wel de html-vertaling van een aanhalingsteken gebruiken, ’ (vb auto’s)

```

l[0]=new Array();l[0][0]=100;
l[0][1]="";
l[0][2]='0';
l[0][3]=new Array();
l[0][3][0]=new Array('Op spoor 11','<b>Niet goed. De trien op spoor 11 rijdt naar Berchem.<br><br>Kijk goed naar het bord!</b>',0,0,1);
l[0][3][1]=new Array('Op spoor 9','<b>Niet goed. Carlo gaat niet naar oostende of Geraardsbergen.<br><br>Kijk goed naar het bord!</b>',0,0,1);
l[0][3][2]=new Array('Op spoor 5','<b>Ja, Dat is het juste spoor!<br><br>Ga naar de volgende oefening.</b>',1,100,1);
l[1]=new Array();l[1][0]=100;
l[1][1]="";
l[1][2]='0';
l[1][3]=new Array();
l[1][3][0]=new Array('De trein naar Brussel.','<b>Niet goed. Deze trein stopt niet in Halle.<br><br>Kijk goed naar het bord!</b>',0,0,1);
l[1][3][1]=new Array('De trein naar Hasselt.','<b>Niet goed. Deze trein stopt niet in HALLE.<br><br>Kijk goed naar het bord!</b>',0,0,1);
l[1][3][2]=new Array('De trein richting Geraardsbergen.','<b>Goed. Deze trein stopt in Halle.<br><br>Ga naar de volgende oefening.</b>',1,100,1);
l[2]=new Array();l[2][0]=100;
l[2][1]="";
l[2][2]='0';
l[2][3]=new Array();
l[2][3][0]=new Array('8.05','<b>Neeen, dat is de trein met bestemming Antwerpen. <br>Hij stopt niet in Halle.<br><br>Kijk goed naar het bord!</b>',0,0,1);
l[2][3][1]=new Array('8.30','<b>Neeen, deze trein rijdt tot Hasselt. De trein stopt niet in HALLE.<br><br>Kijk goed naar het bord!</b>',0,0,1);
l[2][3][2]=new Array('8.23','<b>Goed. Deze trein rijdt naar Geraardsbergen.<br>De trein stopt in Halle!<br><br>Ga naar de volgende oefening.</b>',1,100,1);
l[3]=new Array();l[3][0]=100;
l[3][1]="";

```

```

I[3][2]='0';
I[3][3]=new Array();
I[3][3][0]=new Array('half negen','<b>Niet goed. Dat is de trein naar Hasselt.<br><br><br>Kijk goed naar het
bord!</b>',0,0,1);
I[3][3][1]=new Array('8.05','<b>Niet goed. De trein naar Brussel heeft <font color="red">5 minuten</font>
vertraging.<br><br>Kijk goed naar het bord!<b>',0,0,1);
I[3][3][2]=new Array('tien over acht','<b>Goed! Normaal vertrekt de trein om 8.05.<br>Vandaag vertrekt de trein met
<br><font color="red">5 minuten</font> vertraging.<br><br>Ga naar de volgende oefening.</b>',1,100,1);
I[4]=new Array();I[4][0]=100;
I[4][1]="";
I[4][2]='0';
I[4][3]=new Array();
I[4][3][0]=new Array('Om half negen.','<b>Niet correct.<br><br>Kijk goed naar het bord!</b>',0,0,1);
I[4][3][1]=new Array('Om tien over acht.','<b>Niet goed. De trein <i>vertrekt</i> om 8.10</b><br><br>Kijk goed naar het
bord!',0,0,1);
I[4][3][2]=new Array('Om kwart voor negen.','<b>Goed!<br><br>Je bent klaar.</b>',1,100,1);

```

eventueel bijkomende vraag

```

I[5]=new Array();I[5][0]=100;
I[5][1]='';
I[5][2]='0';
I[5][3]=new Array();
I[5][3][0]=new Array('feedback',0,0,1);           (0,0,1 = fout antwoord)
I[5][3][1]=new Array('feedback',0,0,1);
I[5][3][2]=new Array('feedback',1,100,1);         (1,100,1= correct antwoord)

```

```

I=new Array();

```

```

function StartUp(){
    RemoveBottomNavBarForIE();

    //If there's only one question, no need for question navigation controls
    if (QsToShow < 2){
        document.getElementById('QNav').style.display = 'none';
    }

    //Stash the instructions so they can be redisplayed
    strInstructions = document.getElementById('InstructionsDiv').innerHTML;

    PreloadImages('afbeeldingen/huilen.gif');

    CompleteEmptyFeedback();

    SetUpQuestions();
    ClearTextBoxes();
    CreateStatusArray();

    //Check search string for q parameter
    if (document.location.search.length > 0){
        if (ShuffleQs == false){
            var JumpTo =
parseInt(document.location.search.substring(1,document.location.search.length))-1;
            if (JumpTo <= QsToShow){
                ChangeQ(JumpTo);
            }
        }
    }

    //Undocumented function added 10/12/2004
    ShowSpecialReadingForQuestion();
}
function ShowHideQuestions(){
    FuncBtnOut(document.getElementById('ShowMethodButton'));
    document.getElementById('ShowMethodButton').style.display = 'none';
    if (ShowingAllQuestions == false){
        for (var i=0; i<QArray.length; i++){
            QArray[i].style.display = "";
        }
        document.getElementById('Questions').style.listStyleType = 'decimal';
    }
}

```

```

document.getElementById('OneByOneReadout').style.display = 'none';
document.getElementById('ShowMethodButton').innerHTML = ShowOneByOneCaption;
ShowingAllQuestions = true;
}
else{
    for (var i=0; i<QArray.length; i++){
        if (i != CurrQNum){
            QArray[i].style.display = 'none';
        }
    }
    document.getElementById('Questions').style.listStyleType = 'none';
    document.getElementById('OneByOneReadout').style.display = "";
    document.getElementById('ShowMethodButton').innerHTML = ShowAllQuestionsCaption;
    ShowingAllQuestions = false;
}
document.getElementById('ShowMethodButton').style.display = 'inline';
}

function CreateStatusArray(){
    var QNum, ANum;
    //For each item in the item array
    for (QNum=0; QNum<l.length; QNum++){
    //Check if the question still exists (hasn't been nuked by showing a random selection)
        if (document.getElementById('Q_' + QNum) != null){
            State[QNum] = new Array();
            State[QNum][0] = -1; //Score for this q; -1 shows question not done yet
            State[QNum][1] = new Array(); //answers
            for (ANum = 0; ANum<l[QNum][3].length; ANum++){
                State[QNum][1][ANum] = 0; //answer not chosen yet; when chosen, will store its
                position in the series of choices
            }
            State[QNum][2] = 0; //tries at this q so far
            State[QNum][3] = 0; //incrementing percent-correct values of selected answers
            State[QNum][4] = 0; //penalties incurred for hints
            State[QNum][5] = ""; //Sequence of answers chosen by number
        }
        else{
            State[QNum] = null;
        }
    }
}

function CalculateOverallScore(){
    var TotalWeighting = 0;
    var TotalScore = 0;

    for (var QNum=0; QNum<State.length; QNum++){
        if (State[QNum] != null){
            if (State[QNum][0] > -1){
                TotalWeighting += l[QNum][0];
                TotalScore += (l[QNum][0] * State[QNum][0]);
            }
        }
    }
    if (TotalWeighting > 0){
        Score = Math.floor((TotalScore/TotalWeighting)*100);
    }
    else{
    //if TotalWeighting is 0, no questions so far have any value, so
    //no penalty should be shown.
        Score = 100;
    }
}

function CheckFinished(){
    var FB = "";
    var AllDone = true;
    for (var QNum=0; QNum<State.length; QNum++){
        if (State[QNum] != null){
            if (State[QNum][0] < 0){
                AllDone = false;
            }
        }
    }
}

```

```

    }
    }
    if (AllDone == true){
//Report final score and submit if necessary
    CalculateOverallScore();
    FB = YourScores + ' ' + Score + '%.';
    if (ShowCorrectFirstTime == true){
        var CFT = 0;
        for (QNum=0; QNum<State.length; QNum++){
            if (State[QNum] != null){
                if (State[QNum][0] >= 1){
                    CFT++;
                }
            }
        }
        FB += '<br />' + CorrectFirstTime + ' ' + CFT + '/' + QsToShow;
    }
    WriteToInstructions(FB);
    Finished == true;
    TimeOver = true;
    Locked = true;
    Finished = true;
    Detail = '<?xml version="1.0"?><hpnetresult><fields>';
    for (QNum=0; QNum<State.length; QNum++){
        if (State[QNum] != null){
            if (State[QNum][5].length > 0){
                Detail += '<field><fieldname>Question #' + (QNum+1) +
'</fieldname><fieldtype>question-tracking</fieldtype><fieldlabel>Q ' + (QNum+1) +
'</fieldlabel><fieldlabelid>QuestionTrackingField</fieldlabelid><fielddata>' + State[QNum][5] + '</fielddata></field>';
            }
        }
    }
    Detail += '</fields></hpnetresult>';
    setTimeout('Finish()', SubmissionTimeout);
}
}
//-->
//]]>

</script>

</head>
<body onload="Startup()" id="TheBody">

<!-- BeginTopNavButtons -->

<!-- EndTopNavButtons -->

<!-- EndTopNavButtons -->

```

VANAF HIER BEGINT DE EIGENLIJKE OEFENING!!!

Indien je in de titel of in de ondertitel twee kleuren wil gebruiken moet je een korte HTML-tag invoegen. De tekst tussen `` en `` krijgt de door jou aangeduide kleur (hier rood). De rest van de tekst behoudt de kleur, gedefinieerd bij de oorspronkelijke opmaak van de oefening.

Wij gebruikten een steeds terugkerende rode kleur om de herkenbaarheid van de instructie te vergroten (Wat moet je doen?).

```

<div id="InstructionsDiv" class="StdDiv">
    <p id="Instructions"> <br><br><center><table bgcolor="white"><TR><td><b><font color="red">Wat
moet je doen?</font></b> Klik op de knop "dienstregeling". Lees de vragen. Zoek het antwoord op het bord met de
dienstregeling.</b></tr></table></center></p>
</div>

<div id="MainDiv" class="StdDiv">

<div id="QNav" class="QuestionNavigation">

```



```

onclick="CheckMCAnswer(3,2,this)">&nbsp;&nbsp; ?&nbsp;&nbsp;&nbsp;&nbsp;</button>&nbsp;&nbsp;&nbsp;tien over acht</li></ol></li>
<li class="QuizQuestion" id="Q_4" style="display: none;"><p class="QuestionText"><b>5. Hoe laat komt Carlo normaal
in Brussel aan?</b></p><ol class="MCAnswers"><li id="Q_4_0"><button class="FuncButton"
onfocus="FuncBtnOver(this)" onblur="FuncBtnOut(this)" onmouseover="FuncBtnOver(this)"
onmouseout="FuncBtnOut(this)" onmousedown="FuncBtnDown(this)" onmouseup="FuncBtnOut(this)" id="Q_4_0_Btn"
onclick="CheckMCAnswer(4,0,this)">&nbsp;&nbsp;&nbsp;?&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</button>&nbsp;&nbsp;&nbsp;Om half negen.</li><li
id="Q_4_1"><button class="FuncButton" onfocus="FuncBtnOver(this)" onblur="FuncBtnOut(this)"
onmouseover="FuncBtnOver(this)" onmouseout="FuncBtnOut(this)" onmousedown="FuncBtnDown(this)"
onmouseup="FuncBtnOut(this)" id="Q_4_1_Btn"
onclick="CheckMCAnswer(4,1,this)">&nbsp;&nbsp;&nbsp;?&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</button>&nbsp;&nbsp;&nbsp;Om tien over acht.</li><li
id="Q_4_2"><button class="FuncButton" onfocus="FuncBtnOver(this)" onblur="FuncBtnOut(this)"
onmouseover="FuncBtnOver(this)" onmouseout="FuncBtnOut(this)" onmousedown="FuncBtnDown(this)"
onmouseup="FuncBtnOut(this)" id="Q_4_2_Btn"
onclick="CheckMCAnswer(4,2,this)">&nbsp;&nbsp;&nbsp;?&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</button>&nbsp;&nbsp;&nbsp;Om kwart voor
negen.</li></ol></li></ol>
<td width="100">
</TR>
</TABLE></center>
</div>

<div class="Feedback" id="FeedbackDiv">
<div class="FeedbackText" id="FeedbackContent"></div>
<button id="FeedbackOKButton" class="FuncButton" onfocus="FuncBtnOver(this)" onblur="FuncBtnOut(this)"
onmouseover="FuncBtnOver(this)" onmouseout="FuncBtnOut(this)" onmousedown="FuncBtnDown(this)"
onmouseup="FuncBtnOut(this)" onclick="HideFeedback(); return false;">&nbsp;&nbsp;&nbsp;OK&nbsp;&nbsp;&nbsp;</button>
</div>

<!-- BeginBottomNavButtons -->

<!-- EndBottomNavButtons -->

<!-- BeginSubmissionForm -->

<!-- EndSubmissionForm -->

</body>
</html>

```