Google Blockly積木撰寫

WBBIT教育版&BLOCKLYDUINO

講師：鳳山科技中心 傅仲儀主任

2020/8/30 20:00

自訂積木撰寫課程表

08:30-10:00	自訂積木安裝與架構說明
10:00-11:30	Blockly Developer Tools
11:30-12:00	多國語系製作
13:00-13:30	javascript.js常用技巧
13:30-15:30	自訂積木實作

研習講義與檔案下載

PDF簡報檔

https://github.com/fustyles/Workshop/blob/master/Webbit教育版&Blocklyduino自訂積木撰寫.pdf

PPT簡報檔

https://drive.google.com/drive/folders/1Q6I_saFnBYjrw_sCDfQ59W6mjEUI_zd2

上課檔案

https://github.com/fustyles/Workshop/blob/master/2020.8.19_blockly.zip

Google Blockly 學習資源

Blockly Google Developers
https://developers.google.com/blockly

Blockly討論區
https://groups.google.com/g/blockly

Blockly函式庫
https://developers.google.com/blockly/reference/overview

Blockly原始碼離線包
https://github.com/google/blockly/zipball/master

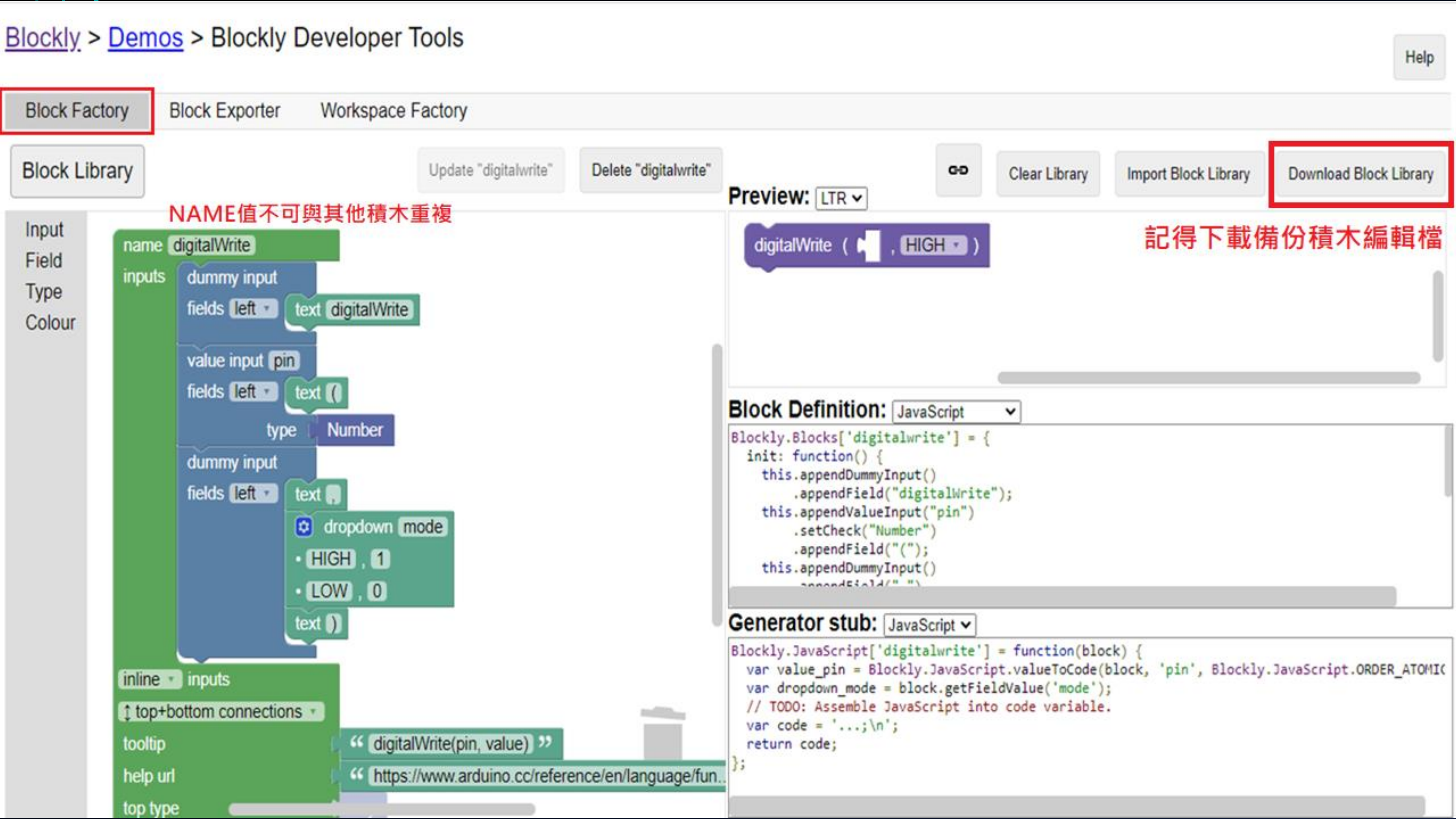
Webduino官方撰寫積木教學文件
https://www.facebook.com/groups/webduino/permalink/1536223946446669/

Blockly Developer Tools (Block Factory)

Blockly Developer Tools (匯出blocks.js, javascript.js, toolbox.xml)
https://blockly-demo.appspot.com/static/demos/blockfactory/index.html

使用教學
https://developers.google.com/blockly/guides/create-custom-blocks/blockly-developer-tools

教學影片
https://www.youtube.com/watch?time_continue=705&v=s2_xaEvcVI0&feature=emb_logo



NAME值不可與其他積木重複

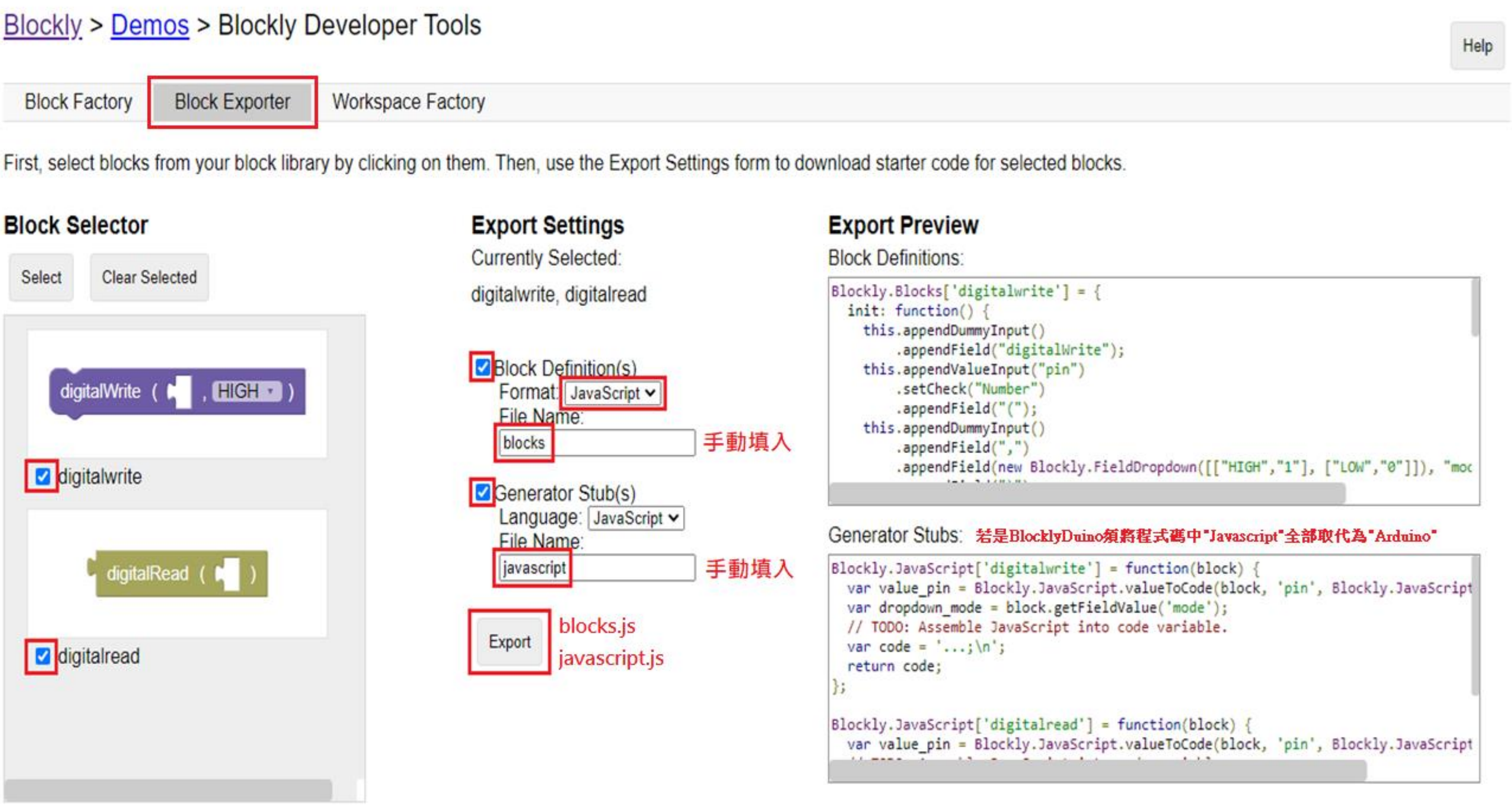
記得下載備份積木編輯檔

Block Definition: JavaScript

```
Blockly.Blocks['digitalwrite'] = {
  init: function() {
    this.appendDummyInput()
        .appendField("digitalWrite");
    this.appendValueInput("pin")
        .setCheck("Number")
        .appendField("(");
    this.appendDummyInput()
        .appendField(",")
```

Generator stub: JavaScript

```
Blockly.JavaScript['digitalwrite'] = function(block) {
  var value_pin = Blockly.JavaScript.valueToCode(block, 'pin', Blockly.JavaScript.ORDER_ATOMIC
  var dropdown_mode = block.getFieldValue('mode');
  // TODO: Assemble JavaScript into code variable.
  var code = '...;\n';
  return code;
};
```



Blockly > Demos > Blockly Developer Tools

Block Factory | Block Exporter | Workspace Factory

First, select blocks from your block library by clicking on them. Then, use the Export Settings form to download starter code for selected blocks.

Block Selector

☑ digitalwrite

☑ digitalread

Export Settings

Currently Selected:

digitalwrite, digitalread

☑ Block Definition(s)
Format: JavaScript
File Name:
blocks 手動填入

☑ Generator Stub(s)
Language: JavaScript
File Name:
javascript 手動填入

blocks.js
javascript.js

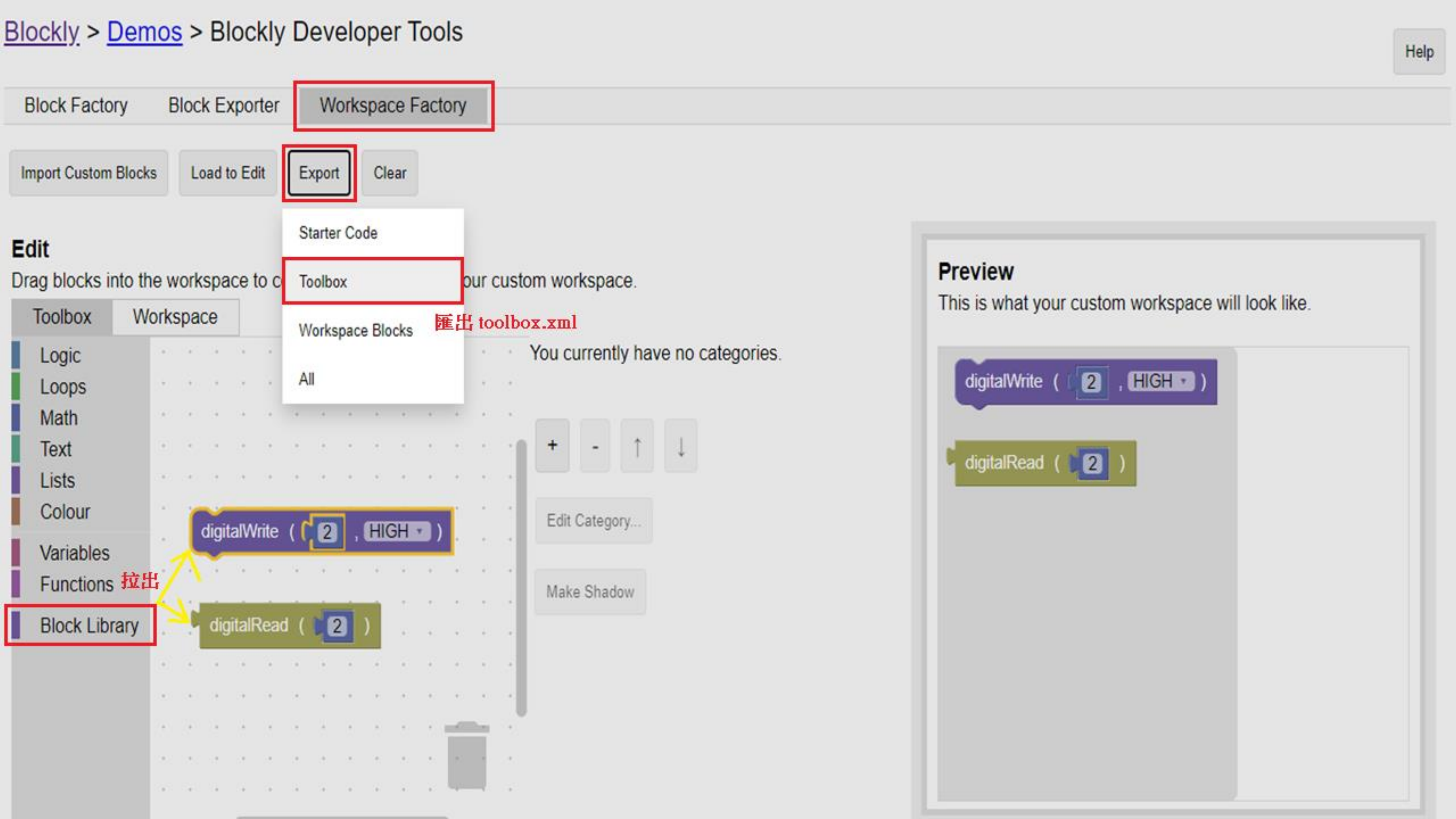
Export Preview

Block Definitions:

```
Blockly.Blocks['digitalwrite'] = {
  init: function() {
    this.appendDummyInput()
        .appendField("digitalWrite");
    this.appendValueInput("pin")
        .setCheck("Number")
        .appendField("(");
    this.appendDummyInput()
        .appendField(",")
        .appendField(new Blockly.FieldDropdown([["HIGH","1"], ["LOW","0"]]), "moc
```

Generator Stubs: 若是BlocklyDuino須將程式碼中"Javascript"全部取代為"Arduino"

```
Blockly.JavaScript['digitalwrite'] = function(block) {
  var value_pin = Blockly.JavaScript.valueToCode(block, 'pin', Blockly.JavaScript
  var dropdown_mode = block.getFieldValue('mode');
  // TODO: Assemble JavaScript into code variable.
  var code = '...;\n';
  return code;
};

Blockly.JavaScript['digitalread'] = function(block) {
  var value_pin = Blockly.JavaScript.valueToCode(block, 'pin', Blockly.JavaScript
```



Edit

Drag blocks into the workspace to c our custom workspace.

You currently have no categories.

Preview

This is what your custom workspace will look like.

匯出 toolbox.xml

拉出

Javascript 學習資源

免費編輯軟體 Notepad++

https://notepad-plus-plus.org/downloads/

Javascript 編輯器介紹

https://kknews.cc/zh-tw/code/nanm2ng.html

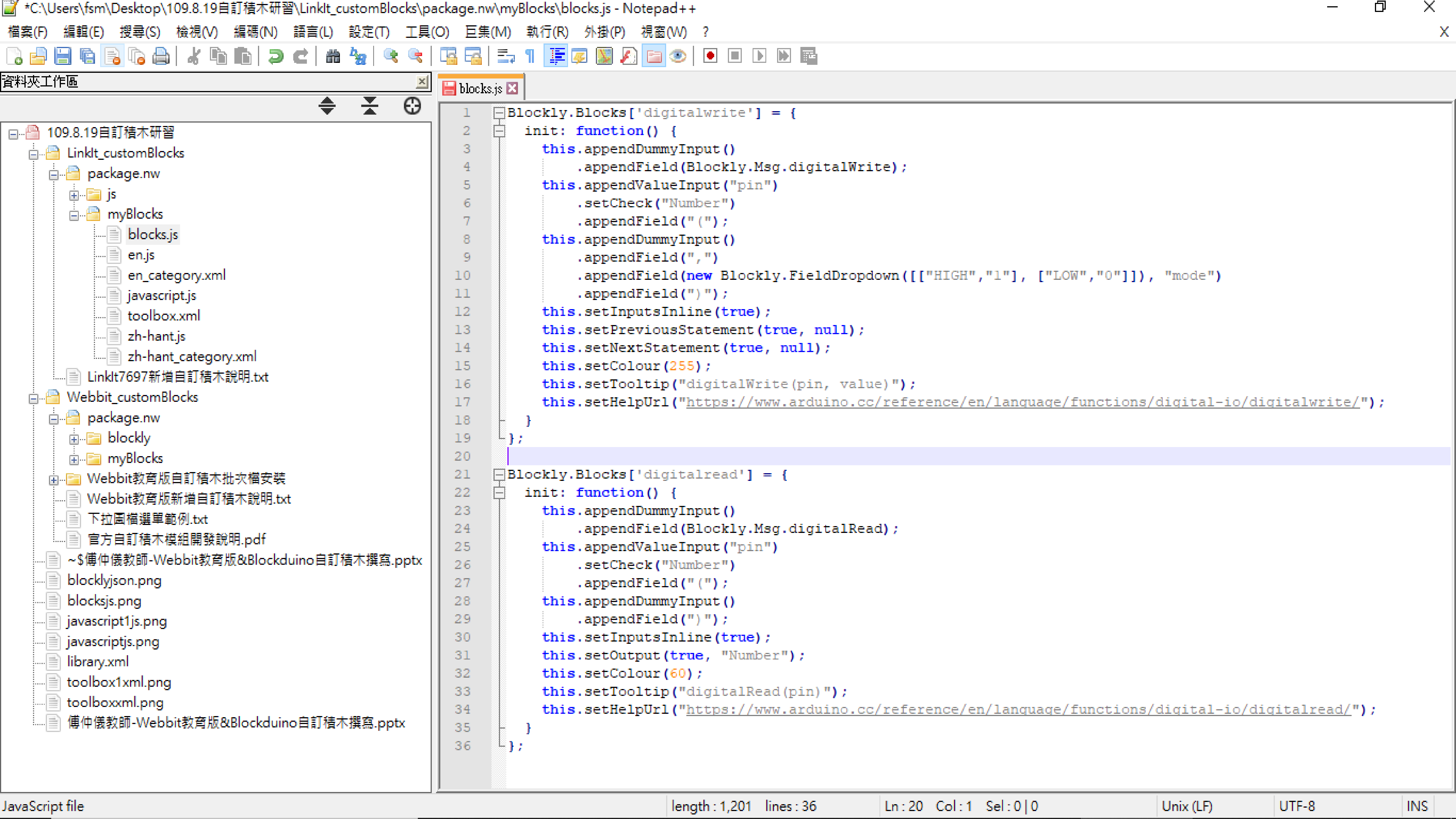
https://www.temok.com/blog/top-20-javascript-ide-source-code-editors/

JavaScript 教學文件

https://www.w3schools.com/js/

JavaScript Online Editor

https://www.w3schools.com/js/tryit.asp?filename=tryjs_myfirst


檔案(F) 編輯(E) 搜尋(S) 檢視(V) 編碼(N) 語言(L) 設定(T) 工具(O) 巨集(M) 執行(R) 外掛(P) 視窗(W) ?
資料夾工作區
blocks.js

*C:\Users\fsm\Desktop\109.8.19自訂積木研習\LinkIt_customBlocks\package.nw\myBlocks\blocks.js - Notepad++

```
109.8.19自訂積木研習
    LinkIt_customBlocks
        package.nw
            js
            myBlocks
                blocks.js
                en.js
                en_category.xml
                javascript.js
                toolbox.xml
                zh-hant.js
                zh-hant_category.xml
        LinkIt7697新增自訂積木說明.txt
    Webbit_customBlocks
        package.nw
            blockly
            myBlocks
        Webbit教育版自訂積木批次檔安裝
        Webbit教育版新增自訂積木說明.txt
        下拉選單範例.txt
        官方自訂積木模組開發說明.pdf
    ~$傅仲儀教師-Webbit教育版&Blockduino自訂積木撰寫.pptx
    blocklyjson.png
    blocksjs.png
    javascript1js.png
    javascriptjs.png
    library.xml
    toolbox1xml.png
    toolboxxml.png
    傅仲儀教師-Webbit教育版&Blockduino自訂積木撰寫.pptx
```

```javascript
Blockly.Blocks['digitalwrite'] = {
  init: function() {
    this.appendDummyInput()
        .appendField(Blockly.Msg.digitalWrite);
    this.appendValueInput("pin")
        .setCheck("Number")
        .appendField("(");
    this.appendDummyInput()
        .appendField(",")
        .appendField(new Blockly.FieldDropdown([["HIGH","1"], ["LOW","0"]]), "mode")
        .appendField(")");
    this.setInputsInline(true);
    this.setPreviousStatement(true, null);
    this.setNextStatement(true, null);
    this.setColour(255);
    this.setTooltip("digitalWrite(pin, value)");
    this.setHelpUrl("https://www.arduino.cc/reference/en/language/functions/digital-io/digitalwrite/");
  }
};

Blockly.Blocks['digitalread'] = {
  init: function() {
    this.appendDummyInput()
        .appendField(Blockly.Msg.digitalRead);
    this.appendValueInput("pin")
        .setCheck("Number")
        .appendField("(");
    this.appendDummyInput()
        .appendField(")");
    this.setInputsInline(true);
    this.setOutput(true, "Number");
    this.setColour(60);
    this.setTooltip("digitalRead(pin)");
    this.setHelpUrl("https://www.arduino.cc/reference/en/language/functions/digital-io/digitalread/");
  }
};
```

JavaScript file   length : 1,201   lines : 36   Ln : 20   Col : 1   Sel : 0 | 0   Unix (LF)   UTF-8   INS

Webduino自訂積木撰寫

Webbit教育版安裝

- 下載網址：Google搜尋"WebBitSetup.exe"

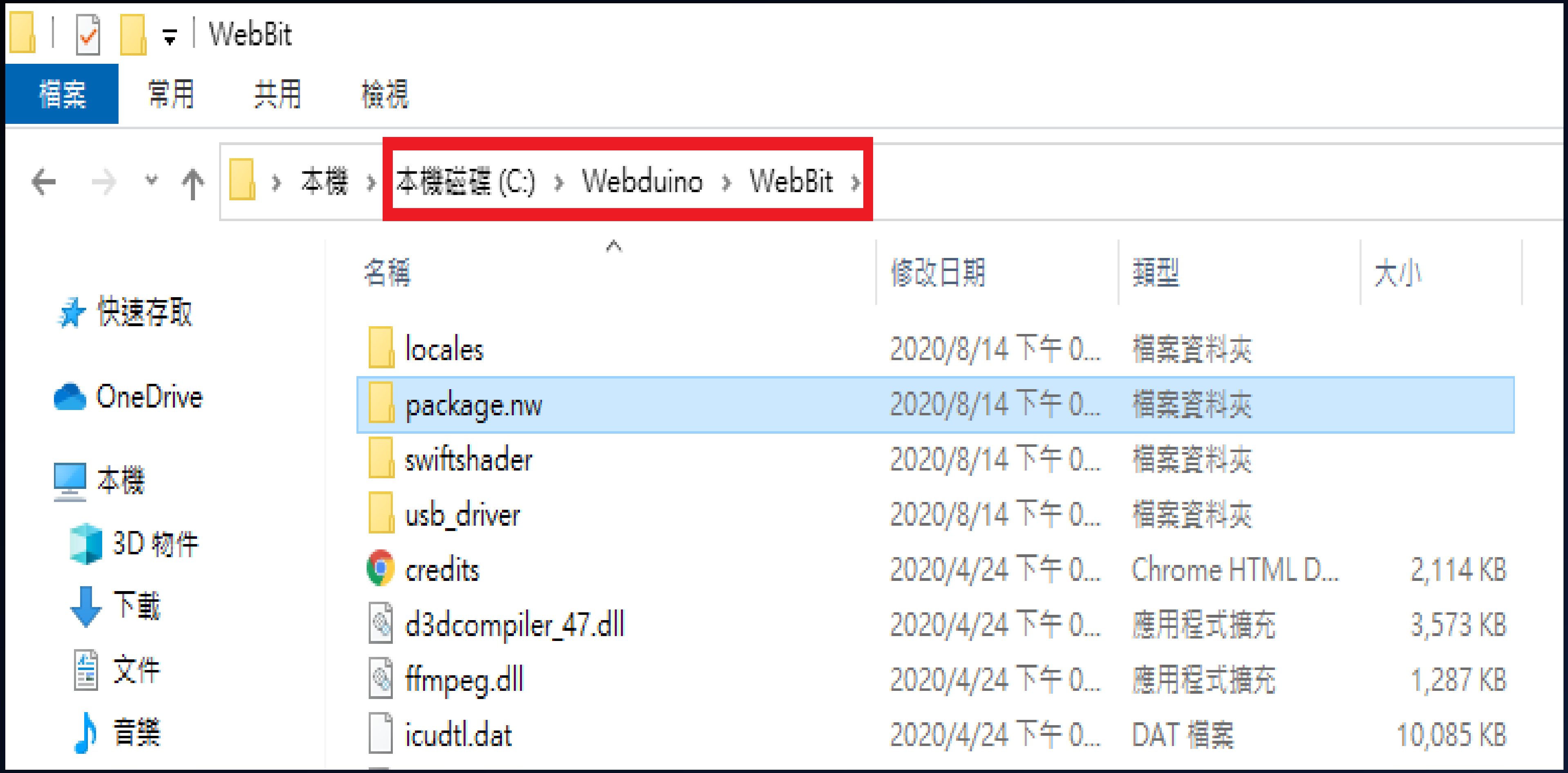
  離線版：https://ota.webduino.io/WebBitInstaller/WebBitSetup.exe

  網頁版：https://webbit.webduino.io/blockly/ (無法使用USB連線)
- 驅動程式：http://www.wch.cn/download/CH341SER_ZIP.html

  (離線版安裝已內建驅動程式)
- 更新韌體：離線版接上Webbit後可自動偵測線上更新

Webbit教育版新增自訂積木(法一)

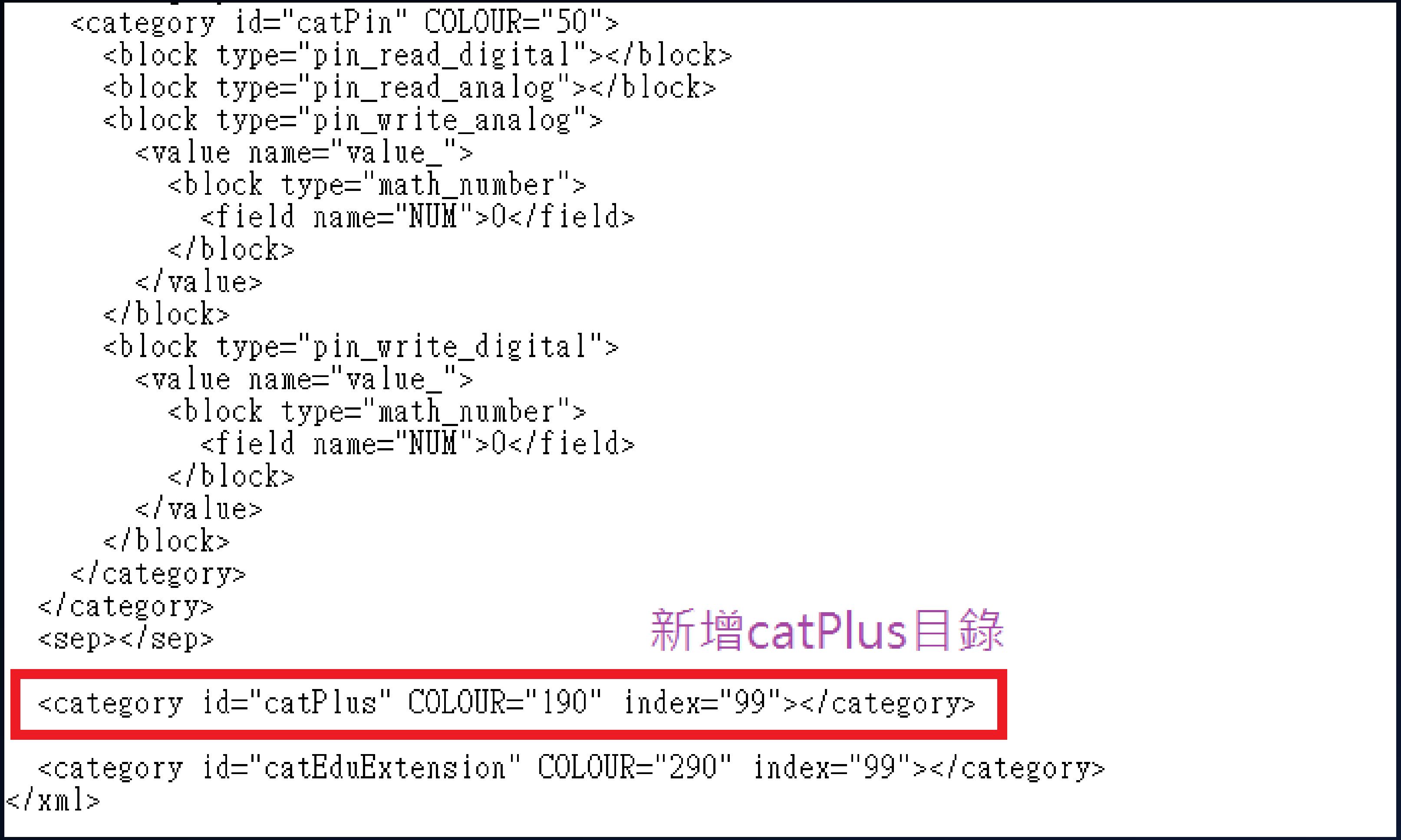
將資料夾Webbit_customBlocks\package.nw覆蓋至
C:\Webduino\WebBit\ package.nw



名稱	修改日期	類型	大小
locales	2020/8/14 下午 0...	檔案資料夾	
package.nw	2020/8/14 下午 0...	檔案資料夾	
swiftshader	2020/8/14 下午 0...	檔案資料夾	
usb_driver	2020/8/14 下午 0...	檔案資料夾	
credits	2020/4/24 下午 0...	Chrome HTML D...	2,114 KB
d3dcompiler_47.dll	2020/4/24 下午 0...	應用程式擴充	3,573 KB
ffmpeg.dll	2020/4/24 下午 0...	應用程式擴充	1,287 KB
icudtl.dat	2020/4/24 下午 0...	DAT 檔案	10,085 KB

開啟目錄 C:\Webduino\WebBit\package.nw\blockly\toolbox\index.xml 編輯新增「進階 catPlus」程式碼，可將自訂積木設定掛載在此目錄下。

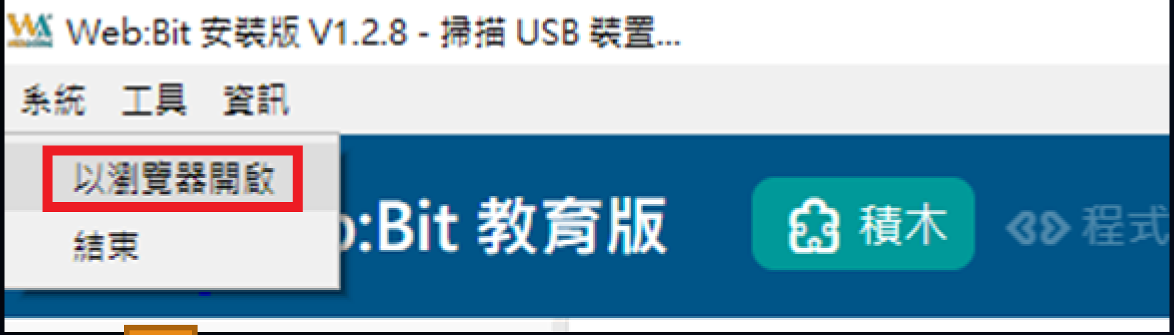
<category id="catPlus" COLOUR="190" index="99"> </category>

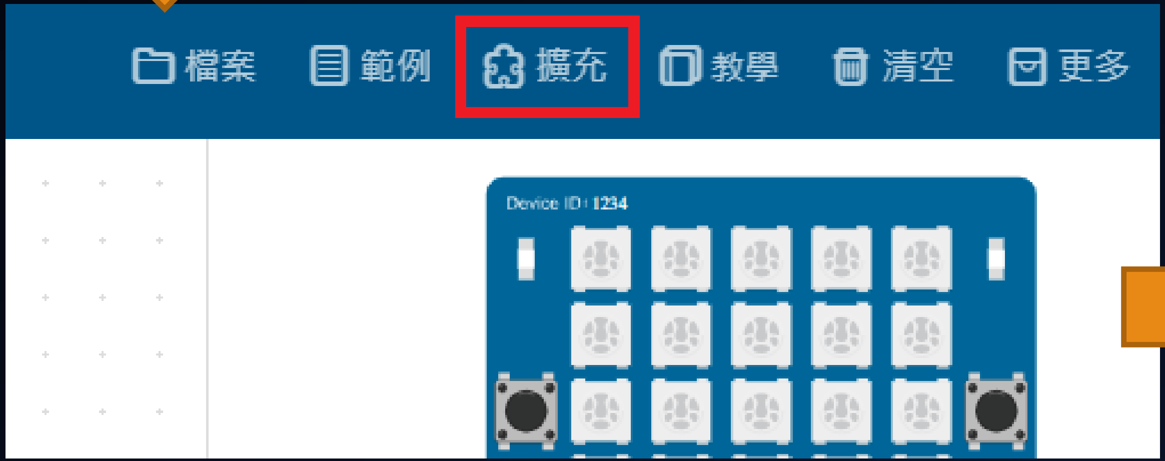


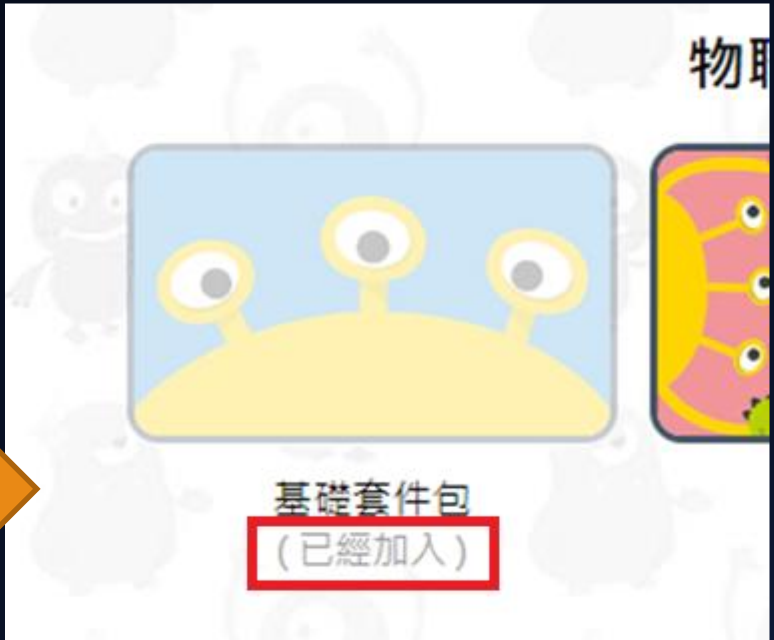
```
    <category id="catPin" COLOUR="50">
      <block type="pin_read_digital"></block>
      <block type="pin_read_analog"></block>
      <block type="pin_write_analog">
        <value name="value_">
          <block type="math_number">
            <field name="NUM">0</field>
          </block>
        </value>
      </block>
      <block type="pin_write_digital">
        <value name="value_">
          <block type="math_number">
            <field name="NUM">0</field>
          </block>
        </value>
      </block>
    </category>
  </category>
  <sep></sep>

  <category id="catPlus" COLOUR="190" index="99"></category>

  <category id="catEduExtension" COLOUR="290" index="99"></category>
</xml>
```

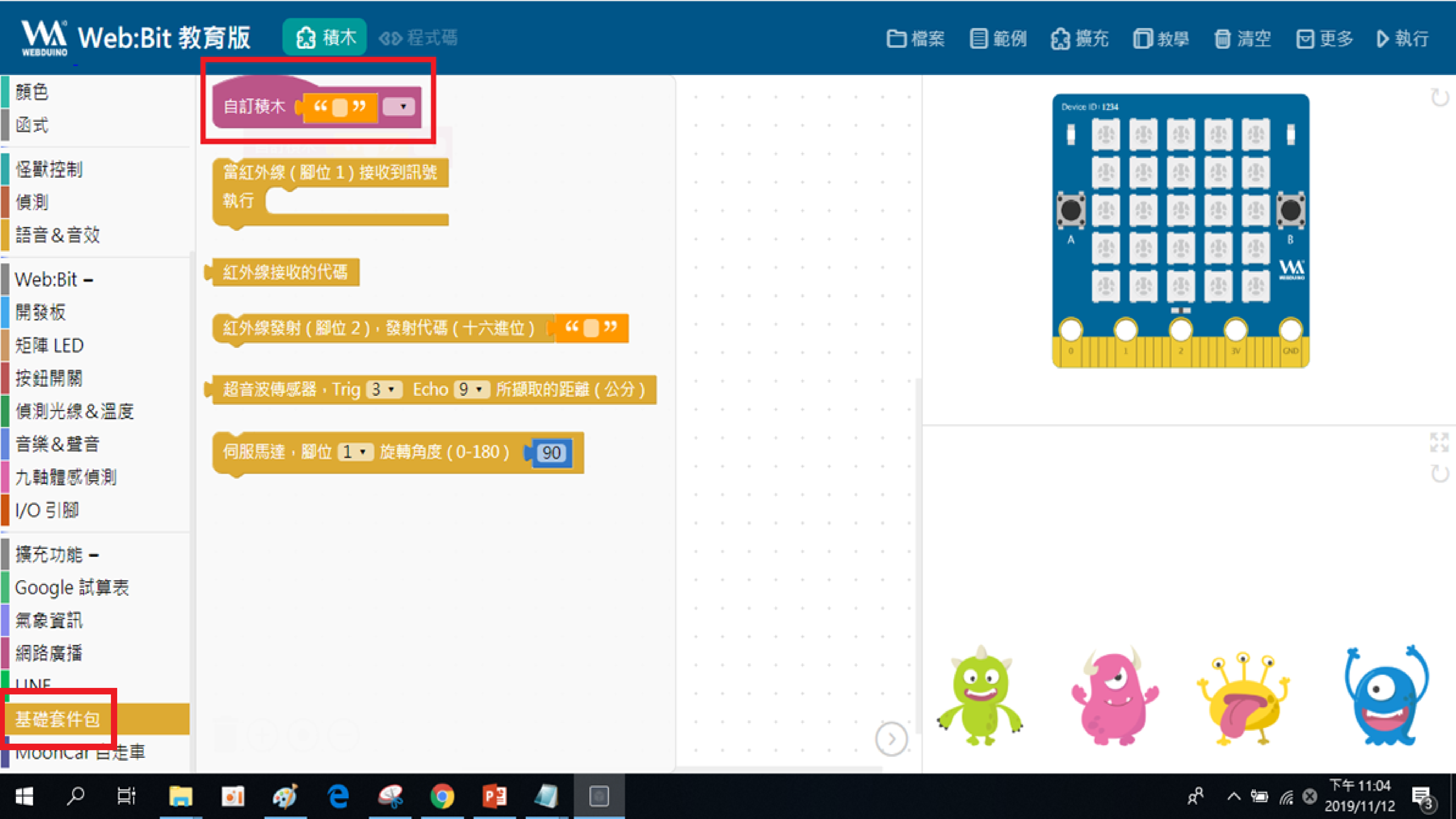
新增catPlus目錄

「以瀏覽器開啟」利於自訂積木撰寫與除錯，且AI影像辨識積木才可正常執行。









自訂積木 “ ”

當紅外線（腳位 1）接收到訊號
執行

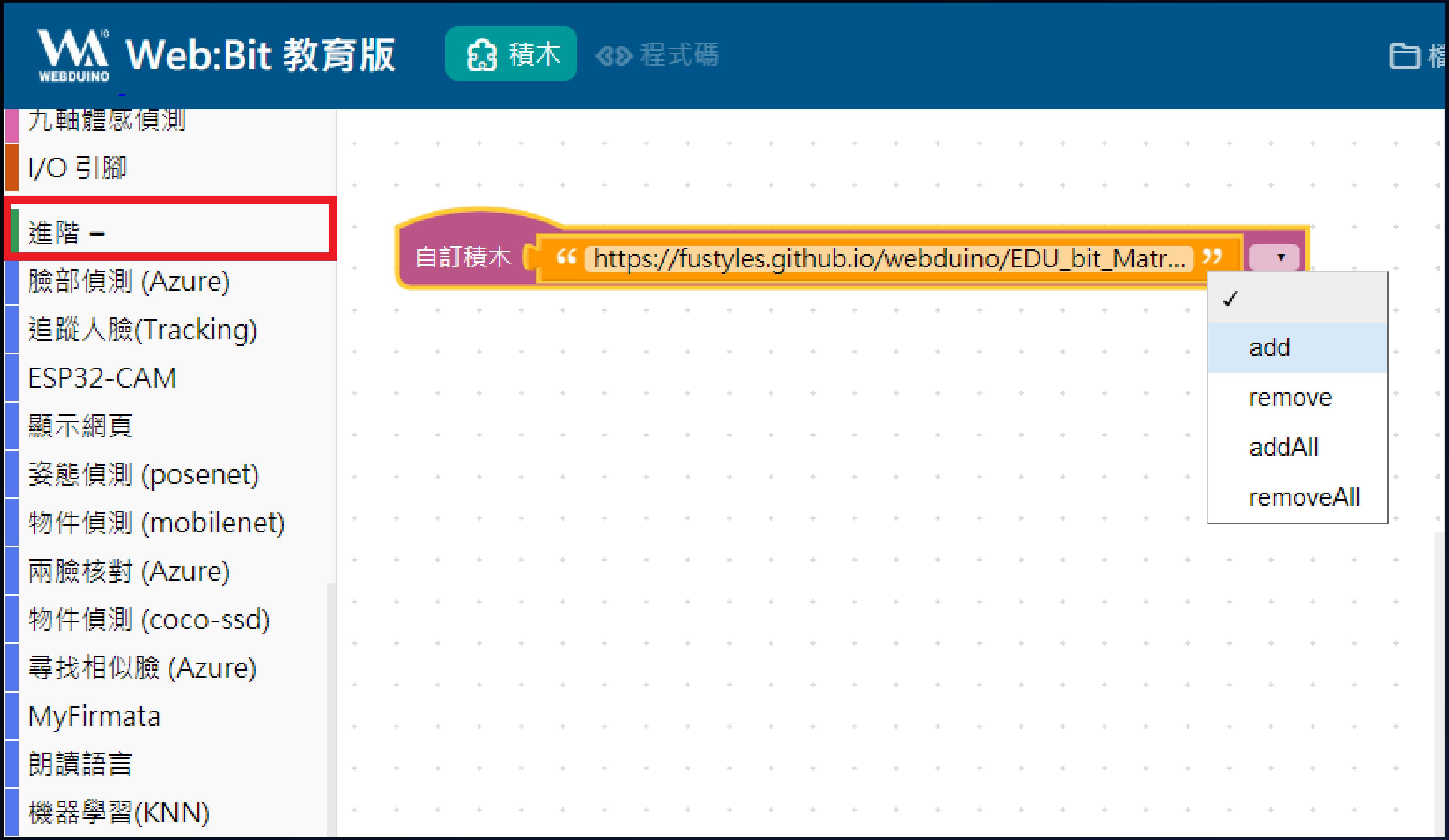
紅外線接收的代碼

紅外線發射（腳位 2），發射代碼（十六進位） “ ”

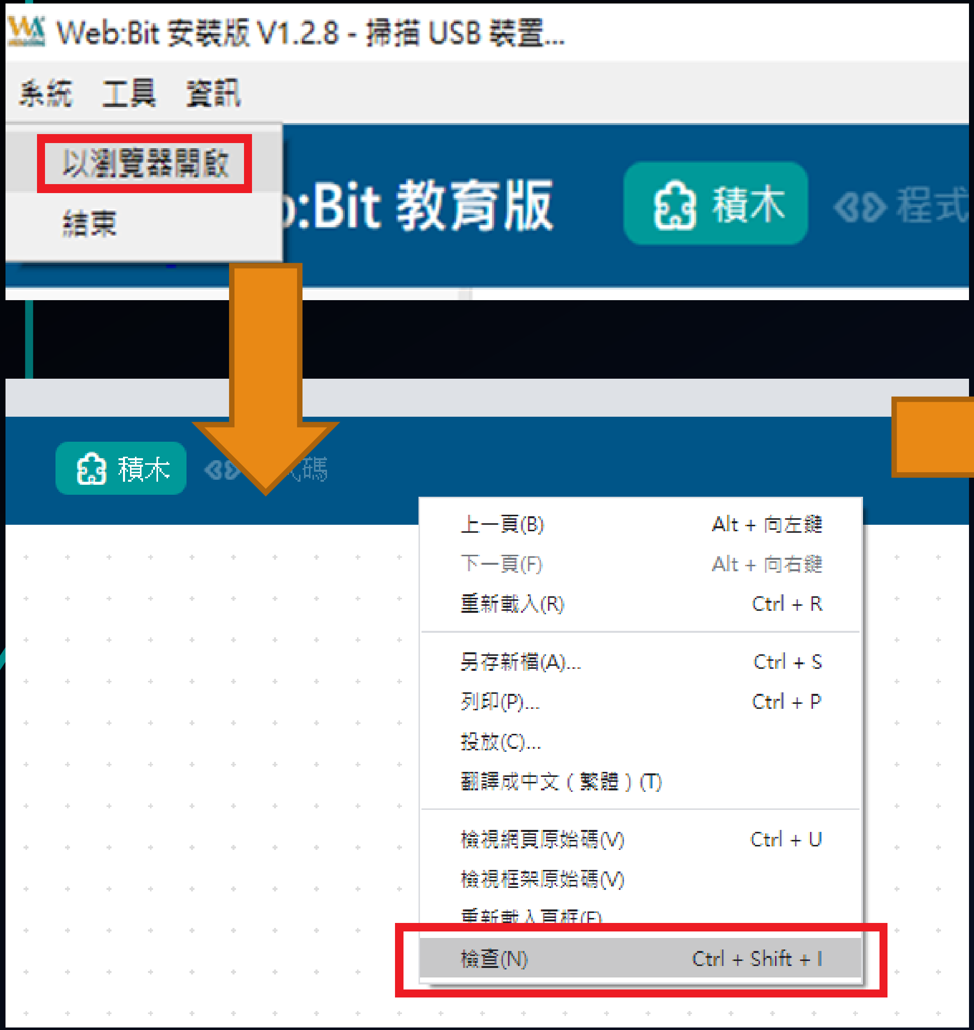
超音波傳感器，Trig 3 Echo 9 所擷取的距離（公分）

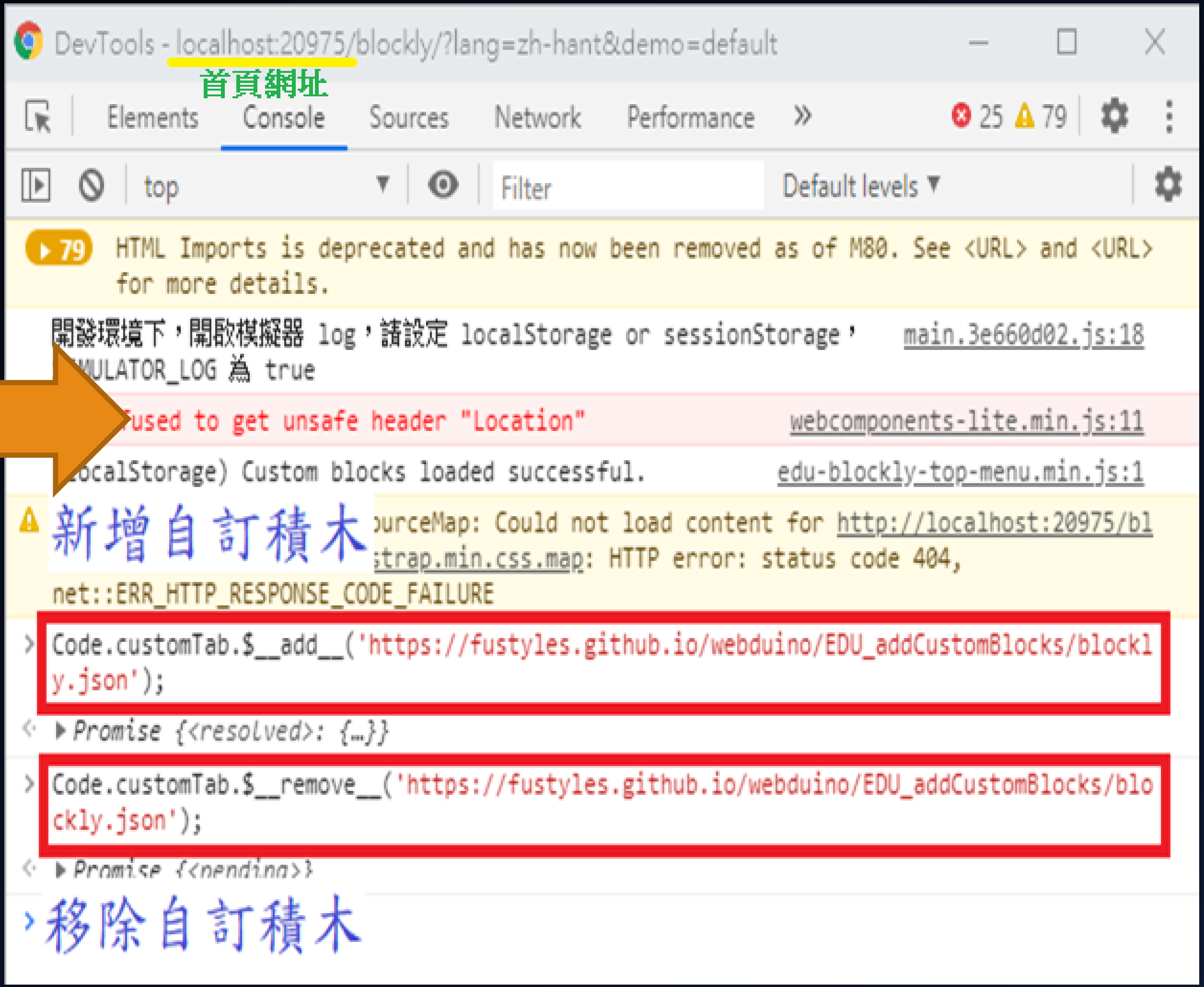
伺服馬達，腳位 1 旋轉角度（0-180） 90

填入自訂積木連結下拉選單點選add，或點選addAll新增所有自訂積木。
https://github.com/fustyles/webduino/blob/master/CustomBlock.txt



Webbit教育版新增自訂積木(法二)





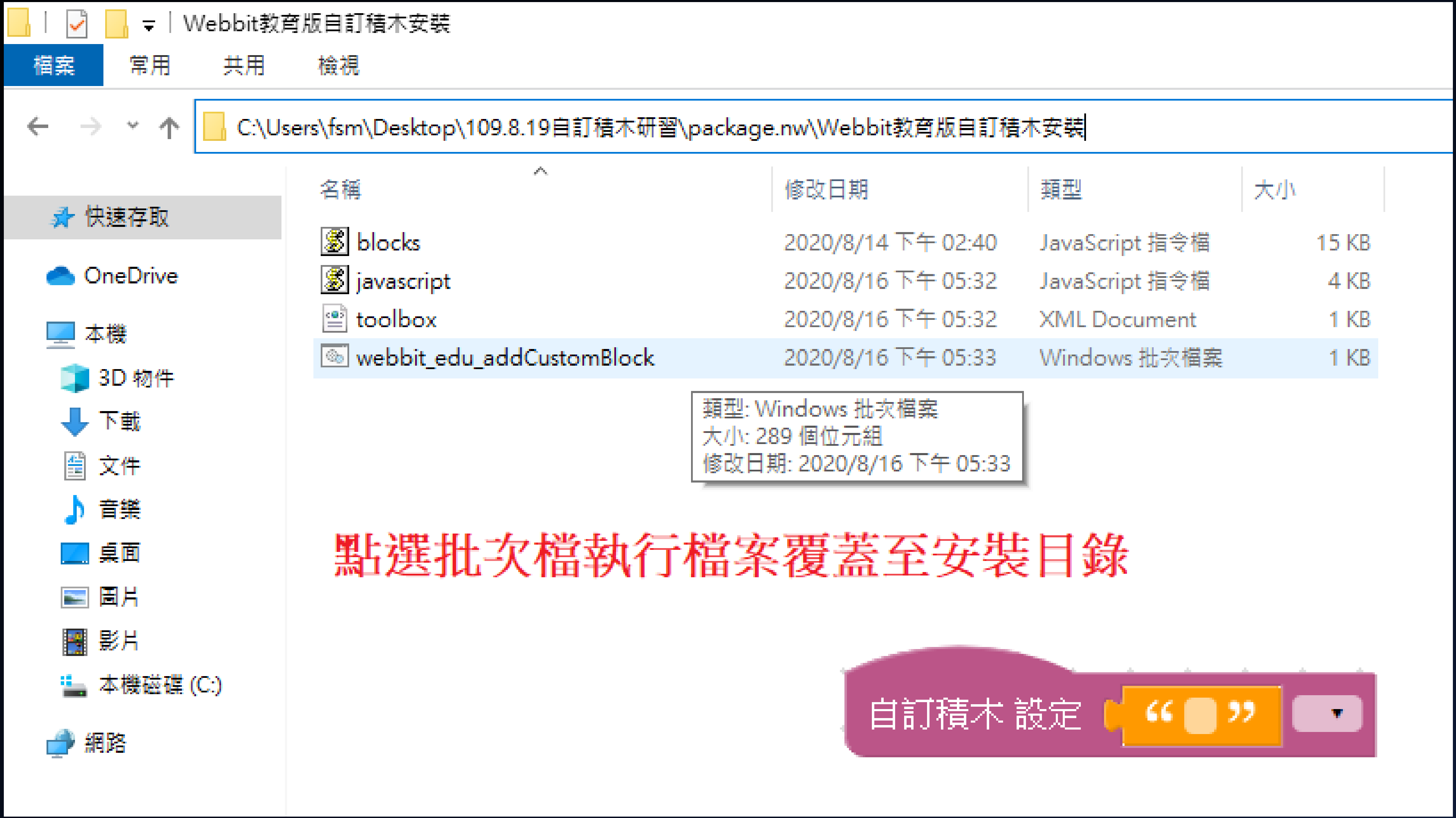
```
HTML Imports is deprecated and has now been removed as of M80. See <URL> and <URL>
for more details.
開發環境下，開啟模擬器 log，請設定 localStorage or sessionStorage，   main.3e660d02.js:18
 MULATOR_LOG 為 true
 used to get unsafe header "Location"                  webcomponents-lite.min.js:11
 ocalStorage) Custom blocks loaded successful.         edu-blockly-top-menu.min.js:1
 ourceMap: Could not load content for http://localhost:20975/bl
 strap.min.css.map: HTTP error: status code 404,
net::ERR_HTTP_RESPONSE_CODE_FAILURE
> Code.customTab.$__add__('https://fustyles.github.io/webduino/EDU_addCustomBlocks/blockly.json');
< ▶Promise {<resolved>: {…}}
> Code.customTab.$__remove__('https://fustyles.github.io/webduino/EDU_addCustomBlocks/blockly.json');
< ▶Promise {<pending>}
```

新增自訂積木

移除自訂積木

指令參考：https://github.com/fustyles/webduino/blob/master/EDU_addCustomBlocks/blockly/blocks.js

Webbit教育版新增自訂積木(法三)



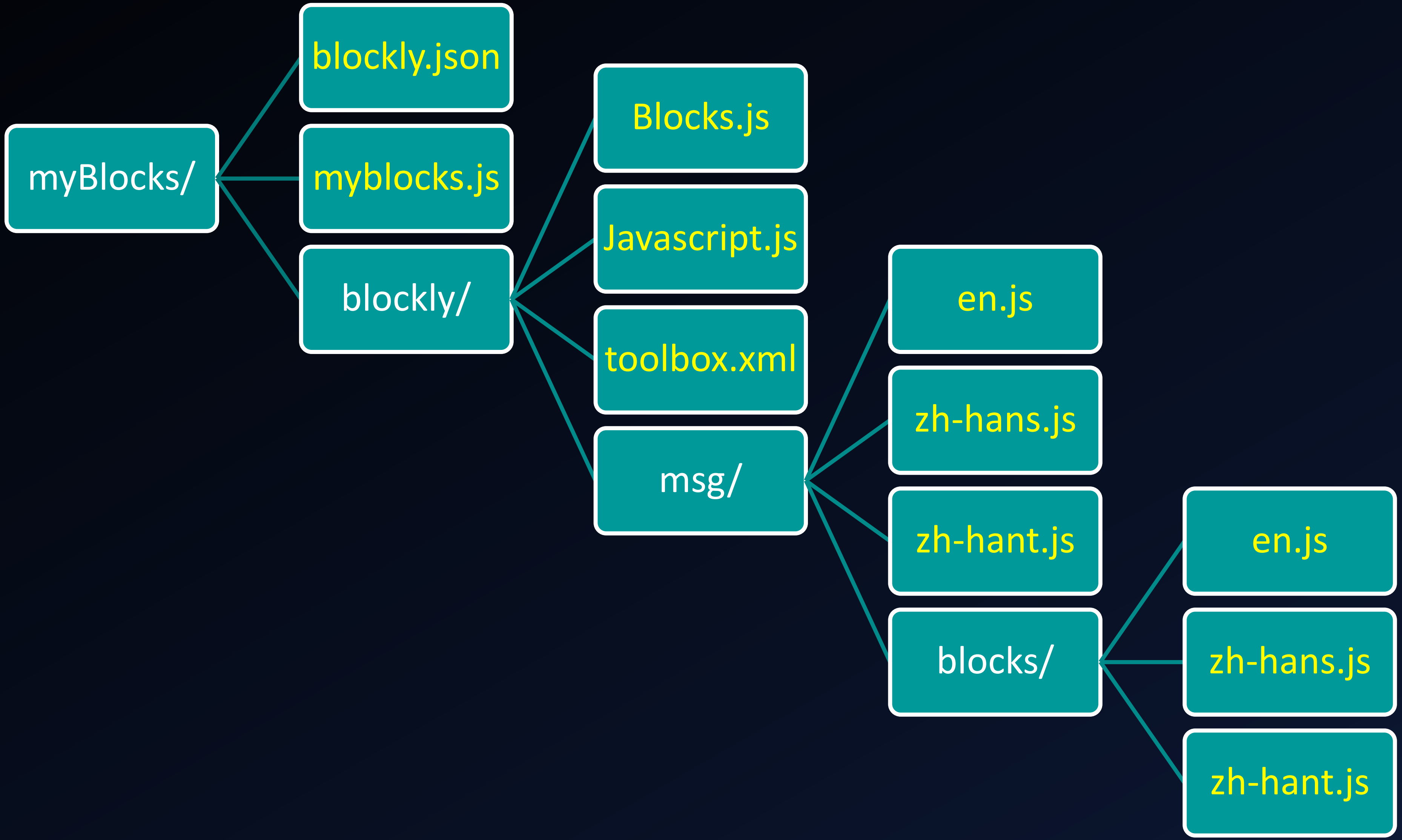
C:\Users\fsm\Desktop\109.8.19自訂積木研習\package.nw\Webbit教育版自訂積木安裝

名稱	修改日期	類型	大小
blocks	2020/8/14 下午 02:40	JavaScript 指令檔	15 KB
javascript	2020/8/16 下午 05:32	JavaScript 指令檔	4 KB
toolbox	2020/8/16 下午 05:32	XML Document	1 KB
webbit_edu_addCustomBlock	2020/8/16 下午 05:33	Windows 批次檔案	1 KB

類型: Windows 批次檔案
大小: 289 個位元組
修改日期: 2020/8/16 下午 05:33

點選批次檔執行檔案覆蓋至安裝目錄

Webduino檔案架構(雲端平台、教育版、Kebbi)



myBlocks/

blockly.json

myblocks.js

blockly/

Blocks.js

Javascript.js

toolbox.xml

msg/

en.js

zh-hans.js

zh-hant.js

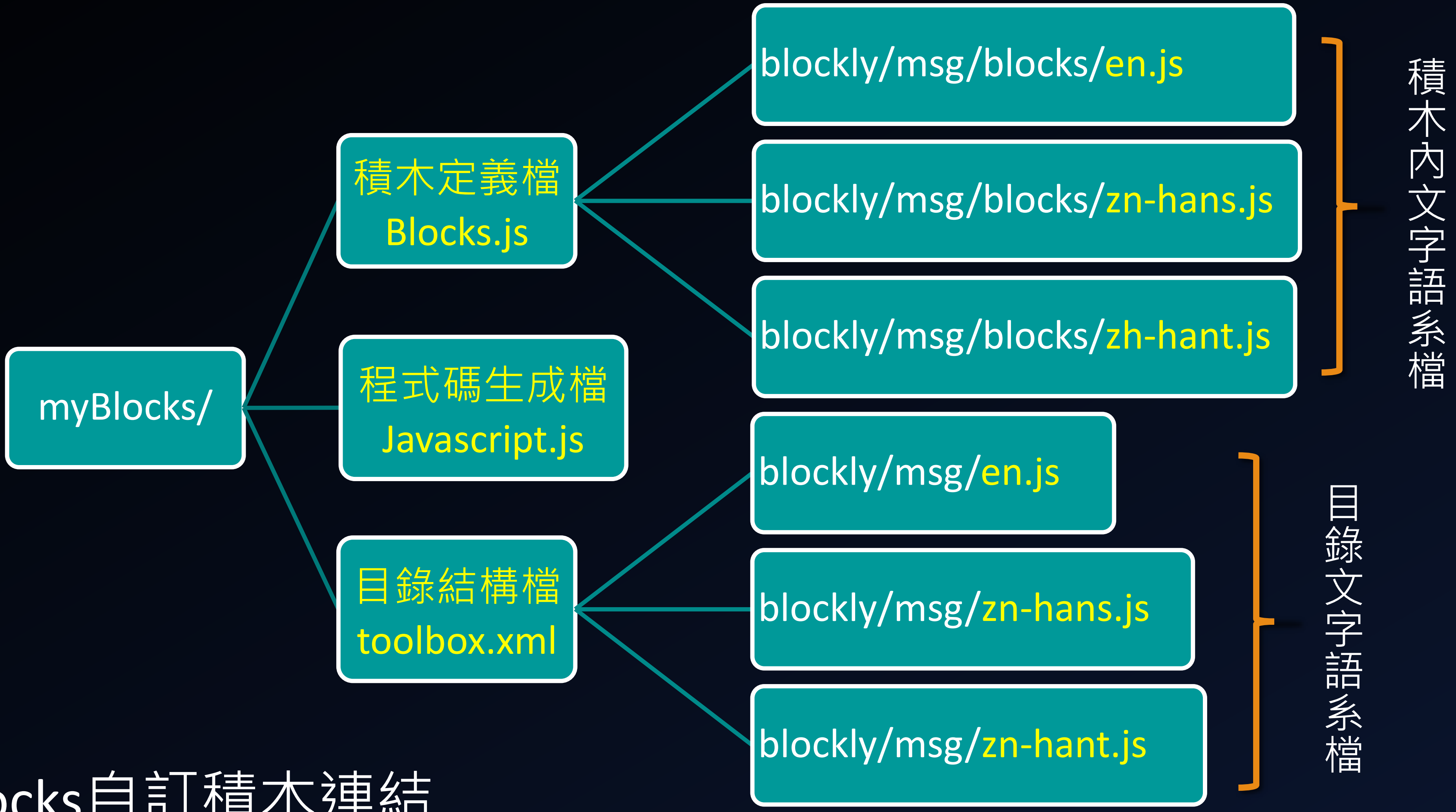
blocks/

en.js

zh-hans.js

zh-hant.js

Webduino自訂積木架構



myBlocks自訂積木連結

http://localhost:20975/myBlocks/blockly.json

自訂積木撰寫流程 (Webduino)

Blockly Developer Tools 製作自訂積木：
可匯出blocks.js, javascript.js, toolbox.xml 三個積木主要檔案。

myBlocks\blockly.json編輯：
修改"types"加入新增積木NAME。修改"dependencies"加入自訂JS函式庫路徑，也可為外部函式庫網路路徑。

myBlocks\blockly\blocks.js編輯：
加入匯出blocks.js內的所有程式碼。

myBlocks\blockly\blocks.js編輯：
固定文字修改成動態語系變數，變數格式為"Blockly.Msg.自訂名稱"。例如固定文字"pin"更改成動態語系變數Blockly.Msg.pin

myBlocks\blockly\msg\blocks\en.js(英文), myBlocks\blockly\msg\blocks\zh-hans.js(簡中), myBlocks\blockly\msg\blocks\zh-hant.js(繁中)編輯：
對應myBlocks\blockly\blocks.js內"Blockly.Msg.自訂名稱" 動態語系變數，分別設定變數不同語系文字值。

myBlocks\blockly\javascript.js編輯：
加入匯出javascript.js內的所有程式碼。

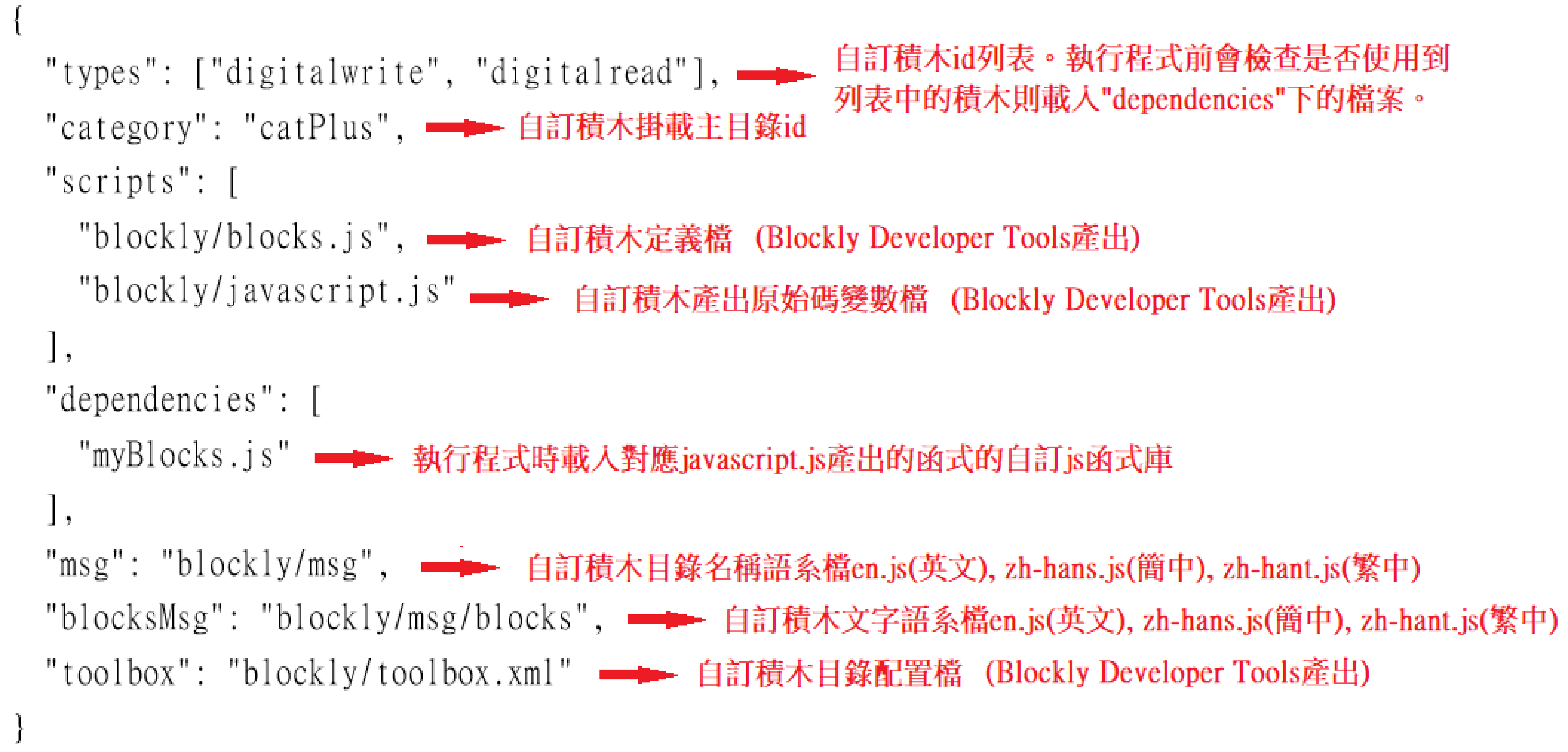
myBlocks\blockly\javascript.js編輯：
修改程式碼 var code = "...;\n";或var code = "...";結合積木內部取得變數值或函式利用字串組合產出程式碼內容，可在此新增額外的程式碼插入程式最上方。可在此步驟規劃是否使用自訂JS函式並於myBlocks\myBlocks.js 內新增對應的函式。

myBlocks\myBlocks.js編輯：
新增對應myBlocks\blockly\javascript.js的var code="自訂函式";程式碼內的自訂函式。

myBlocks\blockly\toolbox.xml編輯：
參考範例檔主目錄與子目錄格式，可新增主目錄或子目錄。貼入匯出toolbox.xml內僅區塊<block...></block>的程式碼至指定的主目錄或子目錄。

myBlocks\blockly\msg\en.js(英文), myBlocks\blockly\msg\zh-hans.js(簡中), myBlocks\blockly\msg\zh-hant.js(繁中)編輯：
對照myBlocks\blockly\toolbox.xml內<category id="目錄ID"...>，設定目錄動態語系變數格式為"MSG.目錄ID"，例如category id="myBlocks"則變數名為MSG.myBlocks，此變數可自動設定目錄NAME值。

積木佈署檔 Blockly.json (Webduino)

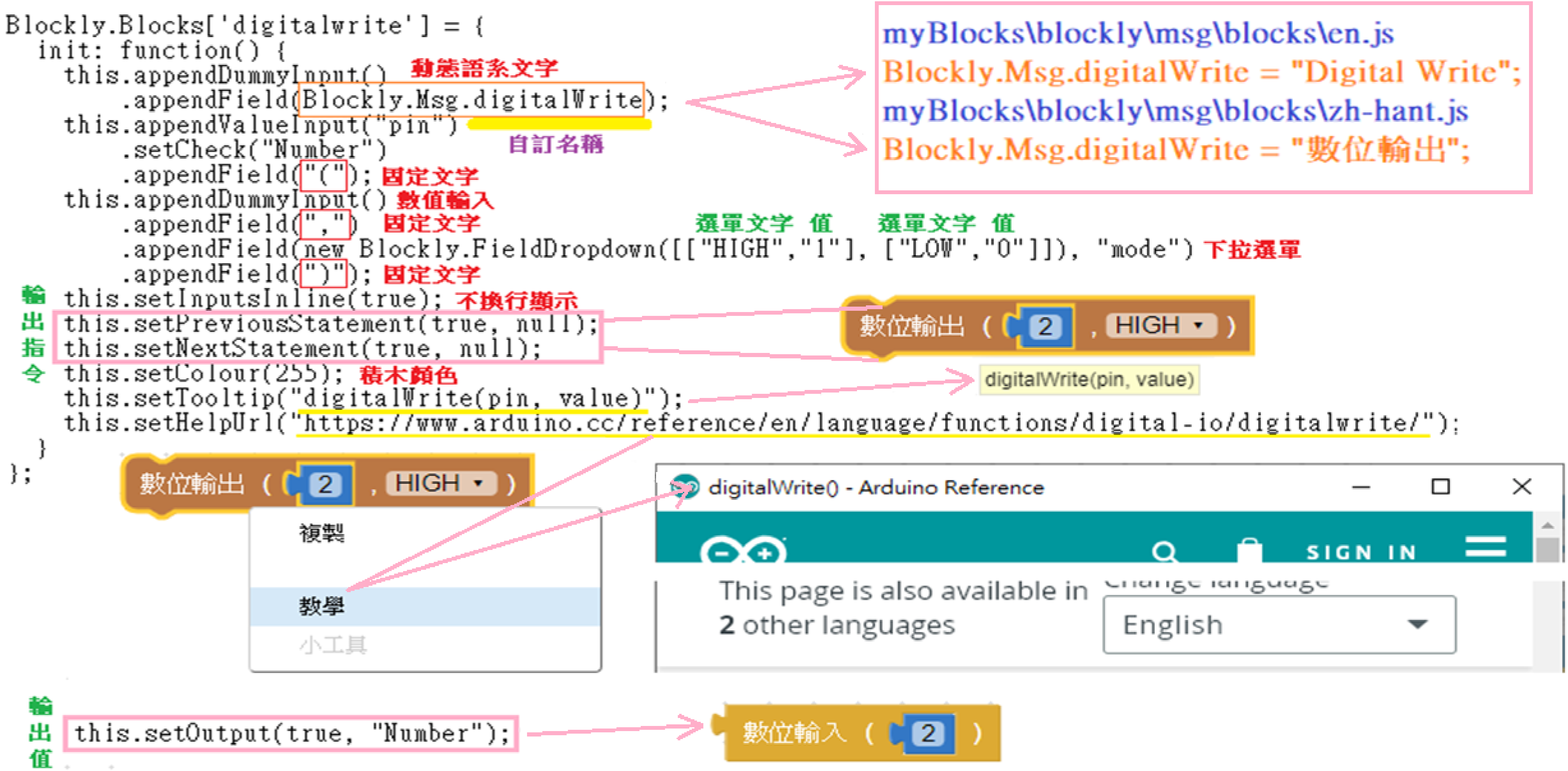


```
{
  "types": ["digitalwrite", "digitalread"],  →  自訂積木id列表。執行程式前會檢查是否使用到
                                                列表中的積木則載入"dependencies"下的檔案。
  "category": "catPlus",  →  自訂積木掛載主目錄id
  "scripts": [
    "blockly/blocks.js",  →  自訂積木定義檔  (Blockly Developer Tools產出)
    "blockly/javascript.js"  →  自訂積木產出原始碼變數檔  (Blockly Developer Tools產出)
  ],
  "dependencies": [
    "myBlocks.js"  →  執行程式時載入對應javascript.js產出的函式的自訂js函式庫
  ],
  "msg": "blockly/msg",  →  自訂積木目錄名稱語系檔en.js(英文), zh-hans.js(簡中), zh-hant.js(繁中)
  "blocksMsg": "blockly/msg/blocks",  →  自訂積木文字語系檔en.js(英文), zh-hans.js(簡中), zh-hant.js(繁中)
  "toolbox": "blockly/toolbox.xml"  →  自訂積木目錄配置檔  (Blockly Developer Tools產出)
}
```

積木定義檔 blocks.js (Webduino)

```
Blockly.Blocks['digitalwrite'] = {  init: function() {
  this.appendDummyInput()
    .appendField("digitalWrite");    ➔  .appendField(Blockly.Msg.digitalWrite);
  this.appendValueInput("pin")            修改為動態語系變數
    .setCheck("Number")
    .appendField("(");
  this.appendDummyInput()
    .appendField(",")
    .appendField(new Blockly.FieldDropdown([["HIGH","1"], ["LOW","0"]]), "mode")
    .appendField(")");
this.setInputsInline(true);
this.setPreviousStatement(true, null);
this.setNextStatement(true, null);
this.setColour(255);
this.setTooltip("digitalWrite(pin, value)");
this.setHelpUrl("https://www.arduino.cc/reference/en/language/functions/digital-io/digitalwrite/");
}};
```

積木定義檔 blocks.js (Webduino)



```
Blockly.Blocks['digitalwrite'] = {
  init: function() {
    this.appendDummyInput()
        .appendField(Blockly.Msg.digitalWrite);
    this.appendValueInput("pin")
        .setCheck("Number")
        .appendField("(");
    this.appendDummyInput()
        .appendField(",")
        .appendField(new Blockly.FieldDropdown([["HIGH","1"], ["LOW","0"]]), "mode")
        .appendField(")");
    this.setInputsInline(true);
    this.setPreviousStatement(true, null);
    this.setNextStatement(true, null);
    this.setColour(255);
    this.setTooltip("digitalWrite(pin, value)");
    this.setHelpUrl("https://www.arduino.cc/reference/en/language/functions/digital-io/digitalwrite/");
  }
};
```

動態語系文字

自訂名稱

固定文字

數值輸入

固定文字

選單文字 值 選單文字 值

下拉選單

固定文字

不換行顯示

輸出指令

積木顏色

myBlocks\blockly\msg\blocks\en.js
Blockly.Msg.digitalWrite = "Digital Write";
myBlocks\blockly\msg\blocks\zh-hant.js
Blockly.Msg.digitalWrite = "數位輸出";

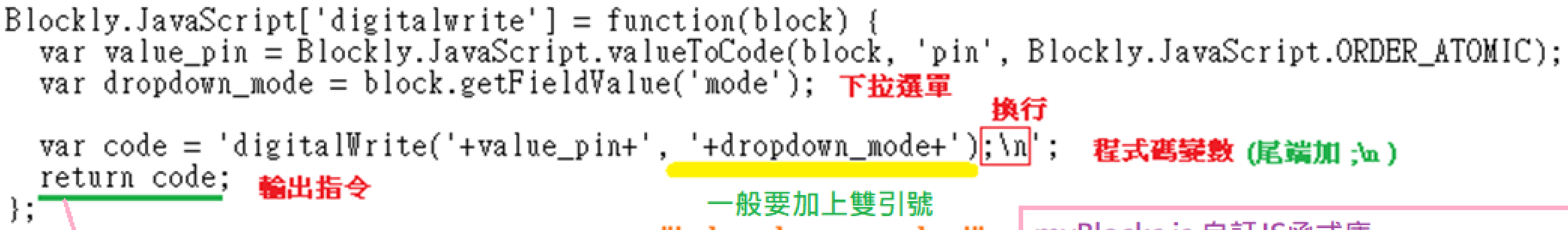
```
this.setOutput(true, "Number");
```

輸出值

程式碼產出檔 javascript.js (Webduino)

```
Blockly.JavaScript['digitalwrite'] = function(block) {
  var value_pin = Blockly.JavaScript.valueToCode(block, 'pin', Blockly.JavaScript.ORDER_ATOMIC);
  var dropdown_mode = block.getFieldValue('mode');

  // TODO: Assemble JavaScript into code variable.
  var code = '...;\n';   →  var code = 'digitalWrite('+value_pin+', '+dropdown_mode+');\n' ;
                            可結合JS函式庫

  return code;
};
```

程式碼產出檔 javascript.js (Webduino)



```
Blockly.JavaScript['digitalwrite'] = function(block) {
  var value_pin = Blockly.JavaScript.valueToCode(block, 'pin', Blockly.JavaScript.ORDER_ATOMIC);
  var dropdown_mode = block.getFieldValue('mode');

  var code = 'digitalWrite('+value_pin+', '+dropdown_mode+');\n';
  return code;
};
```

值輸入

下拉選單

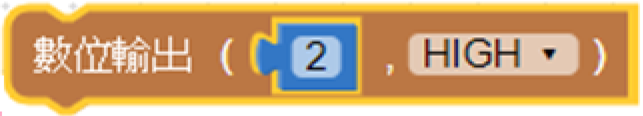
換行

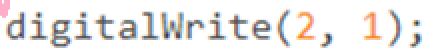
程式碼變數 (尾端加 ;\n)

輸出指令

一般要加上雙引號

"'+dropdown_mode+'"

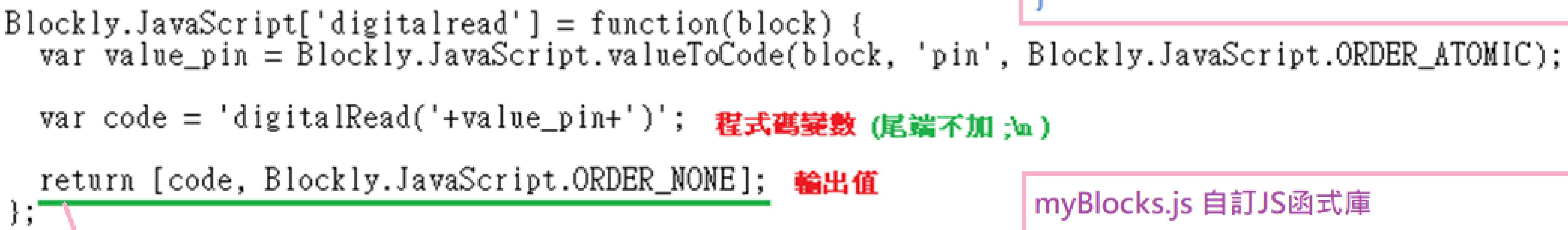




```
digitalWrite(2, 1);
```

myBlocks.js 自訂JS函式庫

```
function digitalWrite(pin, val) {
    var url = "http://192.168.1.100/?digitalwrite="+pin+";"+val;
    console.log(url);
    fetch(url);
}
```

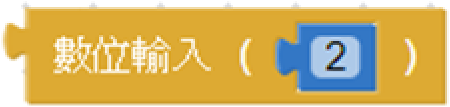


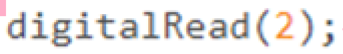
```
Blockly.JavaScript['digitalread'] = function(block) {
  var value_pin = Blockly.JavaScript.valueToCode(block, 'pin', Blockly.JavaScript.ORDER_ATOMIC);

  var code = 'digitalRead('+value_pin+')';

  return [code, Blockly.JavaScript.ORDER_NONE];
};
```

值輸入

程式碼變數 (尾端不加 ;\n)

輸出值



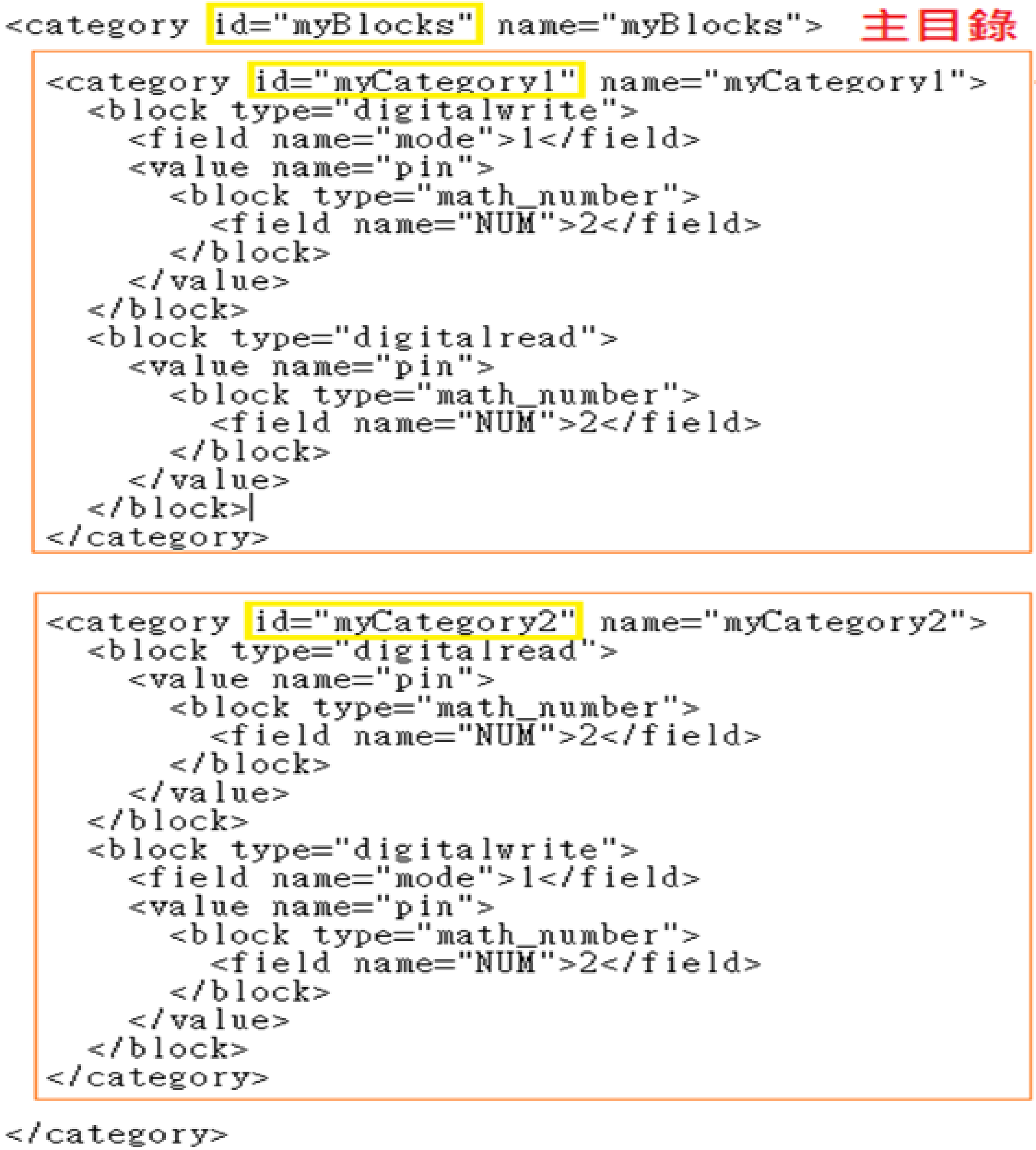


```
digitalRead(2);
```

myBlocks.js 自訂JS函式庫

```
function digitalRead(pin) {
    var url = "http://192.168.1.100/?digitalread="+pin;
    console.log(url);
    return url;
}
```

目錄結構檔 toolbox.xml (Webduino) 對應id



```
<category id="myBlocks" name="myBlocks">
```

主目錄

```
  <category id="myCategory1" name="myCategory1">
    <block type="digitalwrite">
      <field name="mode">1</field>
      <value name="pin">
        <block type="math_number">
          <field name="NUM">2</field>
        </block>
      </value>
    </block>
    <block type="digitalread">
      <value name="pin">
        <block type="math_number">
          <field name="NUM">2</field>
        </block>
      </value>
    </block>
  </category>
```

子目錄1

```
  <category id="myCategory2" name="myCategory2">
    <block type="digitalread">
      <value name="pin">
        <block type="math_number">
          <field name="NUM">2</field>
        </block>
      </value>
    </block>
    <block type="digitalwrite">
      <field name="mode">1</field>
      <value name="pin">
        <block type="math_number">
          <field name="NUM">2</field>
        </block>
      </value>
    </block>
  </category>
```

子目錄2

```
</category>
```

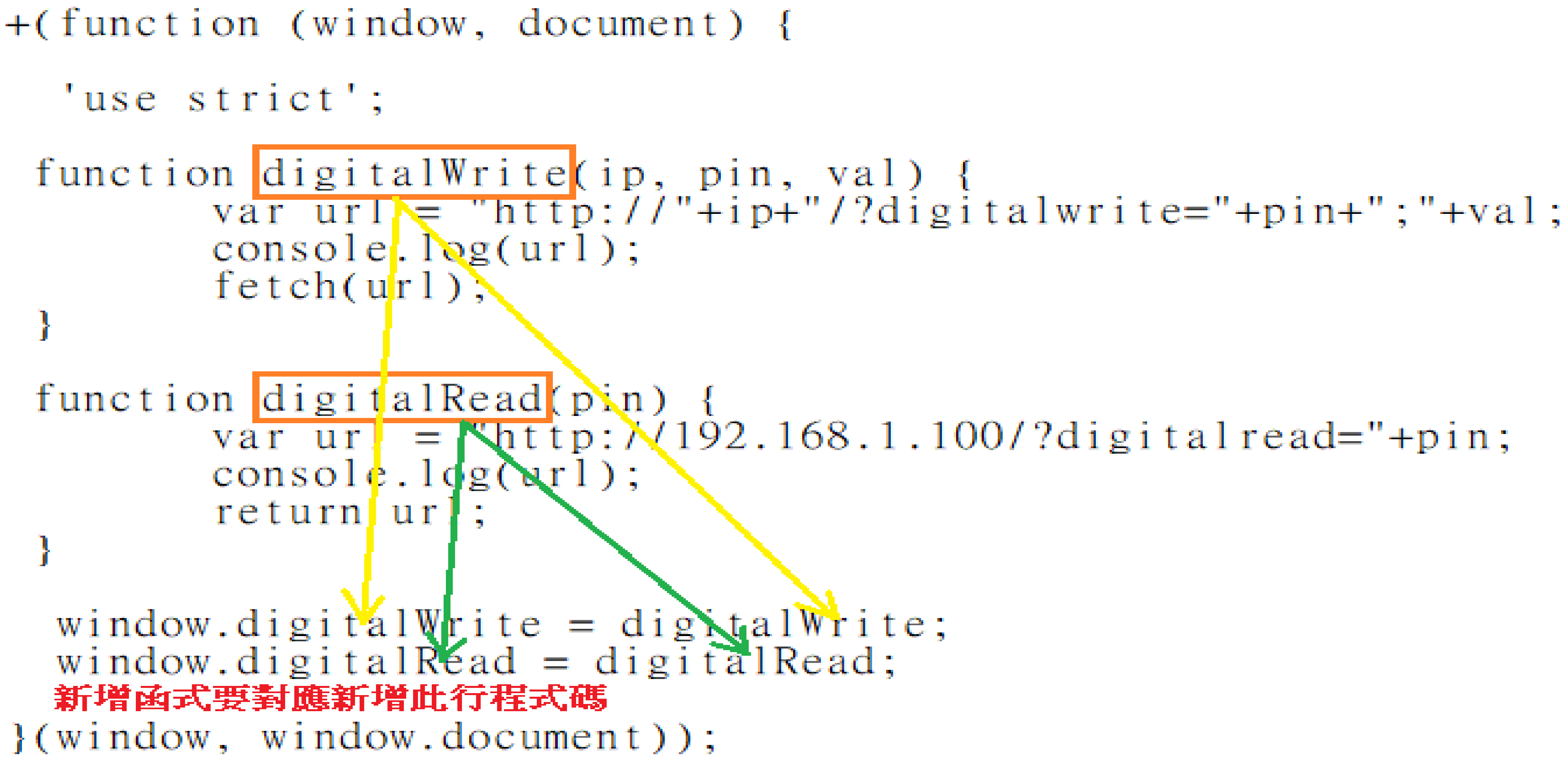
myBolck\blockly\msg\en.js

```
MSG.myBlocks = "My Blocks";
MSG.myCategory1 = "Category 1";
MSG.myCategory2 = "Category 2";
```

myBolck\blockly\msg\zh-hant.js

```
MSG.myBlocks = "我的積木";
MSG.myCategory1 = "目錄 1";
MSG.myCategory2 = "目錄 2";
```

自訂JS函式庫 myBlocks.js (Webduino)

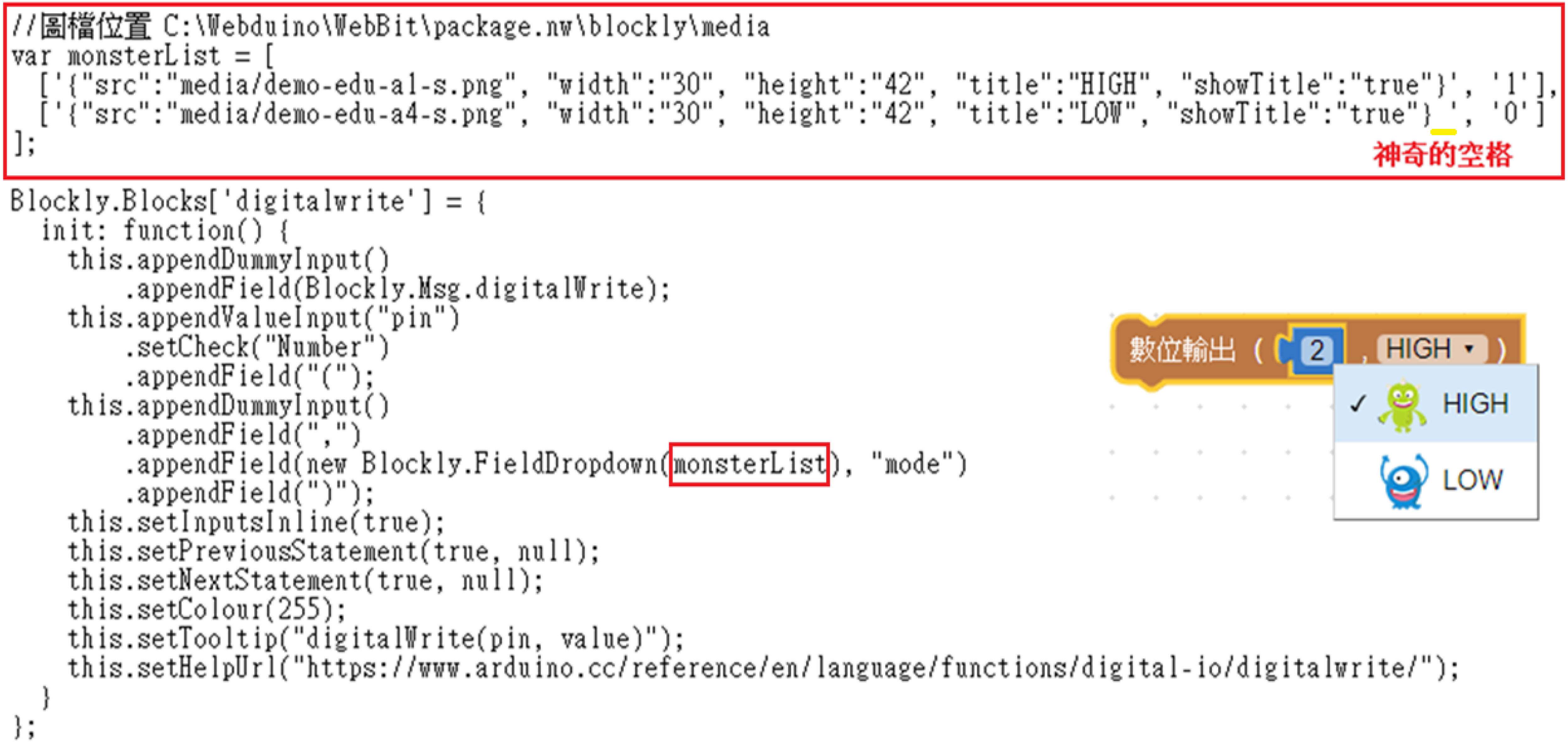


```
+(function (window, document) {

  'use strict';

 function digitalWrite(ip, pin, val) {
        var url = "http://"+ip+"/?digitalwrite="+pin+";"+val;
        console.log(url);
        fetch(url);
 }

 function digitalRead(pin) {
        var url = "http://192.168.1.100/?digitalread="+pin;
        console.log(url);
        return url;
 }

  window.digitalWrite = digitalWrite;
  window.digitalRead = digitalRead;
```

新增函式要對應新增此行程式碼

```
}(window, window.document));
```

下拉圖檔選單 (Webduino)



```
//圖檔位置 C:\Webduino\WebBit\package.nw\blockly\media
var monsterList = [
  ['{"src":"media/demo-edu-a1-s.png", "width":"30", "height":"42", "title":"HIGH", "showTitle":"true"}', '1'],
  ['{"src":"media/demo-edu-a4-s.png", "width":"30", "height":"42", "title":"LOW", "showTitle":"true"} ', '0']
];
```

神奇的空格

```
Blockly.Blocks['digitalwrite'] = {
  init: function() {
    this.appendDummyInput()
        .appendField(Blockly.Msg.digitalWrite);
    this.appendValueInput("pin")
        .setCheck("Number")
        .appendField("(");
    this.appendDummyInput()
        .appendField(",")
        .appendField(new Blockly.FieldDropdown(monsterList), "mode")
        .appendField(")");
    this.setInputsInline(true);
    this.setPreviousStatement(true, null);
    this.setNextStatement(true, null);
    this.setColour(255);
    this.setTooltip("digitalWrite(pin, value)");
    this.setHelpUrl("https://www.arduino.cc/reference/en/language/functions/digital-io/digitalwrite/");
  }
};
```

Javascript.js常用技巧 (Webduino)

xmlHttp取得XML資料與AJAX取得跨網域資料

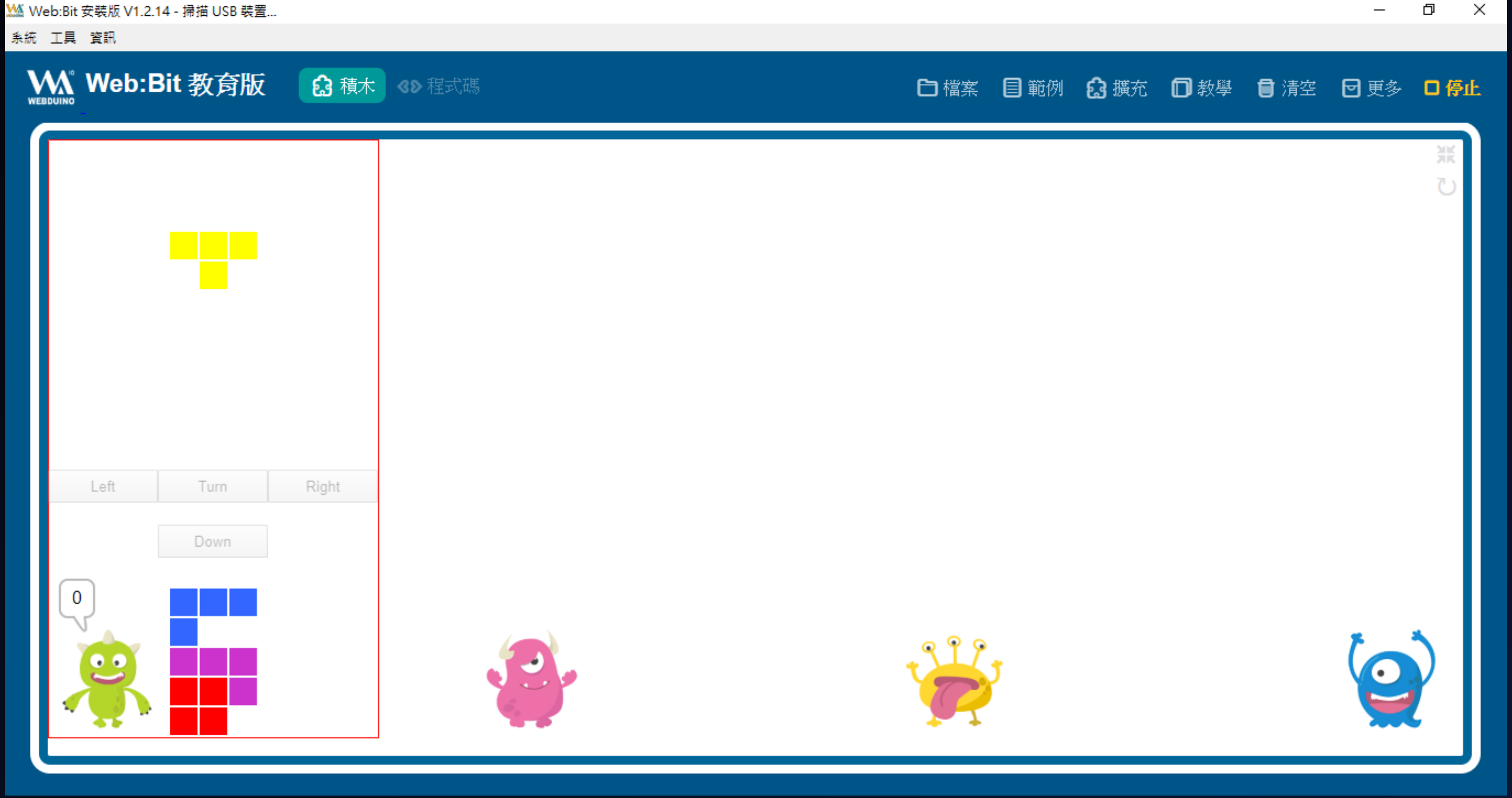
https://github.com/fustyles/webduino/blob/master/AirQuality_Taiwan_20180121/AirQualityTaiwan.js

遊戲元素自訂積木原始碼解說

https://github.com/fustyles/webduino/blob/master/GameElements_20190131/blockly/javascript.js

新增遊戲元素積木指令

https://github.com/fustyles/webduino/blob/master/EDU_addCustomBlocks/blockly/blocks.js



BlocklyDuino自訂積木撰寫

BlocklyDuino v3 Beta 5

下載網址：https://github.com/MediaTek-Labs/BlocklyDuino-for-LinkIt/releases/tag/3.0.312b

64位元 blocklyduino-3.0.312b-win64-ide.zip

32位元 blocklyduino-3.0.312b-win32-ide.zip

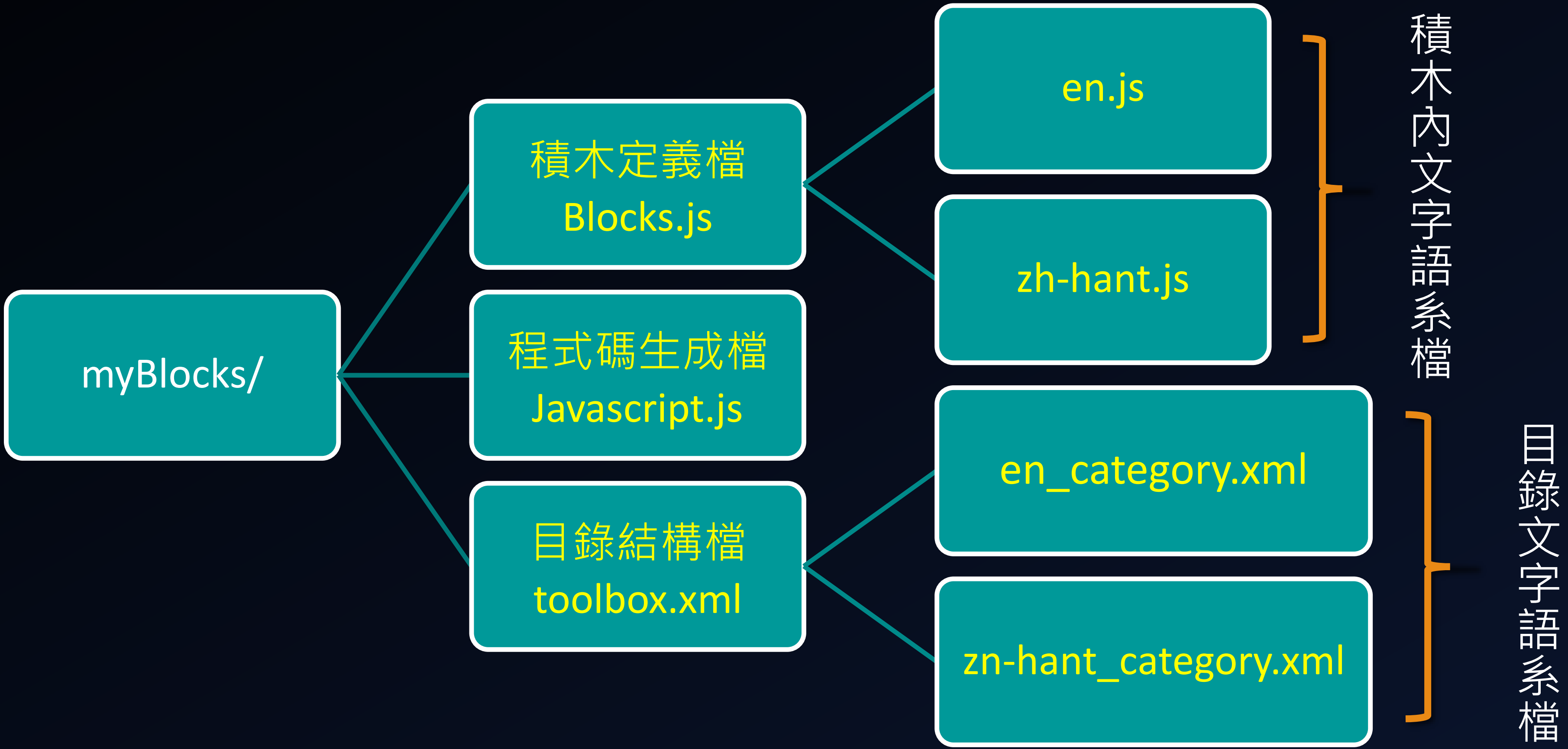
使用指南：https://docs.labs.mediatek.com/linkit-7697-blocklyduino/linkit-7697-blocklyduino-12879411.html

吉哥積木：https://sites.google.com/jes.mlc.edu.tw/ljj/linkit7697/如何安裝吉哥自製積木

ICSHOP：https://github.com/iCShopMgr/LinkIt7697_Bit_for_BlocklyDuino

法蘭斯積木：https://github.com/fustyles/webduino/tree/master/LinkIt7697/FranceFu

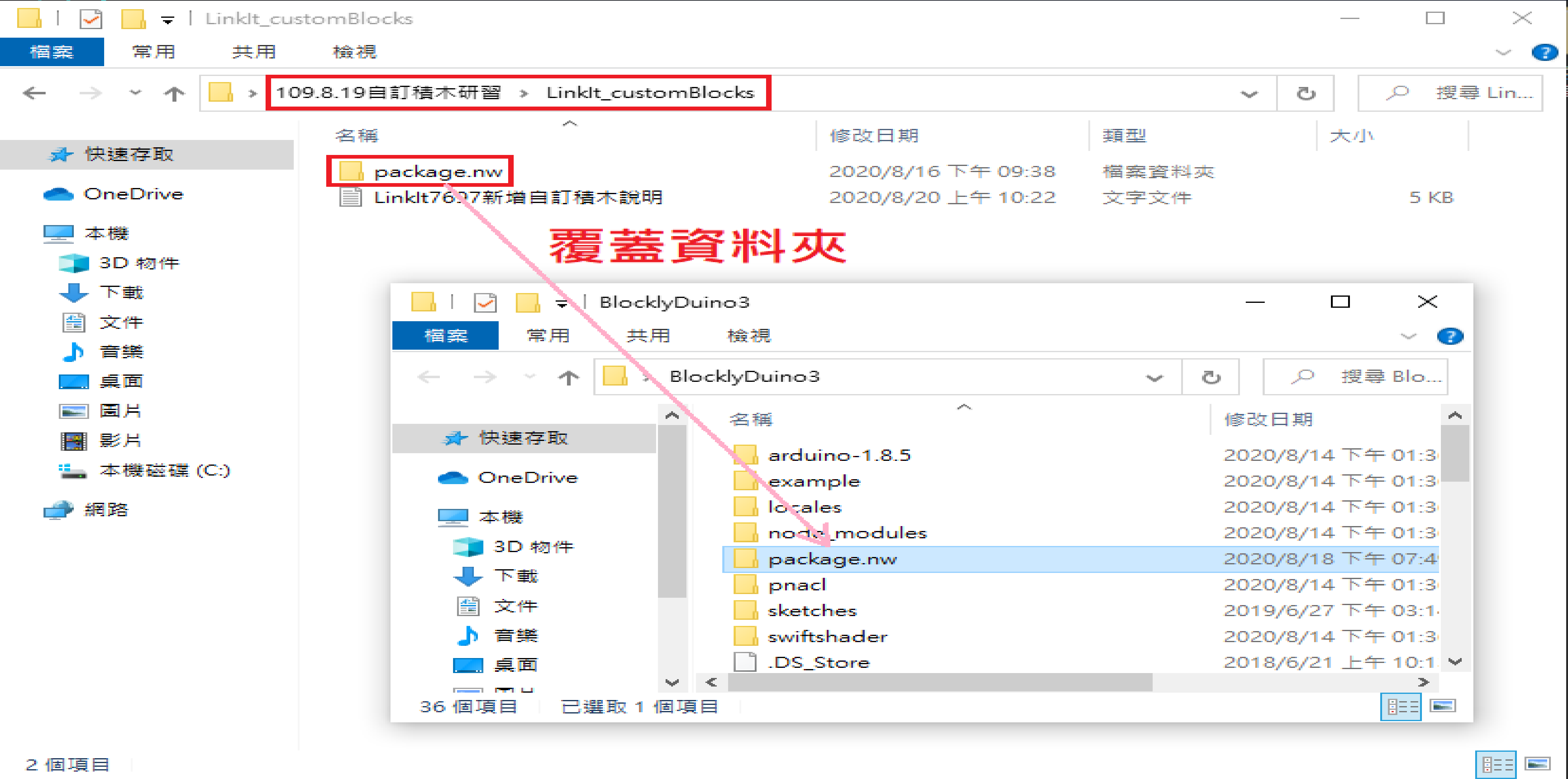
BlocklyDuino自訂積木架構



自訂積木本機連結

./myBlocks/

BlocklyDuino自訂積木安裝



109.8.19自訂積木研習 > LinkIt_customBlocks

名稱	修改日期	類型	大小
package.nw	2020/8/16 下午 09:38	檔案資料夾	
LinkIt7697新增自訂積木說明	2020/8/20 上午 10:22	文字文件	5 KB

覆蓋資料夾

BlocklyDuino3

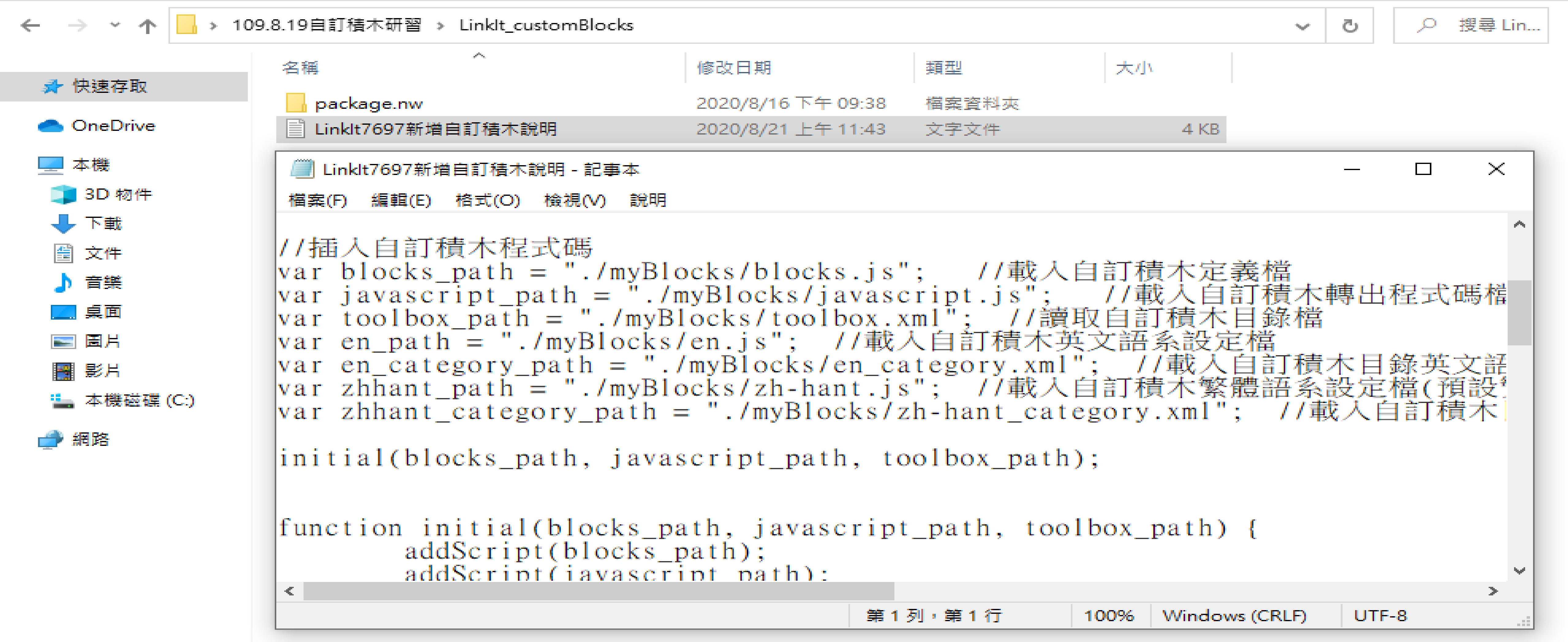
名稱	修改日期
arduino-1.8.5	2020/8/14 下午 01:3
example	2020/8/14 下午 01:3
locales	2020/8/14 下午 01:3
node_modules	2020/8/14 下午 01:3
package.nw	2020/8/18 下午 07:4
pnacl	2020/8/14 下午 01:3
sketches	2019/6/27 下午 03:1
swiftshader	2020/8/14 下午 01:3
.DS_Store	2018/6/21 上午 10:1

36 個項目 已選取 1 個項目

BlocklyDuino手動插入新增自訂積木程式碼避免覆蓋其他自訂積木設定

將程式碼區塊複製插入在BlocklyDuino3\package.nw\js\init.js檔此行程式碼之前

Blockly.inject(document.getElementById('content_blocks'),{



```
//插入自訂積木程式碼
var blocks_path = "./myBlocks/blocks.js";   //載入自訂積木定義檔
var javascript_path = "./myBlocks/javascript.js";   //載入自訂積木轉出程式碼檔
var toolbox_path = "./myBlocks/toolbox.xml";  //讀取自訂積木目錄檔
var en_path = "./myBlocks/en.js";  //載入自訂積木英文語系設定檔
var en_category_path = "./myBlocks/en_category.xml";  //載入自訂積木目錄英文語
var zhhant_path = "./myBlocks/zh-hant.js";  //載入自訂積木繁體語系設定檔(預設
var zhhant_category_path = "./myBlocks/zh-hant_category.xml";  //載入自訂積木

initial(blocks_path, javascript_path, toolbox_path);


function initial(blocks_path, javascript_path, toolbox_path) {
        addScript(blocks_path);
        addScript(javascript_path);
```

自訂積木撰寫流程 (BlocklyDuino)

Blockly Developer Tools 製作自訂積木：
可匯出blocks.js, javascript.js, toolbox.xml 三個積木主要檔案。

myBlocks\blocks.js編輯：
加入匯出blocks.js內的所有程式碼。

myBlocks\blocks.js編輯：
固定文字修改成動態語系變數，變數格式為"Blockly.Msg.自訂名稱"。例如固定文字"pin"更改成動態語系變數Blockly.Msg.pin

myBlocks\en.js(英文), myBlocks\zh-hant.js(繁中)編輯：
對應myBlocks\blocks.js內"Blockly.Msg.自訂名稱" 動態語系變數，分別設定變數不同語系文字值。

myBlocks\javascript.js編輯：
加入匯出javascript.js內的所有程式碼。

myBlocks\javascript.js編輯：
修改程式碼 var code = "...;\n";或var code = "...";結合積木內部取得變數值或函式利用字串組合產出程式碼內容。可在此新增額外的程式碼插入程式最上方或setup(){}區塊中。若有使用到Arduino IDE未內建的函式庫，須於資料夾BlocklyDuino3\arduino-1.8.5\libraries\內添加有使用到的函式庫資料夾。

myBlocks\blockly\toolbox.xml編輯：
參考範例檔主目錄與子目錄格式，可新增主目錄或子目錄。貼入匯出toolbox.xml內僅區塊<block...></block>的程式碼至指定的主目錄或子目錄。

myBlocks\en_category.xml (英文), myBlocks\zh-hant_category.xml.js(繁中)編輯：
目錄結構檔myBlocks\toolbox.xml中
<category id="..." name="目錄NAME"...></category> ，
目錄語系檔中
<category><name>目錄NAME</name><replace>語系文字</replace></category>。

目錄結構檔myBlocks\toolbox.xml中
< category id="..." name="myBlocks"></category> ，
目錄語系檔中zh-hant_category.xml.js
<category><name>myBlocks</name><replace>我的積木</replace></category>

積木定義檔 blocks.js (Blocklyduino)

```
Blockly.Blocks['digitalwrite'] = {  init: function() {
  this.appendDummyInput()
    .appendField("digitalWrite");    ➔  .appendField(Blockly.Msg.digitalWrite);
  this.appendValueInput("pin")           修改為動態語系變數
    .setCheck("Number")
    .appendField("(");
  this.appendDummyInput()
    .appendField(",")
    .appendField(new Blockly.FieldDropdown([["HIGH","1"], ["LOW","0"]]), "mode")
    .appendField(")");
this.setInputsInline(true);
this.setPreviousStatement(true, null);
this.setNextStatement(true, null);
this.setColour(255);
this.setTooltip("digitalWrite(pin, value)");
this.setHelpUrl("https://www.arduino.cc/reference/en/language/functions/digital-io/digitalwrite/");
}};
```

積木定義檔 blocks.js (Blocklyduino)

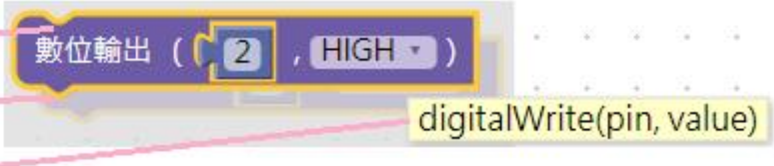
```
Blockly.Blocks['digitalwrite'] = {
  init: function() {
    this.appendDummyInput()  動態語系文字
        .appendField(Blockly.Msg.digitalWrite);
    this.appendValueInput("pin")
        .setCheck("Number")            自訂名稱
        .appendField("("); 固定文字
    this.appendDummyInput() 數值輸入
        .appendField(",")  固定文字              選單文字  值     選單文字  值
        .appendField(new Blockly.FieldDropdown([["HIGH","1"], ["LOW","0"]]), "mode") 下拉選單
        .appendField(")"); 固定文字
輸  this.setInputsInline(true); 不換行顯示
出  this.setPreviousStatement(true, null);
指  this.setNextStatement(true, null);
令  this.setColour(255); 積木顏色
    this.setTooltip("digitalWrite(pin, value)");
    this.setHelpUrl("https://www.arduino.cc/reference/en/language/functions/digital-io/digitalwrite/");
  }
};
```

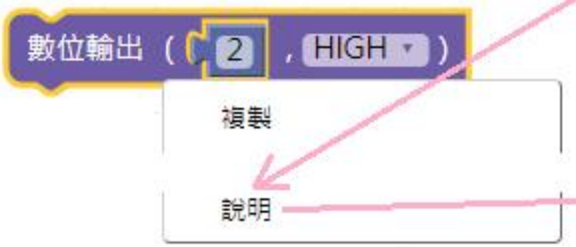
en.js

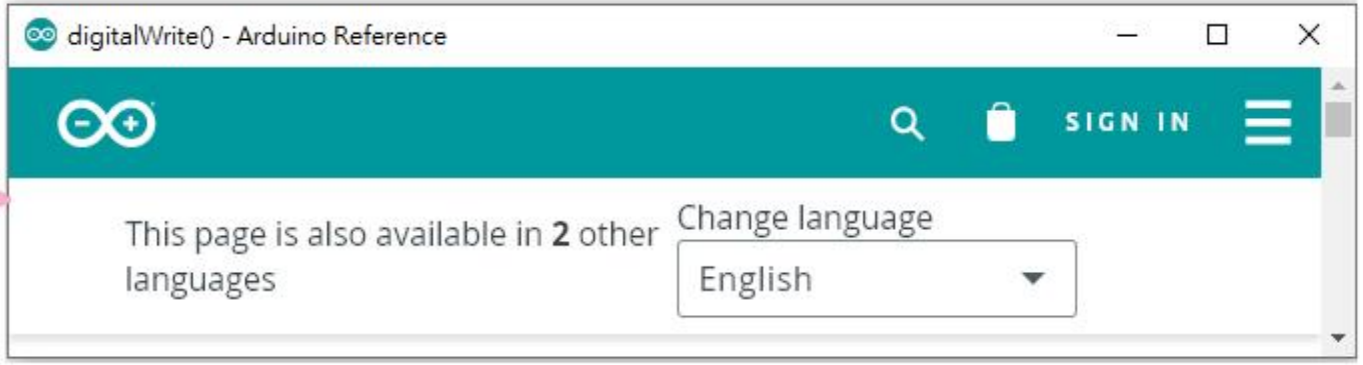
Blockly.Msg.digitalWrite = "Digital Write";

zh-hant.js

Blockly.Msg.digitalWrite = "數位輸出";





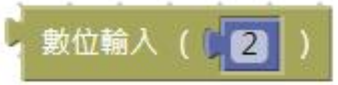


This page is also available in **2** other languages

Change language

English

輸出值 this.setOutput(true, "Number");



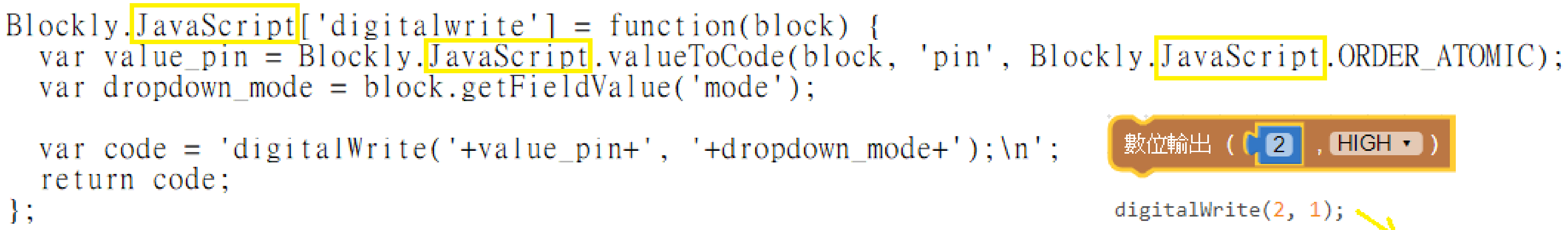
程式碼產出檔 javascript.js (Blocklyduino)

```
Blockly.JavaScript['digitalwrite'] = function(block) {
  var value_pin = Blockly.JavaScript.valueToCode(block, 'pin', Blockly.JavaScript.ORDER_ATOMIC);
  var dropdown_mode = block.getFieldValue('mode');

  // TODO: Assemble JavaScript into code variable.
  var code = '...;\n';
  return code;
};
```

將程式碼中所有Javascript取代為Ardunio，將積木值或函式以字串合併方式撰寫code值輸出程式碼。可在此新增額外的程式碼插入程式最上方或setup(){}區塊中。

```
Blockly.Arduino['digitalwrite'] = function(block) {
  var value_pin = Blockly.Arduino.valueToCode(block, 'pin', Blockly.Arduino.ORDER_ATOMIC);
  var dropdown_mode = block.getFieldValue('mode');

 // TODO: Assemble Arduino into code variable.
  var code = 'digitalWrite('+value_pin+', '+dropdown_mode+');\n' ;   //可結合函式庫
  return code;
};
```

程式碼產出檔 javascript.js (Blocklyduino)

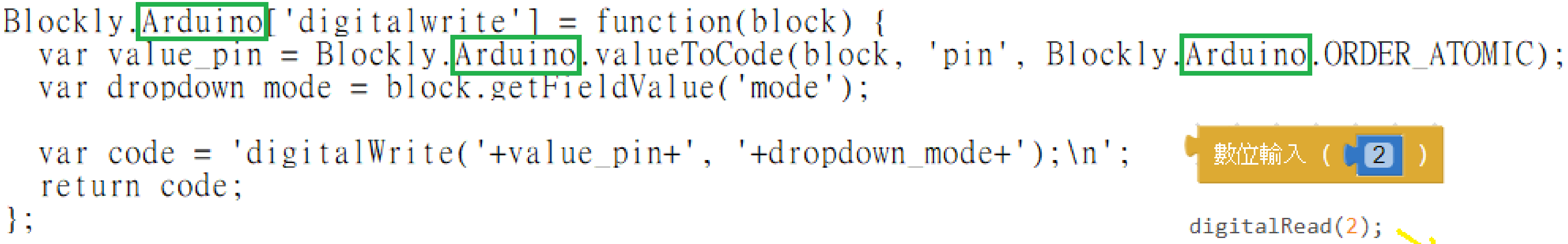


```
Blockly.JavaScript['digitalwrite'] = function(block) {
  var value_pin = Blockly.JavaScript.valueToCode(block, 'pin', Blockly.JavaScript.ORDER_ATOMIC);
  var dropdown_mode = block.getFieldValue('mode');

  var code = 'digitalWrite('+value_pin+', '+dropdown_mode+');\n';
  return code;
};
```

```
digitalWrite(2, 1);
```

Arduino函式庫

將Blockly Developer Tools 產生的程式碼所有"Javascript"取代為"Arduino"

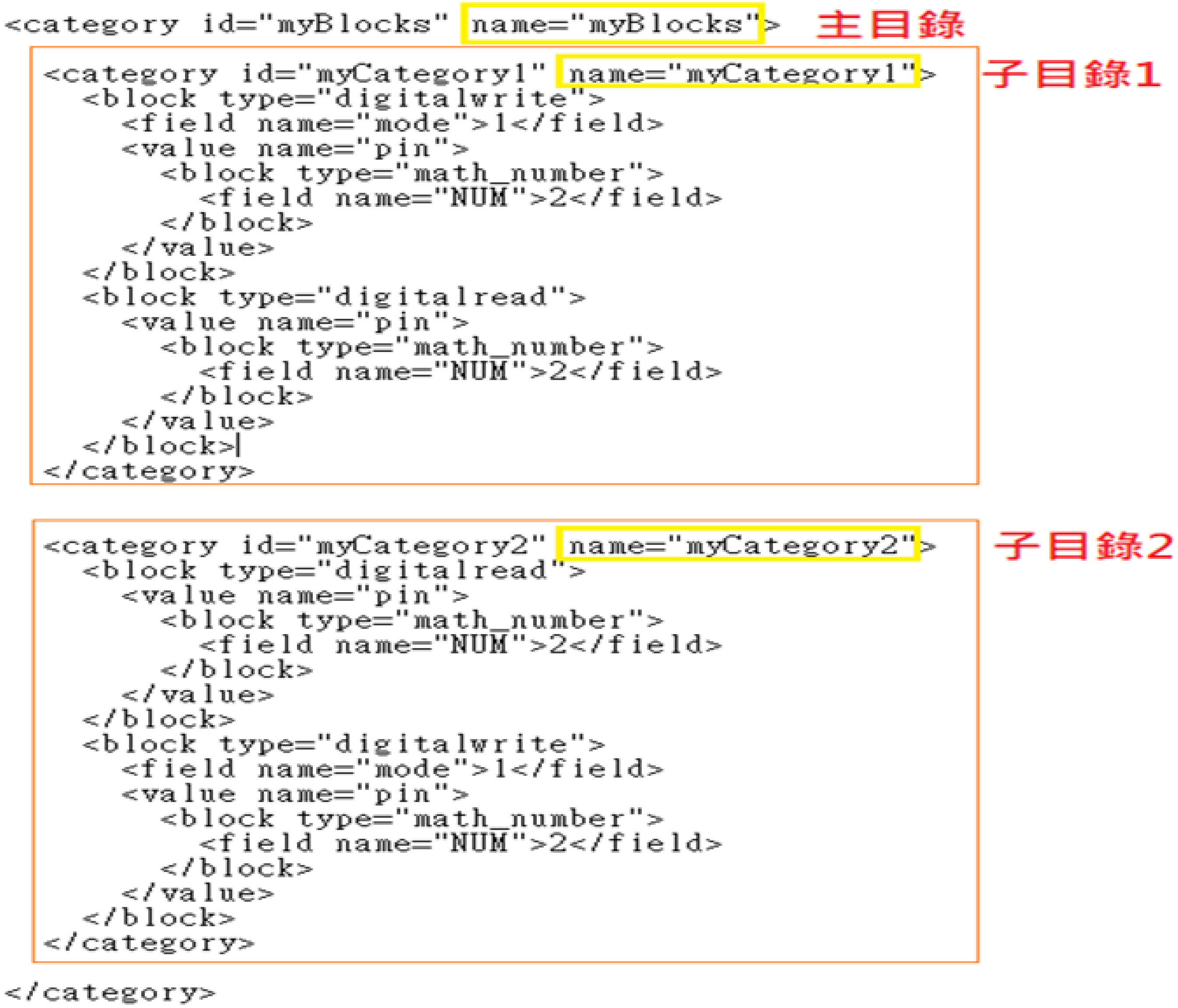


```
Blockly.Arduino['digitalwrite'] = function(block) {
  var value_pin = Blockly.Arduino.valueToCode(block, 'pin', Blockly.Arduino.ORDER_ATOMIC);
  var dropdown mode = block.getFieldValue('mode');

  var code = 'digitalWrite('+value_pin+', '+dropdown_mode+');\n';
  return code;
};
```

```
digitalRead(2);
```

Arduino函式庫

目錄結構檔 toolbox.xml (BlocklyDuino) 對應name



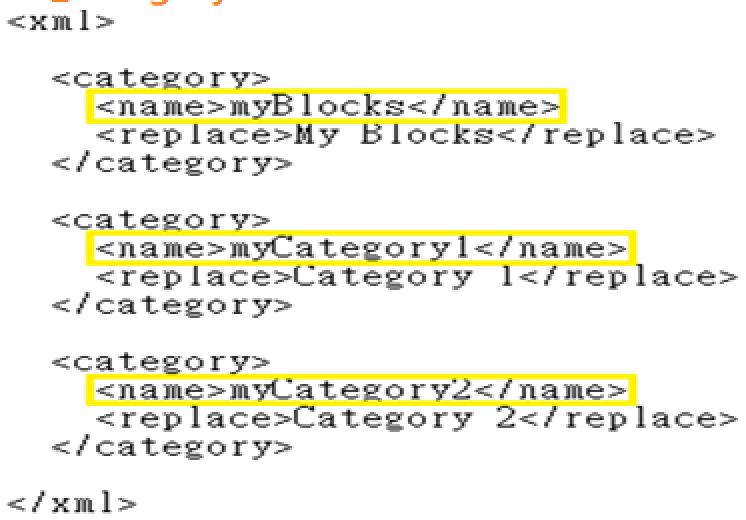
```
<category id="myBlocks" name="myBlocks">
  <category id="myCategory1" name="myCategory1">
    <block type="digitalwrite">
      <field name="mode">1</field>
      <value name="pin">
        <block type="math_number">
          <field name="NUM">2</field>
        </block>
      </value>
    </block>
    <block type="digitalread">
      <value name="pin">
        <block type="math_number">
          <field name="NUM">2</field>
        </block>
      </value>
    </block>
  </category>

  <category id="myCategory2" name="myCategory2">
    <block type="digitalread">
      <value name="pin">
        <block type="math_number">
          <field name="NUM">2</field>
        </block>
      </value>
    </block>
    <block type="digitalwrite">
      <field name="mode">1</field>
      <value name="pin">
        <block type="math_number">
          <field name="NUM">2</field>
        </block>
      </value>
    </block>
  </category>
</category>
```

主目錄

子目錄1

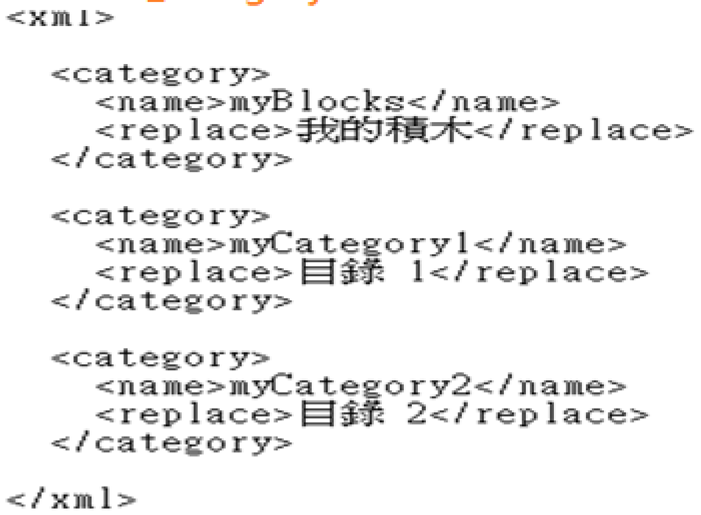
子目錄2

en_category.xml



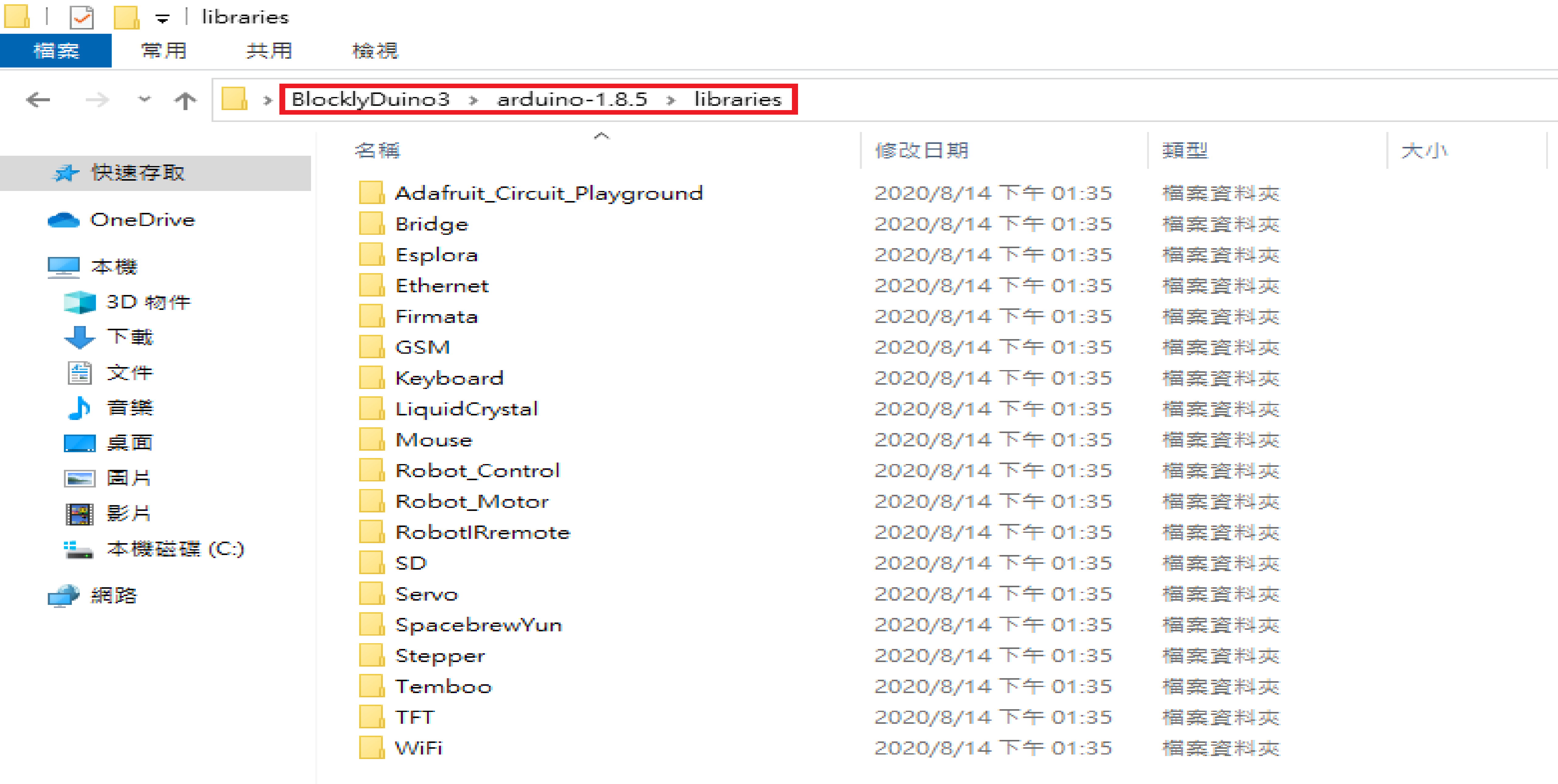
```
<xml>

  <category>
    <name>myBlocks</name>
    <replace>My Blocks</replace>
  </category>

  <category>
    <name>myCategory1</name>
    <replace>Category 1</replace>
  </category>

  <category>
    <name>myCategory2</name>
    <replace>Category 2</replace>
  </category>

</xml>
```

zh-hant_category.xml



```
<xml>

  <category>
    <name>myBlocks</name>
    <replace>我的積木</replace>
  </category>

  <category>
    <name>myCategory1</name>
    <replace>目錄 1</replace>
  </category>

  <category>
    <name>myCategory2</name>
    <replace>目錄 2</replace>
  </category>

</xml>
```

自訂積木所需新增函式庫置放目錄(BlocklyDuino)

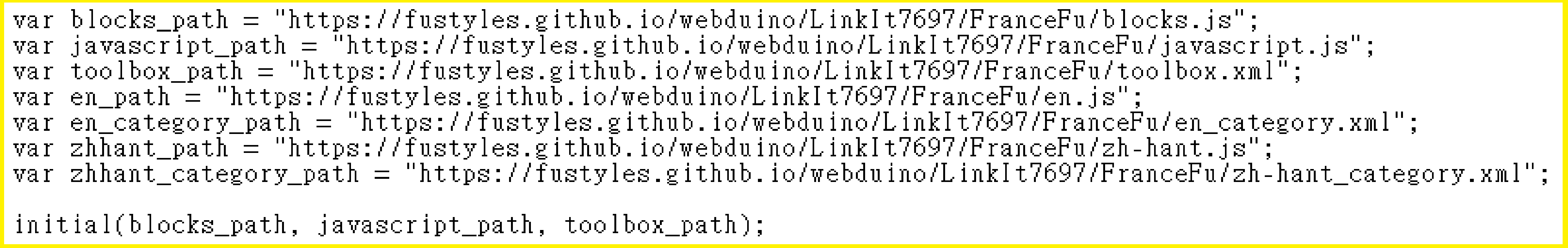


BlocklyDuino3 › arduino-1.8.5 › libraries

名稱	修改日期	類型	大小
Adafruit_Circuit_Playground	2020/8/14 下午 01:35	檔案資料夾	
Bridge	2020/8/14 下午 01:35	檔案資料夾	
Esplora	2020/8/14 下午 01:35	檔案資料夾	
Ethernet	2020/8/14 下午 01:35	檔案資料夾	
Firmata	2020/8/14 下午 01:35	檔案資料夾	
GSM	2020/8/14 下午 01:35	檔案資料夾	
Keyboard	2020/8/14 下午 01:35	檔案資料夾	
LiquidCrystal	2020/8/14 下午 01:35	檔案資料夾	
Mouse	2020/8/14 下午 01:35	檔案資料夾	
Robot_Control	2020/8/14 下午 01:35	檔案資料夾	
Robot_Motor	2020/8/14 下午 01:35	檔案資料夾	
RobotIRremote	2020/8/14 下午 01:35	檔案資料夾	
SD	2020/8/14 下午 01:35	檔案資料夾	
Servo	2020/8/14 下午 01:35	檔案資料夾	
SpacebrewYun	2020/8/14 下午 01:35	檔案資料夾	
Stepper	2020/8/14 下午 01:35	檔案資料夾	
Temboo	2020/8/14 下午 01:35	檔案資料夾	
TFT	2020/8/14 下午 01:35	檔案資料夾	
WiFi	2020/8/14 下午 01:35	檔案資料夾	

package.nw\js\Init.js檔編輯新增數個自訂積木 (BlocklyDuino)

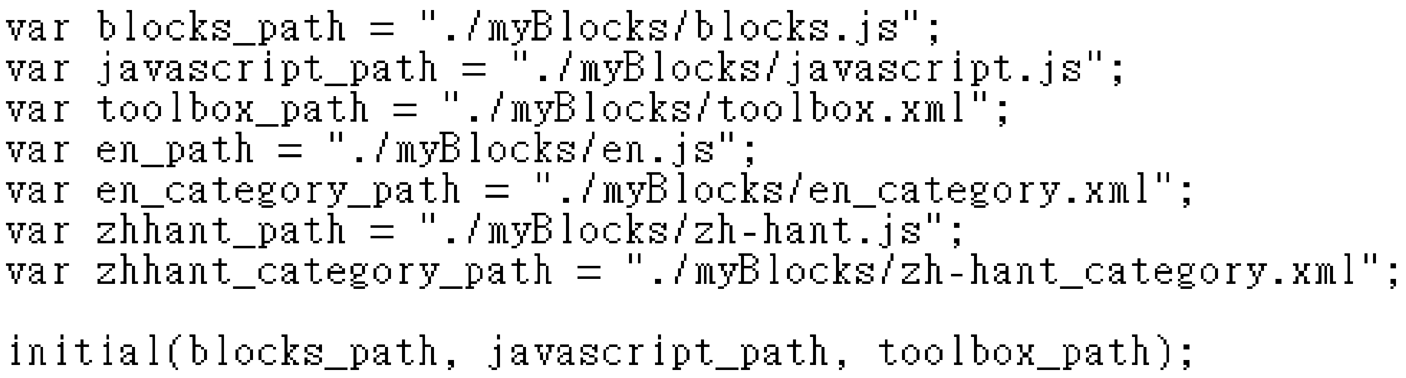
法蘭斯自訂積木



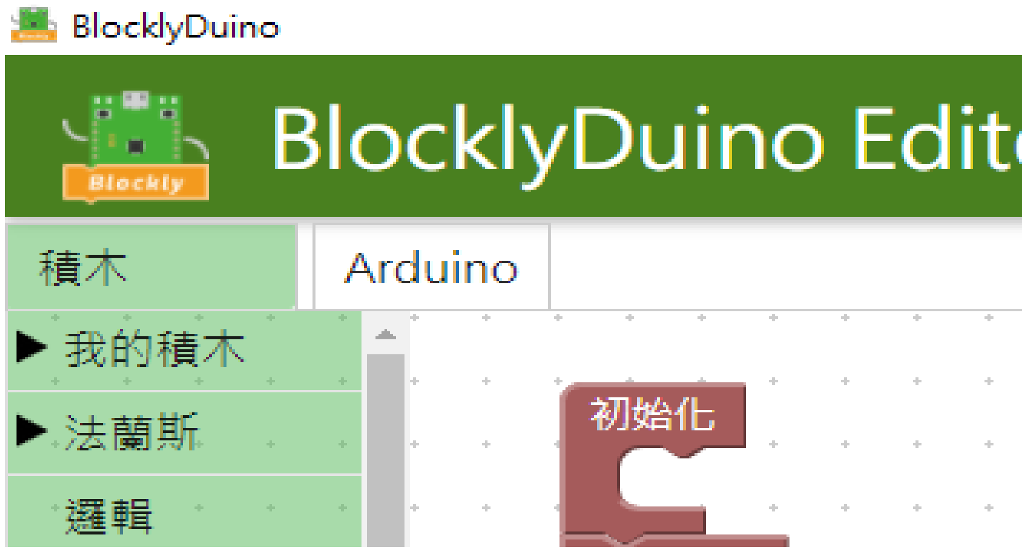
```
var blocks_path = "https://fustyles.github.io/webduino/LinkIt7697/FranceFu/blocks.js";
var javascript_path = "https://fustyles.github.io/webduino/LinkIt7697/FranceFu/javascript.js";
var toolbox_path = "https://fustyles.github.io/webduino/LinkIt7697/FranceFu/toolbox.xml";
var en_path = "https://fustyles.github.io/webduino/LinkIt7697/FranceFu/en.js";
var en_category_path = "https://fustyles.github.io/webduino/LinkIt7697/FranceFu/en_category.xml";
var zhhant_path = "https://fustyles.github.io/webduino/LinkIt7697/FranceFu/zh-hant.js";
var zhhant_category_path = "https://fustyles.github.io/webduino/LinkIt7697/FranceFu/zh-hant_category.xml";

initial(blocks_path, javascript_path, toolbox_path);
```

可重複新增此區塊程式碼設定自訂積木路徑載入多個自訂積木

我的自訂積木



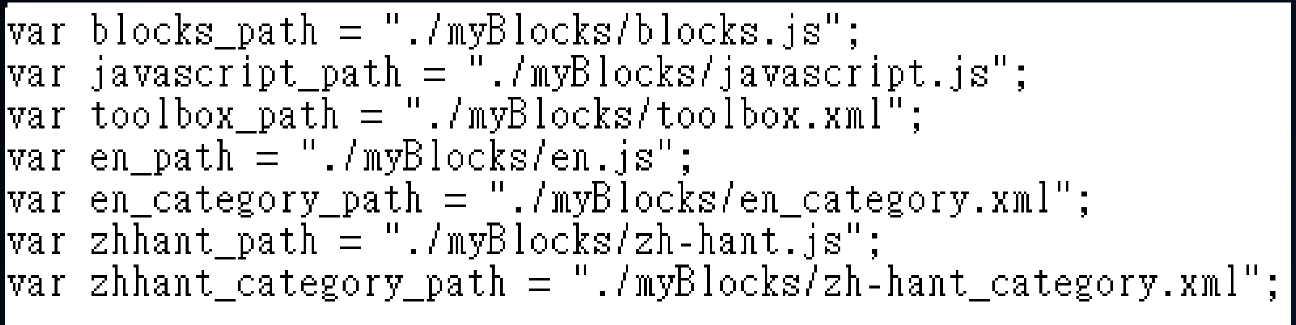
```
var blocks_path = "./myBlocks/blocks.js";
var javascript_path = "./myBlocks/javascript.js";
var toolbox_path = "./myBlocks/toolbox.xml";
var en_path = "./myBlocks/en.js";
var en_category_path = "./myBlocks/en_category.xml";
var zhhant_path = "./myBlocks/zh-hant.js";
var zhhant_category_path = "./myBlocks/zh-hant_category.xml";

initial(blocks_path, javascript_path, toolbox_path);
```



自訂積木路徑設定 (BlocklyDuino)

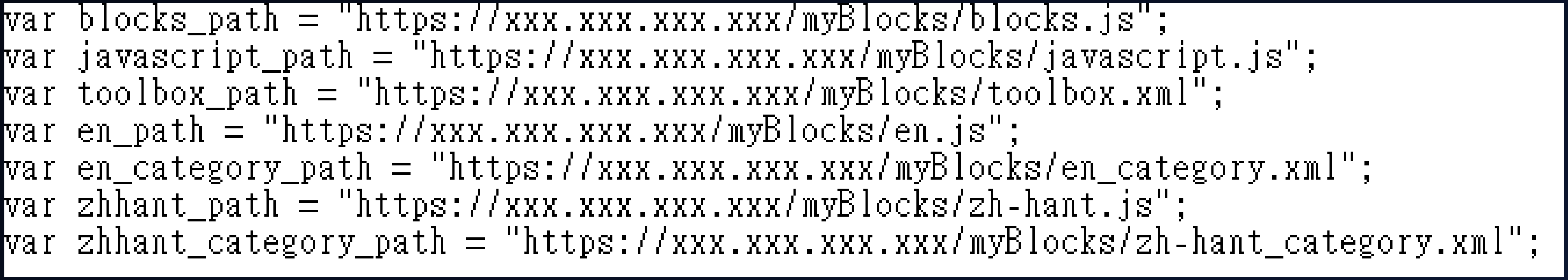
開啟package.nw\js\init.js編輯將本機路徑修改成遠端路徑

本機路徑



```
var blocks_path = "./myBlocks/blocks.js";
var javascript_path = "./myBlocks/javascript.js";
var toolbox_path = "./myBlocks/toolbox.xml";
var en_path = "./myBlocks/en.js";
var en_category_path = "./myBlocks/en_category.xml";
var zhhant_path = "./myBlocks/zh-hant.js";
var zhhant_category_path = "./myBlocks/zh-hant_category.xml";
```

遠端路徑



```
var blocks_path = "https://xxx.xxx.xxx.xxx/myBlocks/blocks.js";
var javascript_path = "https://xxx.xxx.xxx.xxx/myBlocks/javascript.js";
var toolbox_path = "https://xxx.xxx.xxx.xxx/myBlocks/toolbox.xml";
var en_path = "https://xxx.xxx.xxx.xxx/myBlocks/en.js";
var en_category_path = "https://xxx.xxx.xxx.xxx/myBlocks/en_category.xml";
var zhhant_path = "https://xxx.xxx.xxx.xxx/myBlocks/zh-hant.js";
var zhhant_category_path = "https://xxx.xxx.xxx.xxx/myBlocks/zh-hant_category.xml";
```

javascript.js實用技巧 (BlocklyDuino)

置於程式碼最上方

```
Blockly.Arduino.definitions_['自訂名稱'] = '#include <函式庫名稱.h>';
```

取得目前Setup(){} 區塊內程式碼

```
var statements_setup = Blockly.Arduino.statementToCode(block, 'setup');
```

取得目前loop(){} 區塊內程式碼

```
var statements_loop = Blockly.Arduino.statementToCode(block, 'loop');
```

取得某statements_input區塊內程式碼

```
var statements_custom = Blockly.Arduino.statementToCode(block, 'NAME值');
```

插入程式碼置於Setup(){}區塊內

```
Blockly.Arduino.setups_['自訂名稱]= "pinMode(2, OUTPUT);\n";
```

插入程式碼置於Setup(){}區塊內最前方

```
Blockly.Arduino.setups_.manual_add = "pinMode(2, OUTPUT);\n" + statements_setup;
```

插入程式碼置於Setup(){}區塊內最後方

```
Blockly.Arduino.setups_.manual_add = statements_setup + "pinMode(2, OUTPUT);\n" ;
```

javascript.js範例01 (BlocklyDuino)

```
Blockly.Arduino['digitalwrite'] = function(block) {
 //置於程式碼最上方
 Blockly.Arduino.definitions_['LinkIt_wifi'] = '#include <LWiFi.h>';

 //取得輸入值
 var value_pin = Blockly.Arduino.valueToCode(block, 'pin', Blockly.Arduino.ORDER_ATOMIC);
 //取得下拉選單值
 var dropdown_mode = block.getFieldValue('mode');

 //插入程式碼置於Setup(){}區塊內
 Blockly.Arduino.setups_["setup_digitalwrite_"+value_pin]="pinMode("+value_pin+", "+dropdown_mode+");";

 //輸出產生的程式碼
 var code = 'digitalWrite('+value_pin+', '+dropdown_mode+');\n';
 return code;
};
```

進階控制技巧

https://blockly-demo.appspot.com/static/tests/playground.html?dir=ltr&toolbox=test-blocks

法蘭斯自訂積木 (Webduino)

法蘭斯點矩陣
https://fustyles.github.io/webduino/EDU_bit_MatrixLed_20190827/blockly.json

遊戲元素

https://fustyles.github.io/webduino/GameElements_20190131/blockly.json

Javascript 指令擴充
https://fustyles.github.io/webduino/Instruction_20181213/blockly.json

朗讀語言擴充

https://fustyles.github.io/webduino/EDU_speak_setting/blockly.json

監看程式碼 (首頁開啟開發人員工具)

https://fustyles.github.io/webduino/ShowCode_20181216/blockly.json

姿態辨識 (tfjs posenet)
https://fustyles.github.io/webduino/posenet_20190822/blockly.json

物件辨識(tfjs mobilenet)
https://fustyles.github.io/webduino/mobilenet_20190821/blockly.json

物件辨識(tfjs coco-ssd)
https://fustyles.github.io/webduino/coco-ssd_20190821/blockly.json

臉部偵測 (tfjs face-api.js)
https://fustyles.github.io/webduino/faceapi_20200124/blockly.json

臉部辨識 (tfjs face-api.js)
https://fustyles.github.io/webduino/faceapi_20200402/blockly.json

深度學習 (tfjs KNN-Classifier)
https://fustyles.github.io/webduino/knn-classifier_20190608/blockly.json

語音辨識 (google)
https://fustyles.github.io/webduino/SpeechRecognition_20191225/blockly.json

圖像分割 (tfjs deeplab)
https://fustyles.github.io/webduino/deeplab_20200125/blockly.json

身體偵測 (tfjs bodypix2)
https://fustyles.github.io/webduino/bodypix2_20200125/blockly.json

身體偵測 (tfjs bodypix1)
https://fustyles.github.io/webduino/bodypix1_20200125/blockly.json

人臉偵測 (Tracking.js)
https://fustyles.github.io/webduino/Tracking_20190917/blockly.json

顏色偵測 (Tracking.js)
https://fustyles.github.io/webduino/Tracking_20200625/blockly.json

手勢偵測 (tfjs handpose)
https://fustyles.github.io/webduino/handpose_20200614/blockly.json

臉部網格偵測 (tfjs facemesh)
https://fustyles.github.io/webduino/Facemesh_20200626/blockly.json

臉部偵測 (tfjs brazeface)
https://fustyles.github.io/webduino/Blazeface_20200627/blockly.json

機械學習 (tfjs Machine Learning)
https://fustyles.github.io/webduino/teachablemachine_20200729/blockly.json

文字偵測 (tesseract.js)
https://fustyles.github.io/webduino/tesseract.js_20200615/blockly.json

Line Bot
https://fustyles.github.io/webduino/LineBot_20181027/blockly.json

Telegram Bot
https://fustyles.github.io/webduino/Telegram_20200809/blockly.json

QR code辨識 (instascan)
https://fustyles.github.io/webduino/instascan.js_20200824/blockly.json

物件辨識(Microsoft Azure Custom Vision)
https://fustyles.github.io/webduino/Azure_ClassifyImage_20190901/blockly.json

物件辨識(Microsoft Azure Custom Vision + TFJS)
https://fustyles.github.io/webduino/Azure_customvision-tfjs_20200128/blockly.json

臉部辨識(Microsoft Azure Face API – Face Detect)
https://fustyles.github.io/webduino/Azure_FaceDetection_20190901/blockly.json

尋找相似臉(Microsoft Azure Face API - Find Similar Face)
https://fustyles.github.io/webduino/Azure_FaceFindSimilar_20191117/blockly.json

驗證同一人(Microsoft Azure Face API API – Verify Face To Face)
https://fustyles.github.io/webduino/ Azure_FaceToFaceVerify_20191118/blockly.json

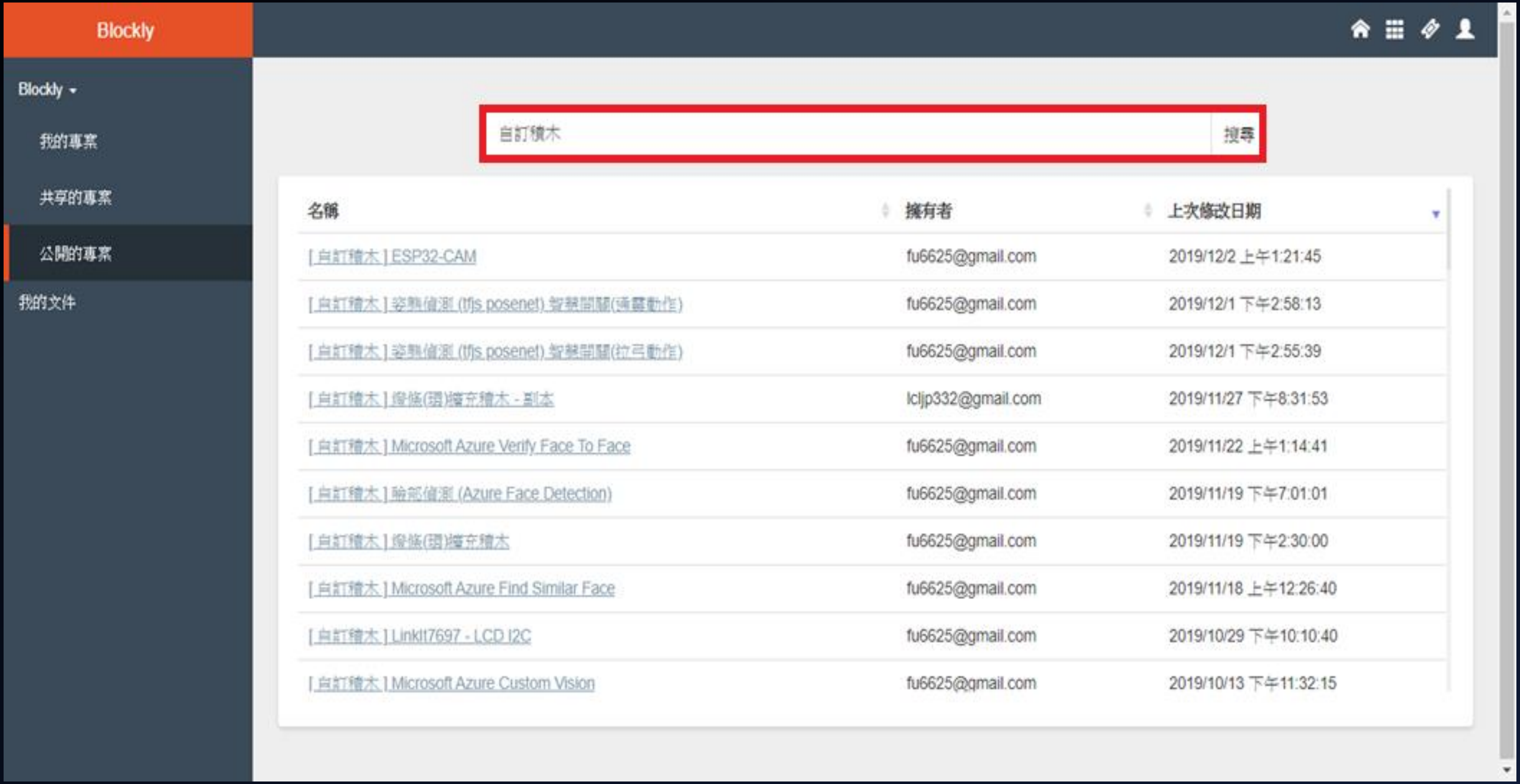
ESP32-CAM (雲端平台網址須由https改成http)
https://fustyles.github.io/webduino/ESP32-CAM_20191201/blockly.json
[韌體]
https://github.com/fustyles/Arduino/tree/master/ESP32-CAM_MyBlockly_JSON

WebBit (雲端平台網址須由https改成http)
https://fustyles.github.io/webduino/WebBit_20190225/blockly.json
[韌體]
https://github.com/fustyles/Arduino/blob/master/WebBit_ESP32_MyBlockly_JSON.ino

WiFiBoard (ESP32 、 LinkIt7697)
https://fustyles.github.io/webduino/ESP8266_20190128/blockly.json
[ESP32韌體]
https://github.com/fustyles/Arduino/blob/master/ESP32_MyBlockly_JSON.ino
[LinkIt7697韌體]
https://github.com/fustyles/Arduino/blob/master/LinkIt7697_MyBlockly_JSON.ino

自訂積木範例 https://github.com/fustyles/webduino

雲端平台搜尋「自訂積木」



名稱	擁有者	上次修改日期
[自訂積木] ESP32-CAM	fu6625@gmail.com	2019/12/2 上午1:21:45
[自訂積木] 姿勢偵測 (tfjs posenet) 智慧開關(揮舞動作)	fu6625@gmail.com	2019/12/1 下午2:58:13
[自訂積木] 姿勢偵測 (tfjs posenet) 智慧開關(拉弓動作)	fu6625@gmail.com	2019/12/1 下午2:55:39
[自訂積木] 燈條(環)擴充積木 - 副本	lcljp332@gmail.com	2019/11/27 下午8:31:53
[自訂積木] Microsoft Azure Verify Face To Face	fu6625@gmail.com	2019/11/22 上午1:14:41
[自訂積木] 臉部偵測 (Azure Face Detection)	fu6625@gmail.com	2019/11/19 下午7:01:01
[自訂積木] 燈條(環)擴充積木	fu6625@gmail.com	2019/11/19 下午2:30:00
[自訂積木] Microsoft Azure Find Similar Face	fu6625@gmail.com	2019/11/18 上午12:26:40
[自訂積木] LinkIt7697 - LCD I2C	fu6625@gmail.com	2019/10/29 下午10:10:40
[自訂積木] Microsoft Azure Custom Vision	fu6625@gmail.com	2019/10/13 下午11:32:15