Learning Javascript using puzzles and games

This page describes the ship game [1]

It is similar to the traditional game of "Hangman".

Making your own games

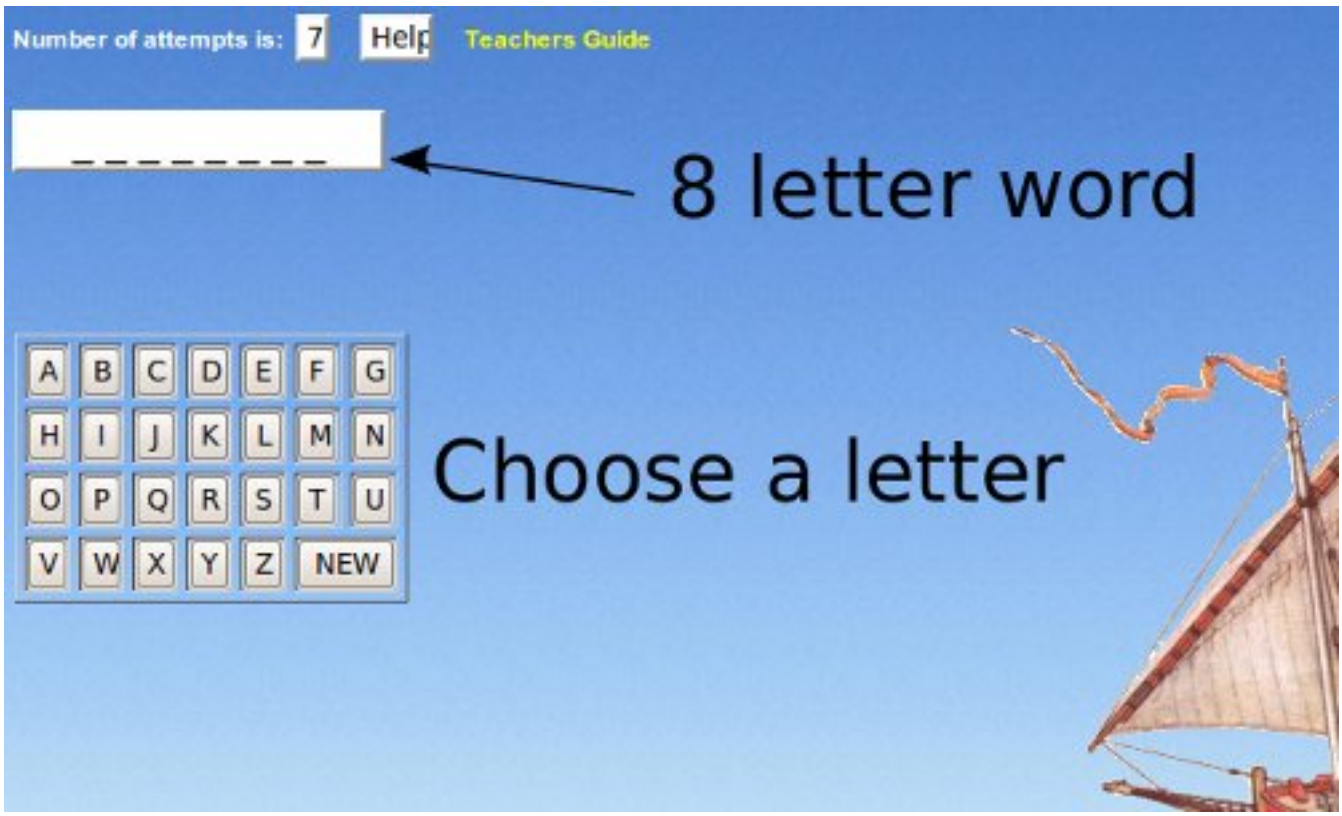
The aim of this document is to enable you to learn how the game works so you can make your own games. Don't worry, you don't need to know all the programming details to start off with. You could, for example, change the ship picture to something different. The ship pictures used are:

- **ship0.png**
- **ship0.png**
- **ship1.png**
- **ship2.png**
- **ship3.png**
- **ship4.png**
- **ship5.png**
- **ship6.png**
- **ship7.png**

So if you replace the references in the program code to ship0.png with a different picture you will change the game a bit. You can learn about how different parts of the program works by experimenting and making your own similar games. Of course we would like you to contribute them so everyone else can use them.

What you need to know to get started

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');



The game **Ship** uses **Javascript** and standard **HTML**. Javascript is a computer programming language used by web browsers to make pages that are described by HTML tags to be interactive.

In this game, the players have to guess a word by choosing a letter at a time. Underscore characters display the number of the letters in the word. The player uses a mouse to click on the letters of the alphabet below the word being guessed.

If the player's guess is accurate, the correct letter is displayed.

If the player's guess is inaccurate, the ship sinks a little.

After seven attempts, the ship will sink completely and the player will see a notification announcing that they were unsuccessful in guessing the word. (The number of guess attempts remaining is displayed in the game by a text "**Number of Attempts**").

If the player succeeds, the table will display a message indicating that the player has been successful. The time it took the player to guess the word correctly is displayed and a message asking if th player wants to repeat the game.

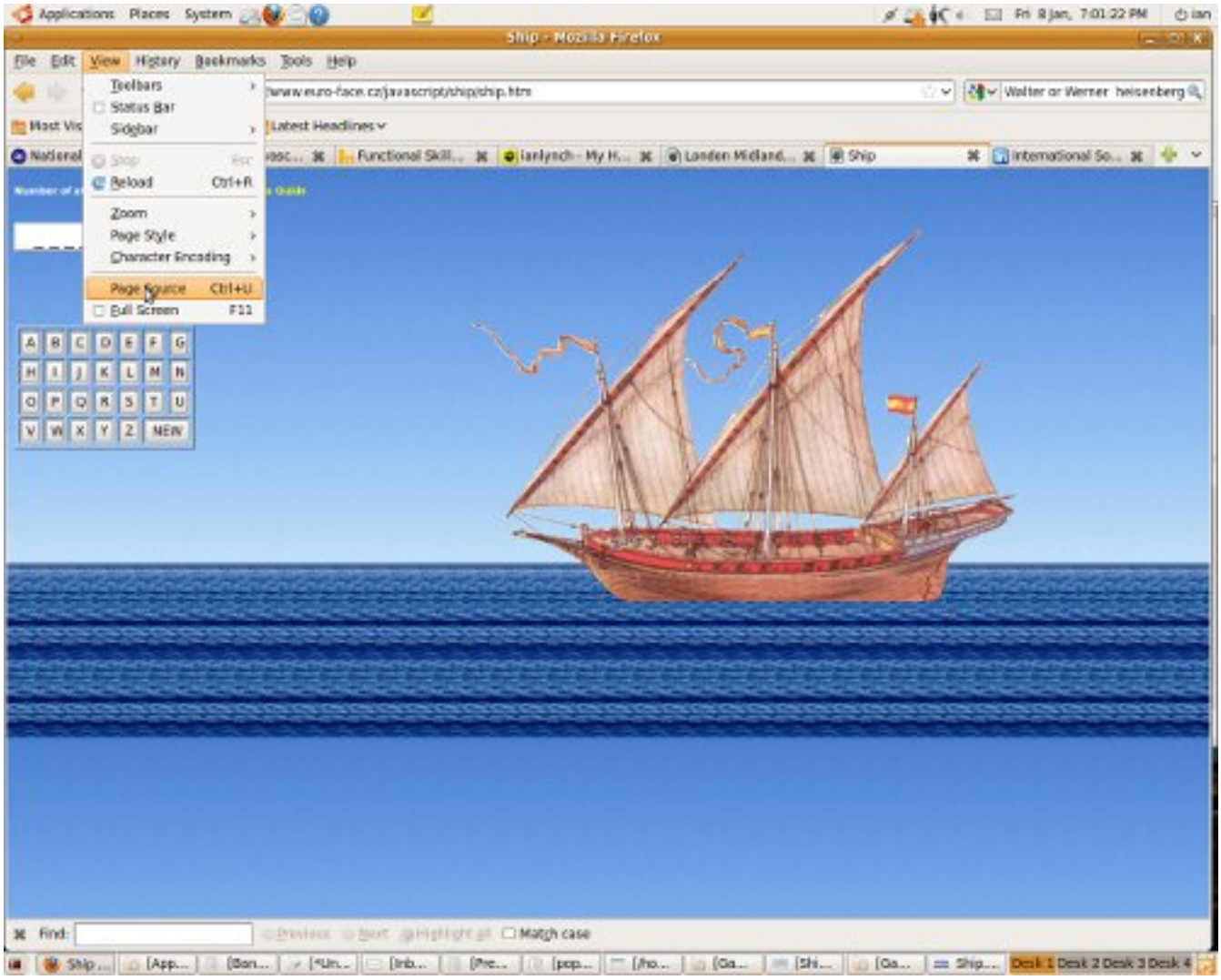
The player can use a help feature (the "**Help**" button), which describes the word being guessed. The game uses a database of words related to computer technology. Anothr simple way to change the game would be to change the words to a different topic. For example a game where all the words are capital cities or words related to biology.

A "**Teachers Guide**" button is the link to a brief description of the game and its purpose in supporting the **Ingot** certificates.

So where is the program code?

Javascript runs in a web page so click here [2] to call up the Ship game in your browser.

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');



Under view you will find a menu item to look at the source code for your page. It might be called something a little different in different browsers, the example shown is Firefox.

Right-clicking the mouse on the link [2] to the page will bring up a menu and the item "**View Source**" must be selected (the text can differ slightly according to the browser used; the text "**View Source**" refers to the Internet Explorer browser; the Mozilla Firefox uses the text "**View Page Source**").

The text is marked using standard **Windows OS** commands (e.g. the keyboard shortcut **CTRL+A**) and copied into the clipboard (e.g. the keyboard shortcut **CTRL+C**). Any text editor capable of processing the plain text is opened (e.g. the program **Notepad**, which is part of the Windows OS) and the text is pasted into it from the clipboard (e.g. the keyboard shortcut **CTRL+V**).

The text in the word editor is saved into a file. The file can have any name but must have an extension of "**html**" or "**htm**". HTM is the standard extension for files to be opened with an internet web browser. Web browsers simply convert the content of an HTML file to something we can see in a web page. As long as your browser knows where to find the HTML file it can display it as a page. The web address or URL tells the browser where to find the file. This is usually somewhere on a server on the internet but it can just be on your local hard drive.

Another essential file is "**wordlist.js**", which contains a database of the words the user has to guess and the help text. The file can be downloaded by deleting the "**ship.htm**" in the browser address bar and replacing it with "**wordlist.js**". The browser then looks for the file and because it is not HTML it will assume you want to download it.

After pressing the key "**Enter**" on the keyboard, a dialog box will prompt you to **save a file**. The file must be saved under the **same name** and extension and into the **same directory** containing the

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

previous file. This is so that the code in the Javascript matches the names and places correctly.

Images are the last essential part of the game.

The list of images is as follows:

ship0.png

ship0.png

ship1.png

ship2.png

ship3.png

ship4.png

ship5.png

ship6.png

ship7.png

(these are the images of each step displaying the sinking ship)

pozadi.jpg

(this image is the background template)

The images can be downloaded in a similar manner as the "**wordlist.js**" file, but must be saved in the subdirectory named "**images**", which must be created.

This is again so that the names in the code match the places where the information is held.

To check every thing works try it out by double clicking on the file "**ship.htm**" file.

Changing the Words in the game

The file "**wordlist.js**" contains the words we need for the game. It is saved in the variable called "**Wordlist**" in the Javascript program. A variable is a store for some information that can change. In this case the wordlist can be varied to make different games to help learn different sets of words. The words must always be separated by a space and the whole list must be in quotation marks.

(example: **WordList="antivirus audiocity"**)

So try making a game with words that are related to science or a game with words related to maths or if you are really ambitious, words from a different language.

Changing images

The pictures used in the game can be replaced with your own. The game background can be modified and the individual stages of the sinking ship as well. You could have a different type of ship, for example.

Here are some rules that must be obeyed!

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

- the replacement image files must have the identical **name** and **extension**

- the images "**ship0.png**" to "**ship7.png**"must have a transparent background. This is why .png images are used. .png files can have transparency (some old web browsers might not show this). .jpg files don't have transparency but they can be compressed more to save space and enable them to load quicker.

- the game background is only a template, which consists of its repeated parts

- the new images must have the identical size in pixels to the originalif th sinking is to work properly

More advanced changes!

The changes above do not require any knowledge of the Javascript programming language. Wit even a little knowledge you can do quite a lot more and you don't need to learn everything before being able to do useful things.

Names of images, files and places where they are stored

The program code in the file ship.htm contains statements that use the names of files and the places they are stored. You can find them by opening the file ship.htm in a text editor and searching for the word. Programmers call words and similar series of letters and numbers, strings so you might say search for a string as it might be more than one word or just a group of letters.

The name of the background image used in the ship puzzle is given in the following HTML command:

```
<BODY bgcolor=#00FFFF background="images/pozadi.jpg">
```

```
See if you can find this string in the file.
```

If you wanted to use a different name for your background image you would have to search for all the times pozadi.jpg occurs in the file ship.htm and replace it with the new name. This is quick and easy in a text editor. Check that the string pozadi only occurs once in the file.

So to change the background image you can change the file pozadi.jpg in the directory images to say a file called "Sailing Boat.jpg and change the line of code in the ship.htm
from background="images/pozadi.jpg" to background="images/Sailing Boat.jpg". Save your changes and you should have an identical game with a new background.

(Hint: Always keep a copy of the original file in case you make a change and something unexpected happens!)

The names of the sinking ship images are stored in a special type of variable container, an Array. An array is a list of items and the code for it in Javascript is Array(N) where N is the number of itms in the list. Since the list starts at 0 and goes 0, 1, 2, 3, Array (4) has 4 spaces in the list Array(0), Array(1), Array(2) and Array(3)

In our "Ship" Javascript the Array to store the list of picture images is Array (8) because there are 8 images of the ship sinking.

Try searching your file for Array (8)

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

You should find it as part of the following line of code

```
Pic = new Array(8);
```

This is telling the program we want it to set up a new array called Pic with 8 spaces in the list. (0 ----> 7)

So, the array Pic will simply be a set of spaces somewhere in the computer's memory ready to store 8 pictures. At this point the program is just making the links to the pictures so it can find them and use them. The pictures could be of anything at all but since this game is about a sinking ship the pictures used are those of a ship in 8 different positions of sinking.

Some programming

If we wanted to change the number of images, for example, to give more attemps and make the ship sink more gradually, things get a bit more interesting. We need a way of telling the web page to take a number of images fom the image file and display them on the background each time the player asks for another letter.

The program code in the next two lines is called a loop and it is used for doing tasks that repeat.

```
for (n=0; n<8; n++)
```

```
{ Pic[n] = new Image(); Pic[n].src = "images/ship" +n+ ".png"; }
```

It is saying start a counter at 0. (n=0)

Next do the instructions in the { } using the current value of n and then check to see if n is smaller than 8 and if so make it one bigger. Do the instructions in {} using the current value of n which is now 1 and carry on repeating this until n gets to 8.

The part of the statement in the { } Pic[n].src ="images/ship"+n+".png"; is telling the web page to go to the directory images and get the file shipn.png. shipn.png will be ship0.png, ship1.png, etc as th program goes round the loop. The first part of the statement is storing each image in the array ready for use.

To change the number of images we would have to change the number 8 in the Array and then also in the program.

Comments in the program

As mentioned earlier, the game is written using the Javascript and HTML commands. Brief help descriptions of the functions and elements, and the descriptions of the variables are stated directly in the code (in the comment lines). In order to understand the function, basic knowledge of the Javascript and HTML programming is necessary.

(example:

```
/*
```

```
wordlist - variable in file wordlist.js containing list of words
```

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

```
*/
```

)

On the line 9 in the "**ship.htm**" is the following command:

```
<script language="JavaScript" src="wordlist.js" type="text/javascript"></script>
```

The command contains instructions and procedures of "**wordlist.js**" used for initialization of the "**ship.htm**" file.

More experienced programmers can use a different file name and they must change it in this command.

The following procedure monitors the events called by pressing the keyboard keys:

```
<script for=document event="onkeydown()" language="JScript">

//Event Keydown return keycode

if (window.event) KeyDown(window.event.keyCode);

</script>
```

The following procedure provides recognition of the correct letter and its clicking in the ASCII table:

```
// detection pressed key

function KeyDown(whichkey)

{ if (IsOver) return;

if ((whichkey>64)&&(whichkey<91)) Click(String.fromCharCode(whichkey));

if ((whichkey>96)&&(whichkey<123)) Click(String.fromCharCode(whichkey-32));

}
```

The following list contains the variables and the loop that will read the images into the field:

```
/*

m - start initial variable for cycle

m - end initial variable for cycle (number of attempts is <8)

IsOver - true if game is over

Moves - actual attempt
```

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

```
Letters - guessed letters

Word - word which is displayed

StartTime - start time

EndTime - end time

*/

var n, m, IsOver=true, Moves, Letters, Word, StartTime, EndTime;

// Pic - field containing images

Pic = new Array(8);

for (n=0; n<8; n++)

{ Pic[n] = new Image(); Pic[n].src = "images/ship"+n+".png"; }
```

The following function "**Init**" configures the initial variables, initializes the game, and randomly selects the word to be guesses:

```
function Init()

/*

ii - start initial variable for cycle

jj - variable displaying letters in cycle (A-Z is charcode between 65 - 90)

wordlist - variable in file wordlist.js containing list of words

*/

{ var ii, jj;

if (!IsOver)

{ if (! confirm("Do you want to give up the current game?"))

return;

}
```

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

-->

```
document.Options.attempt.value=7;

IsOver=false;

Moves=0;

Letters="";

for (i=65; i<=91; i++)

{ jj=String.fromCharCode(i);

document.Options[jj].value=jj;

}

do

{ do ii=Math.round(Math.random()*WordList.length);

while (WordList.charAt(ii)==" ");

while (WordList.charAt(ii-1)!=" ") ii--;

jj=ii+1;

while (WordList.charAt(jj)!=" ") jj++;

Word=WordList.substring(ii, jj).toUpperCase();

}

while ((Word.length<5)||(Word.length>9))

var ii, ss=" ";

for (i=0; i<Word.length; i++) ss+="_ ";

document.Options.TWord.value=ss;

document.images[0].src=Pic[Moves].src;

Now = new Date();

StartTime = Now.getTime() / 1000;
```

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

```
document.Options.TWord.focus();

document.Options.TWord.blur();

}
```

The following function "**Click**" provides recognition of the letter being pressed, it displays the letter if contained in the word, and it displays the necessary results if the word is guessed correctly or the number of attempts is exhausted:

```
//action after click on any button

function Click(cc)

/*

ii - start initial variable for cycle

ss - variable displaying actual result in button Tword of guess word

*/

{ var ii, ss=" ";

document.Options.TWord.focus();

document.Options.TWord.blur();

if (isOver) return;

if (Letters.indexOf(cc)>=0) return;

document.Options[cc].value=" ";

Letters+=cc;

if (Word.indexOf(cc)>=0)

{ for (ii=0; ii<Word.length; ii++)

{ if (Letters.indexOf(Word.charAt(ii))>=0) ss+=Word.charAt(ii)+" ";

else ss+="_ ";

}
```


(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

```
document.Options.TWord.value=sx;

if (sx.indexOf("_")<0)

{ IsOver=true;

Now = new Date();

EndTime = Now.getTime() / 1000;

i=Math.floor(EndTime - StartTime);

if (window.opener)

{ if (window.opener.SetHighscores)

window.opener.SetHighscores("Hangman",'',i,-1);

}

if (confirm("Super, you solved this game in "+i+" seconds !\nPlay again ?")) Init();

}

}

else

{ document.Options.attempt.value--;

Moves++;

document.images[0].src=Pic[Moves].src;

if (Moves==7)

{ IsOver=true;

alert("You lost !!\nThe word was "+Word+".");

}

}
```


(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

```
}
```

The following HTML code displays the own game elements and graphics and configures the events during the mouse clicks:

```
</head>

<BODY bgcolor=#00FFFF background="images/pozadi.jpg">

<form name="Options">

<p class="uvodni">

Number of attempts is: 

<!- attempt - name of button displaying number off attempts ->

<input type=button value="7" width=15 style="width:15;background-color:#FFFFFF" name="attempt">

<!- napoveda - name button diplaying link on help ->

<input type=button value="Help" OnClick="javascript:DispHelp(Word)" width=35 style="width:35;background-color:#FFFFFF" name=napoveda">

<a href="teachers_guide.htm" target="_blank">Teachers Guide</a>

</p>

<p ALIGN=center>

<table noborder width=100% cellpadding=0 cellspacing=0>

<tr>

<td align=left valign=top>

<table noborder width=102 height=50 cellpadding=0 cellspacing=0>

<tr><td align=center valign=middle>

<!- Tword - name of button displaying guessed word ->

<input type=button value="_ _ _ _ _ _ _" width=125 style="width:180;background-color:#FFFFFF;font-size:20" name="TWord"></td></tr>
```


(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

```
</table>

</td>

<td align=center rowspan=2>

<table noborder cellpadding=0 cellspacing=0>

<tr><td><img src="images/ship0.gif" border=0></td></tr>

</table>

</td>

</tr>

<tr><td align valign=top>

<table noborder width=192 cellpadding=0 cellspacing=0>

<tr><td align=center valign=top><table border cellpadding=0 cellspacing=4><tr><td>

<!- Buttons displaying letters of alphabet ->

<input type="button" name="A" value="A" onClick="Click('A')" width=20 style="width:20"></td><td>

<input type="button" name="B" value="B" onClick="Click('B')" width=20 style="width:20"></td><td>

<input type="button" name="C" value="C" onClick="Click('C')" width=20 style="width:20"></td><td>

<input type="button" name="D" value="D" onClick="Click('D')" width=20 style="width:20"></td><td>

<input type="button" name="E" value="E" onClick="Click('E')" width=20 style="width:20"></td><td>

<input type="button" name="F" value="F" onClick="Click('F')" width=20 style="width:20"></td><td>

<input type="button" name="G" value="G" onClick="Click('G')" width=20 style="width:20"></td></tr><tr><td>

<input type="button" name="H" value="H" onClick="Click('H')" width=20 style="width:20"></td><td>

<input type="button" name="I" value="I" onClick="Click('I')" width=20 style="width:20"></td><td>

<input type="button" name="J" value="J" onClick="Click('J')" width=20 style="width:20"></td><td>
```

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

-->

```
<input type="button" name="K" value="K" onClick="Click('K')" width=20 style="width:20"></td><td>

<input type="button" name="L" value="L" onClick="Click('L')" width=20 style="width:20"></td><td>

<input type="button" name="M" value="M" onClick="Click('M')" width=20 style="width:20"></td><td>

<input type="button" name="N" value="N" onClick="Click('N')" width=20 style="width:20"></td></tr><tr><td>

<input type="button" name="O" value="O" onClick="Click('O')" width=20 style="width:20"></td><td>

<input type="button" name="P" value="P" onClick="Click('P')" width=20 style="width:20"></td><td>

<input type="button" name="Q" value="Q" onClick="Click('Q')" width=20 style="width:20"></td><td>

<input type="button" name="R" value="R" onClick="Click('R')" width=20 style="width:20"></td><td>

<input type="button" name="S" value="S" onClick="Click('S')" width=20 style="width:20"></td><td>

<input type="button" name="T" value="T" onClick="Click('T')" width=20 style="width:20"></td><td>

<input type="button" name="U" value="U" onClick="Click('U')" width=20 style="width:20"></td></tr><tr><td>

<input type="button" name="V" value="V" onClick="Click('V')" width=20 style="width:20"></td><td>

<input type="button" name="W" value="W" onClick="Click('W')" width=20 style="width:20"></td><td>

<input type="button" name="X" value="X" onClick="Click('X')" width=20 style="width:20"></td><td>

<input type="button" name="Y" value="Y" onClick="Click('Y')" width=20 style="width:20"></td><td>

<input type="button" name="Z" value="Z" onClick="Click('Z')" width=20 style="width:20"></td><td colspan=2>

<input type="button" value="NEW" onClick="Init()" width=48 style="width:48"></td></tr></table>

</td></tr>

</form>
```

The following code calls the initialization function of the game:

```
<script language="JavaScript">

Init();

</script>
```

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

```
</BODY>

</HTML>
```

Variable Wordlist in file **wordlist.js** contains list of guessed words.

```
//list of used words in game

WordList=" antivirus audacity computer copyright display drive file firefox flash hardware inkscape internet keyboard license linux malware memory monitor mouse openoffice opensource owner password platform printer processor safety scanner security smartphone software standard storage touchpad trackball virus windows";
```

Function Disphelp in file **wordlist.js** displays window with help text, which descript guessed word.

```
//display window with help text

function DispHelp(sel_word)

{

odkud="\n\nsource: From Wikipedia, the free encyclopedia";

//sel_word - displayed word

switch (sel_word)

{

case "ANTIVIRUS":

alert("This software is used to prevent, detect, and remove malware, including computer viruses, worms, and trojan horses. Such programs may also prevent and remove adware, spyware, and other forms of malware."+odkud);

break;

case "AUDACITY":

alert("There is a digital audio editor and recording application. The application is free, open source and cross-platform, available for Windows, Mac OS X, Linux and BSD."+odkud);

break;
```

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

```
case "COMPUTER":

alert("A machine that manipulates data according to a set of instructions."+odkud);

break;

case "COPYRIGHT":

alert("There is a form of intellectual property that gives the author of an original work exclusive right for a certain time period in relation to that work, including its publication, distribution and adaptation, after which time the work is said to enter the public domain."+odkud);

break;

case "DISPLAY":

alert("The device is an output device for presentation of information for visual, tactile or auditive reception, acquired, stored, or transmitted in various forms. When the input information is supplied as an electrical signal, the display is called as electronic."+odkud);

break;

case "DRIVE":

alert("There is a peripheral device used to record and retrieve information. Main implementations are hard disks, floppy disks and optical discs."+odkud);

break;

case "FILE":

alert("There is a block of arbitrary information, or resource for storing information, which is available to a computer program and is usually based on some kind of durable storage."+odkud);

break;

case "FIREFOX":

alert("There is a free and open source web browser descended from the Mozilla Application Suite and managed by Mozilla Corporation."+odkud);

break;

case "FLASH":

alert("There is a multimedia platform originally acquired by Macromedia and currently developed and distributed by Adobe Systems. Since its introduction in 1996, there has become a popular method for adding animation and interactivity to web pages."+odkud);

break;

case "HARDWARE":
```

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

-->

```
alert("There is a general term for the physical artifacts of a technology. It may also mean the physical components of a computer system."+odkud);

break;

case "INKSCAPE":

alert("There is a vector graphics editor application. It is distributed under a free software license, the GNU GPL."+odkud);

break;

case "INTERNET":

alert("There is a global system of interconnected computer networks that use TCP/IP to serve billions of users worldwide."+odkud);

break;

case "KEYBOARD":

alert("There is an input device, which uses an arrangement of buttons or keys, to act as mechanical levers or electronic switches."+odkud);

break;

case "LICENSE":

alert("It means to give permission. There may be granted by a party ('licensor') to another party ('licensee') as an element of an agreement between those parties."+odkud);

break;

case "LINUX":

alert("There is a generic term referring to Unix-like computer operating systems based on the Linux kernel. Their development is one of the most prominent examples of free and open source software collaboration; typically all the underlying source code can be used, freely modified, and redistributed, both commercially and non-commercially, by anyone under licenses such as the GNU GPL."+odkud);

break;

case "MALWARE":

alert("Short for malicious software, is software designed to infiltrate or damage a computer system without the owner's informed consent."+odkud);

break;

case "MEMORY":

alert("There is an ability to store, retain, and recall information."+odkud);

break;
```

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

```
case "MONITOR":

alert("There is a piece of electrical equipment which displays images generated by devices such as computers, without producing a permanent record."+odkud);

break;

case "MOUSE":

alert("There is a pointing device that functions by detecting two-dimensional motion relative to its supporting surface."+odkud);

break;

case "OPENOFFICE":

alert("There is an open source software application suite available for a number of different computer operating systems. It is distributed as free software and written using its own GUI toolkit."+odkud);

break;

case "OPENSOURCE":

alert("There is computer software for which the source code and certain other rights normally reserved for copyright holders are provided under a software license that meets the Definition or that is in the public domain."+odkud);

break;

case "OWNER":

alert("He/she is related to ownership, the state or fact of exclusive rights and control over property, which may be an object, land/real estate or intellectual property."+odkud);

break;

case "PASSWORD":

alert("There is a secret word or string of characters that is used for authentication, to prove identity or gain access to a resource."+odkud);

break;

case "PLATFORM":

alert("That describes some sort of hardware architecture or software framework (including application frameworks), that allows software to run."+odkud);

break;

case "PRINTER":
```

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

-->

```
alert("Peripheral which produces a hard copy (permanent readable text and/or graphics) of documents stored in electronic form, usually on physical print media such as paper or transparencies."+odkud);

break;

case "PROCESSOR":

alert("There is a system which processes data which has been captured and encoded in a format recognizable by the data processing system or has been created and stored by another unit of an information processing system."+odkud);

break;

case "SAFETY":

alert("There is the state of being 'safe', the condition of being protected against physical, social, spiritual, financial, political, emotional, occupational, psychological, educational or other types or consequences of failure, damage, error, accidents, harm or any other event which could be considered non-desirable."+odkud);

break;

case "SCANNER":

alert("There is a device that optically scans images, printed text, handwriting, or an object, and converts it to a digital image."+odkud);

break;

case "SECURITY":

alert("There is the degree of protection against danger, loss, and criminals."+odkud);

break;

case "SMARTPHONE":

alert("There is a mobile phone offering advanced capabilities, often with PC-like functionality."+odkud);

break;

case "SOFTWARE":

alert("There is a general term used to describe the role that computer programs, procedures and documentation play in a computer system."+odkud);

break;

case "STANDARD":

alert("There is an established norm or requirement. It is usually a formal document that establishes uniform engineering or technical criteria, methods, processes and practices."+odkud);

break;
```

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');

```
case "STORAGE":

alert("There is a device for recording (storing) information (data)."+odkud);

break;

case "TOUCHPAD":

alert("There is a pointing device consisting of specialized surface that can translate the motion and position of a user's fingers to a relative position on screen."+odkud);

break;

case "TRACKBALL":

alert("There is a pointing device consisting of a ball held by a socket containing sensors to detect a rotation of the ball about two axes-like an upside-down mouse with an exposed protruding ball."+odkud);

break;

case "VIRUS":

alert("There is a computer program that can copy itself and infect a computer."+odkud);

break;

case "WINDOWS":

alert("There is a series of software operating systems and graphical user interfaces produced by Microsoft."+odkud);

break;



}

}
```

Source URL: https://theingots.org/community/puzzles

Links
[1] http://ingotgames.org/ship/en/ship.htm
[2] http://www.euro-face.cz/javascript/ship/ship.htm

(function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){ (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o), m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m) })(window,document,'script','//www.google-analytics.com/analytics.js','ga'); ga('create', 'UA-46896377-2', 'auto'); ga('send', 'pageview');