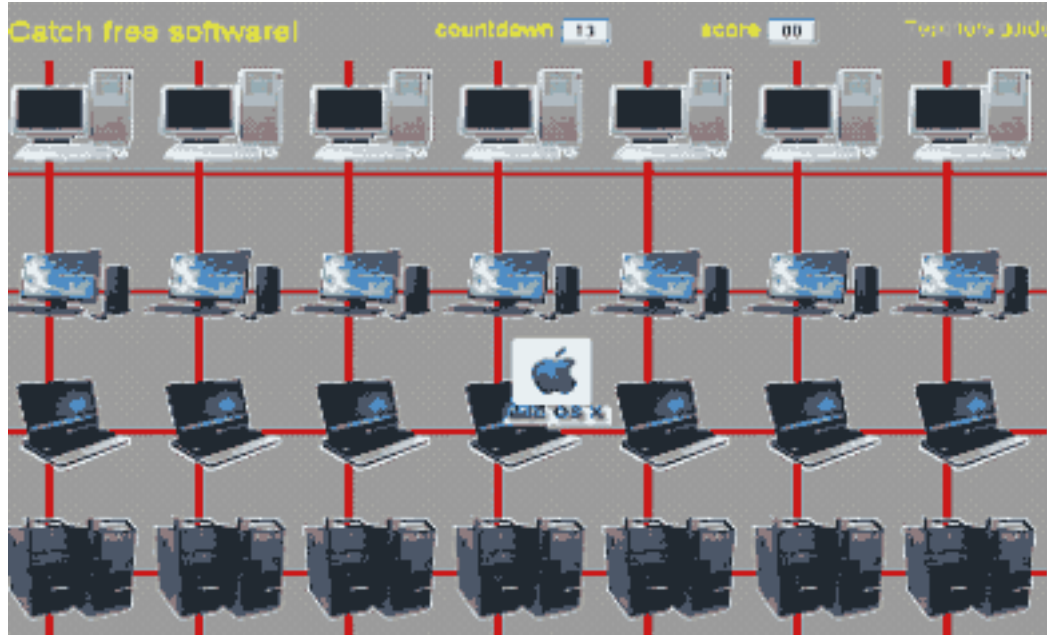


Learning Javascript using the game CATCH FOSS

This page describes the game **CATCH FOSS**



The game **Catcher** is written in **JavaScript** and standard **HTML** commands.

Learning Javascript using the game CATCH FOSS

-->



The player's objective is to make a correct distinction between the free and paid software. The objects appearing on the screen bear the software name and logo and move around in various speeds. The player will receive 10 points after clicking on the correct object (the player successfully recognizes the free software). The player will lose 10 points after clicking on the paid software (the player fails to recognize the paid software). The player can receive a maximum of 50 points as a bonus by clicking on the **"Bonus"** object. The player can also lose 50 points by clicking on the **"Virus"** object.

The player can always see the score received during the play. The player can also see the number of the objects that remain to be displayed on the screen. The total number of the objects is 30.

The **"Teachers Guide"** button will show a brief description of the game and its purpose within the Ingot project.

Game Download

The game can be copied and modified according to your own skills: right clicking the mouse will bring up a menu. The **"View Source"** item on the menu must be selected (the text of the menu can
(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');

differ slightly according to the browser used; the text "**View Source**" refers to the Internet Explorer browser and the "**View Page Source**" is used by the Mozilla Firefox browser).

Mark the text using standard **Windows OS** commands (e.g. **CTRL+A**) and copy it into the clipboard (e.g. **CTRL+C**). Open any text editor capable of processing plain text (e.g. **Notepad**, which is part of Windows OS) and paste the text into the editor from the clipboard (e.g. **CTRL+V**).

Save the text displayed in the word editor as a file of any name and use either "**html**" or "**htm**" extension, which is the standard for the files to be opened in any internet browser.

Images are another essential part of the game and the list of images used is as follows:

L0.gif, L1.gif, L2.gif, L3.gif, L4.gif, L5.gif, R0.gif, R1.gif, R2.gif, R3.gif, R4.gif, R5.gif – software images

back.gif – background image

The images can be downloaded via changing the address in the browser bar: "**catch.htm**" is replaced with "**L0.gif**" as shown on the example below.

(e.g., change **www.ingotgames.org/catcher/en/catch.htm** to **www.ingotgames.org/catcher/en/images/L0.gif**)

After pressing "**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 previous file.

The image must be saved in the subdirectory named "**images**", which must be created beforehand.

Correct function of all downloaded components of the game can be verified by launching the "**catch.htm**" file.

Replacing Images

The game images can be replaced with your own ones. The game background can be modified under the following conditions:

- the image files must have an identical **name** and **extension** (more experienced programmers can use a different type of the image file, provided they modify its source code accordingly as described below)
- the game background is only a template, which consists of its repeated parts
- the game background must be transparent (description of this feature is outside of the scope of this manual and can be found on the web)
- the images must have an identical length and width measured in pixels (description of this feature is outside of the scope of this manual and can be found on the web)

Modification of the Total Number of the Objects to be Displayed

The number of the objects to be displayed can be modified via editing the code of "**catch.htm**".

The "**init**" function must be found(exact wording in the code: **function init()**).

"**NumberObj=30**" is a variable representing the number of the objects to be displayed.

In order to either shorten or prolong the game time, just replace the number 30 with another whole positive number.

Brief Description of the Game Functions in JavaScript Language

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

The following command is present on line 20 in "**catch.htm**":

```
<script src=laydiv.js></script>
```

The command contains the functions of "**laydiv.js**" to launch the "**catch.htm**".

More experienced programmer can use another file name and change it here.

The file contains functions for displaying layers used by the **HTML** or **XHTML** `<div></div>` tags.

The following codes are in the beginning of each file (for browser detection and correct display of the layers):

Learning Javascript using the game CATCH FOSS

-->

```
//detecting browser to dynamically write appropriate DIVs/LAYERS

N=(navigator.appName.indexOf( 'Netscape' )!=-1&&parseInt(navigator.appVersion)<5)
S=(navigator.appName.indexOf( 'Netscape' )!=-1&&parseInt(navigator.appVersion)>4.9)
M=(navigator.appName.indexOf( 'Microsoft' )!=-1)
var Styl,ID,zX,zY,vis,vs,zz
Vis=new Array()
Vis[0]=(M|S) ? "hidden" : "hide"
Vis[1]=(M|S) ? "visible" : "show"
```

The "**GetDiv**" and "**EndDiv**" functions provide the actual display of the layers.

```
//set parameters layer (tag <DIV>)
function GetDiv(divId,divY,divX,divW,divH,bCol,visb,zInd){
bkCol=(bCol!="")?(N)? " bgColor="+bCol:";background:"+bCol:""

Styl = (M|S) ? "<DIV" : "<LAYER"

if(M|S){
Styl+=" ID="+divId
Styl+=" style='position:absolute;top:"+divY+";left:"+divX+";width:"+divW+";height:"+divH+bkCol
Styl+=";visibility:"+Vis[visb]+";z-index:"+zInd
Styl+=">"
}

if(N){
Styl+=" ID="+divId
Styl+=" top="+divY+" left="+divX+" width="+divW+" height="+divH+bkCol
Styl+=" visibility="+Vis[visb]+" z-index="+zInd
Styl+=">"
}
document.writeln(Styl)
}

//end tag layer (tag <DIV>)
function EndDiv(){
(M|S)? document.writeln("</DIV>"): document.writeln("</LAYER>")
}
```

The "**PutIt**" function sets the layer coordinates.

```
//set X and Y axis of layer
function PutIt(ID,zX,zY){
if(N){
document.layers[ID].left=zX
document.layers[ID].top=zY
}
```

The "**Zindex**" function sets the Z-Index of the layer, which provides the overlay of the individual layers over each other (description of this feature is outside of the scope of this manual and can be found on the web).

```
//set Zindex tag <DIV>
function Zind(ID,zz){
if(N){
document.layers[ID].zIndex=zz
}
if(M){
document.all[ID].style.zIndex=zz
}
if(S){
document.getElementById(ID).style.zIndex=zz
}
}
```

The "**Xof**" and "**Yof**" functions return the value of the X and Y layer coordinate.

```
//return X axis layer
function Xof(ID){
if(N){
return parseInt(document.layers[ID].left)
}
if(M){
return parseInt(document.all[ID].style.left)
}
if(S){
return parseInt(document.getElementById(ID).style.left)
}
}
//return Y axis layer
function Yof(ID){
if(N){
return parseInt(document.layers[ID].top)
}
if(M){
return parseInt(document.all[ID].style.top)
}
```

Below is the HTML code of "**catch.htm**" containing the head, style formatting elements and table formatting elements necessary for displaying the game background, text and links.

```
<HTML>
<head>
<META NAME="AUTHOR" CONTENT="">
<META NAME="description" CONTENT="Javascripts Games">
<META NAME="keywords" CONTENT="games, javascript, graphics">
<META NAME="robots" CONTENT="INDEX, ALL">
<link rel="stylesheet" type="text/css" href="css/game.css" />
<title>Catcher</title>
```

```
(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');
```

Learning Javascript using the game CATCH FOSS

```
-->

<meta http-equiv="Content-
Type" content="text/html; charset=windows-1250"><style type="text/css">
<!--
body,td,th {
  font-family: Arial, Helvetica, sans-serif;
}
body {
  background-color: #999;
}
-->
</style></head>

<script src=laydiv.js></script>
<body text="#FFFF00" style="cursor:cross-hairs">
<form name=Form>
<table width=62%>
<tr>
<td width="37%" height="59" align="left" valign="top"><font size=5>Catch free softwar
e!
</td>
<td width="25%" align="center" valign="top"><font size="+1">countdown</font>
  <input type="button" name=Countdown size=5>
<td width="20%" align="center" valign="top"><font size="+1">score</font>
  <input type="button" name=Playscore size=5></td>
<td width="18%" align="right" valign="top"><a href="teachers_guide.htm" target="_blan
k">Teachers guide</a>
</td>
</table>
```

The JavaScript code below sets the game background to alternate between the front and behind of the moving object.

```
<script>
Z=new Array()
Z[0]=new Array(0,0,0,0,0)
Z[1]=new Array(0,3,0,3,0)
Z[2]=new Array(0,0,3,0,3)
Z[3]=new Array(3,0,3,0,0)
Z[4]=new Array(0,0,0,3,3)
Z[5]=new Array(0,0,3,0,0)
//Display Background
GetDiv("PozadiL",50,0,110,600,'#999',1,4)
document.write("<img src='images//back.gif' width=115 height=500>")
EndDiv()
for(c=0;c<5;c++){
GetDiv("Pozadi"+c,50,110+c*110,110,600,',',1,Z[1][c])
document.write("<img src='images//back.gif' width=115 height=500>")
EndDiv()
}
GetDiv("PozadiR",50,660,300,600,'#999',1,4)
document.write("<img src='images//back.gif' width=115 height=500>")
EndDiv()
```

The JavaScript code below is for the random selection of the moving objects including their speed.

```
Side=new Array()
Dir=new Array()
Dir["L"]=5
Dir["R"]=-5

//random select images
for(x=0;x<6;x++){
GetDiv("L"+x,200,5,50,50,'',1,2)
document.write("<a href='javascript:GetHim(\"",x,"')'><img src='images//L"+x+".gif' border=0></a>")
EndDiv()
GetDiv("R"+x,200,660,50,50,'',1,2)
document.write("<a href='javascript:GetHim(\"",x,"')'><img src='images//R"+x+".gif' border=0></a>")
EndDiv()
Side[x]=(Math.random(>.5)?"L":"R")
}
/* Game initialization
   NumberObj - number of objects to end game
   Score - score
   Timer - actual number of objects which were displayed
*/
function Init(){
Timer=0
Score=0
NumberObj=30
document.Form.Playscore.value=Score
Next()
}
```

The **"Next"** function provides the display of the next object.

```
Side=new Array()
Dir=new Array()
Dir["L"]=5
Dir["R"]=-5

//random select images
for(x=0;x<6;x++){
GetDiv("L"+x,200,5,50,50,'',1,2)
document.write("<a href='javascript:GetHim(\"",x,"')'><img src='images//L"+x+".gif' border=0></a>")
EndDiv()
GetDiv("R"+x,200,660,50,50,'',1,2)
document.write("<a href='javascript:GetHim(\"",x,"')'><img src='images//R"+x+".gif' border=0></a>")
EndDiv()
Side[x]=(Math.random(>.5)?"L":"R")
}
/* Game initialization
   NumberObj - number of objects to end game
   Score - score
```

```
    Timer - actual number of objects which were displayed
*/
function Init(){
Timer=0
Score=0
NumberObj=30
document.Form.Playscore.value=Score
Next()
}
```

The "**Next**" functions calls the "**Slide**" function, which provides the actual configuration of the coordinates, appearance and time delay after the previous object stops moving.

```
//Display object on screen
function Slide(){
X=Xof(Obj)+Math.round(dir*(Si/2+1))
PutIt(Obj,X,Y)
if(X>0&&X<670)SlideOut=setTimeout('Slide()',50-Timer)
else {
Side[S]=(Side[S]=="L")?"R":"L"
setTimeout('Next()',700)
```

The "**GetHim**" function increases or decreases the game score or adds the number of the objects after the applicable object is selected.

```
// Increase or decrease Score
function GetHim(i){
PutIt("L"+i,5,200)
PutIt("R"+i,660,200)
if (dir==5)
{
switch (i)
{
case 0:
Score+=10;break;
case 1:
Score-=10;break;
case 2:
Score+=10;break;
case 3:
Score-=10;break;
case 4:
Score+=10;break;
case 5:
Score-=10;break;
}
}
else
{
switch (i)
{
case 0:
```


Learning Javascript using the game CATCH FOSS

-->

```
        Score+=10;break;
    case 1:
        Score-=10;break;
    case 2:
        Score-=10;break;
    case 3:
        Score-=50;break;
    case 4:
        Score+=50;break;
    case 5:
        NumberObj+=10;
        document.Form.Countdown.value=parseInt(NumberObj-Timer)+1;
        Score+=50;
        break;
    }
}
// Score+=10*(i+5)
Side[i]=(Side[i]=="L")?"R":"L"
document.Form.Playscore.value=Score
clearTimeout(SlideOut)
Next()
}
```

The "**GameOver**" function provides the display of the final score and command for another game session.

```
//Display window after Game over function GameOver(){ if(confirm("Game Over!\n\nScore :
"+Score+"\nPlay again?"))location="catch.htm" else location="end.htm" } if(confirm("Ready?"))Init()
else location="end.htm" </script>
```

Below is the standard ending of the file in HTML language.

```
</body> </HTML>
```

Source URL: <https://theingots.org/community/catchfoss>