



KIO4.COM

[Inicio](#)
[Tweet](#)
[FOROS](#)
[Elastix - VoIP](#)
[Estación meteorológica](#)
[Raspberry Pi](#)
[AutoIt \(Programación\)](#)
[B4A \(Basic4Android\)](#)
[B4J \(Basic4Java\)](#)
[Visual Basic Script \(VBS\)](#)
[Visual Basic](#)
[App inventor 2](#)
[ADB Shell - Android](#)
[Firebase \(BD autoactualizable\)](#)
[Cosas de Windows](#)
[Translate:](#)

[PHP - MySQL](#)
[Arduino](#)
[NodeMCU como Arduino](#)
[Webs interesantes](#)
[Búsqueda en este sitio:](#)



Hola amigos, rellena el formulario para recibir los tutoriales correspondientes al curso de Ap

Internet de las Cosas

Create Extensions App inventor 2

Build Extensions App Inventor, easy tutorial - Juan Antonio Villalpando

-- Tutorial de iniciación de App Inventor 2 en español --

[Index tutorial](#)

125B.- Extensions. Pythagoras theorem. Join texts.

- Create an extension to calculate [Pythagoras theorem](#).



0.- In our PC-Windows, we create a folder in disk C:\, call [AppInventorExtensiones](#). In this folder download three packets:

Nombre	Tipo	Tamaño
jdk-8u131-windows-x64.exe	Aplicación	202.784 KB
Git-2.13.2-32-bit.exe	Aplicación	37.670 KB
apache-ant-1.10.1-bin.zip	Archivo WinRAR ZIP	8.618 KB

1.- [Java SE Development Kit](#), I have gone down 8u131 version ([jdk-8u131-windows-x64.exe](#) of 198 MB). [you can download other [recent version](#)].

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

Java SE Development Kit 8u131

You must accept the Oracle Binary Code License Agreement for Java SE to download this software.

Accept License Agreement Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.87 MB	jdk-8u131-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.81 MB	jdk-8u131-linux-arm64-vfp-hflt.tar.gz
Linux x86	164.66 MB	jdk-8u131-linux-i586.rpm
Linux x86	179.39 MB	jdk-8u131-linux-i586.tar.gz
Linux x64	162.11 MB	jdk-8u131-linux-x64.rpm
Linux x64	176.95 MB	jdk-8u131-linux-x64.tar.gz
Mac OS X	226.57 MB	jdk-8u131-macosx-x64.dmg
Solaris SPARC 64-bit	139.79 MB	jdk-8u131-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.13 MB	jdk-8u131-solaris-sparcv9.tar.gz
Solaris x64	140.51 MB	jdk-8u131-solaris-x64.tar.Z
Solaris x64	96.96 MB	jdk-8u131-solaris-x64.tar.gz
Windows x86	191.22 MB	jdk-8u131-windows-i586.exe
Windows x64	198.03 MB	jdk-8u131-windows-x64.exe

2.- **Apache Ant** is a JAVA library for build applications, <http://ant.apache.org/bindownload.cgi>, download: **Ant 1.10.1 (Binary Distributions)** ([apache-ant-1.10.1-bin.zip](#) of 8,6 MB)

ant.apache.org/bindownload.cgi

Current Release of Ant

The Apache Ant team currently maintains two lines of development. The 1.9.x releases require 1.10.x requires Java8 at runtime. Both lines are based off of Ant 1.9.7 and the 1.9.x releases while additional new features are developed for 1.10.x. We recommend using 1.10.1 required to use versions of Java prior to Java8 during the build process.

Currently, Apache Ant 1.9.9 and 1.10.1 are the best available versions, see the [release notes](#)

Note
Ant 1.9.9 and 1.10.1 have been released on 06-Feb-2017 and may not be available on all mirrors for

Tar files may require gnu tar to extract
Tar files in the distribution contain long file names, and may require gnu tar to do the extraction.

1.10.1 .zip archive: [apache-ant-1.10.1-bin.zip](#) [PGP] [SHA1] [SHA512] [MD5]

- 1.9.9 .zip archive: [apache-ant-1.9.9-bin.zip](#) [PGP] [SHA1] [SHA512] [MD5]
- 1.10.1 .tar.gz archive: [apache-ant-1.10.1-bin.tar.gz](#) [PGP] [SHA1] [SHA512] [MD5]
- 1.9.9 .tar.gz archive: [apache-ant-1.9.9-bin.tar.gz](#) [PGP] [SHA1] [SHA512] [MD5]
- 1.10.1 .tar.bz2 archive: [apache-ant-1.10.1-bin.tar.bz2](#) [PGP] [SHA1] [SHA512] [MD5]
- 1.9.9 .tar.bz2 archive: [apache-ant-1.9.9-bin.tar.bz2](#) [PGP] [SHA1] [SHA512] [MD5]
- 1.10.1 .tar.xz archive: [apache-ant-1.10.1-bin.tar.xz](#) [PGP] [SHA1] [SHA512] [MD5]

3.- **Git bash** of <https://git-scm.com/download/win> (**Git-2.13.2-32-bit.exe** of 32 MB). This is as LINUX terminal to run **ant**.

[I have gone down 32-bit, if you download 64-bit also works.]

Es seguro | <https://git-scm.com/download/win>

git --local-branching-on-the-cheap

Search entire site...

Downloading Git

Your download is starting...

You are downloading the latest (**2.13.3**) **64-bit** version of **Git for Windows**. This is the most recent **maintained build**. It was released **5 days ago**, on 2017-07-13.

If your download hasn't started, [click here to download manually](#).

Other Git for Windows downloads

- [Git for Windows Setup](#)
- [32-bit Git for Windows Setup](#)
- [64-bit Git for Windows Setup](#)

The entire **Pro Git book** written by Scott Chacon and

- These are our 3 packets in folder **C:\AppInventorExtensiones**.

Acer (C:) > AppInventorExtensiones >

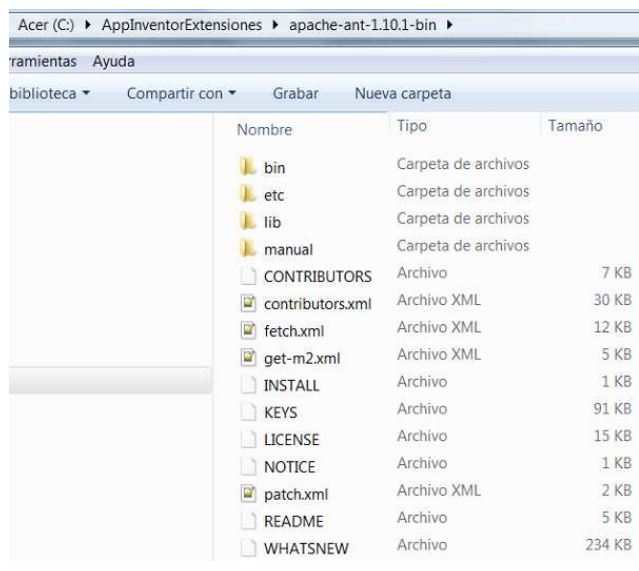
Herramientas Ayuda

biblioteca Compartir con Grabar Nueva carpeta

Nombre	Tipo	Tamaño
jdk-8u131-windows-x64.exe	Aplicación	202.784 KB
Git-2.13.2-32-bit.exe	Aplicación	37.670 KB
apache-ant-1.10.1-bin.zip	Archivo WinRAR ZIP	8.618 KB

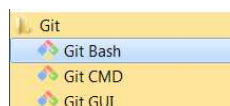
4.- Now unzip **Apache Ant**. When you unzip, perhaps get a double folder:

Usamos cookies propias y de terceros que entre otras cosas recogen datos sobre sus hábitos de navegación y realizan análisis de uso de nuestro sitio. Si continúa navegando consideramos que acepta su uso. **Acepto** Más información



5.- Now install **Git Bash**. All default installation.

- In **Windows Start button** you can look.

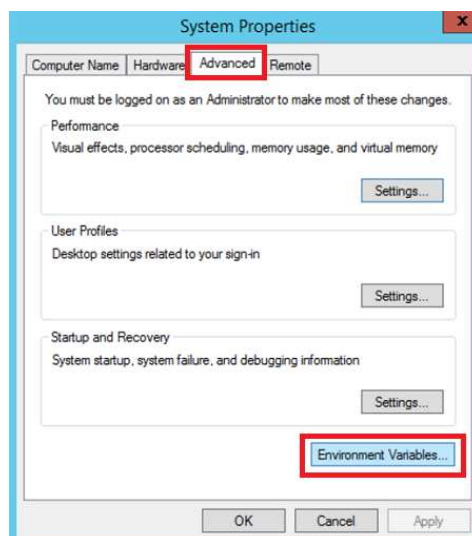


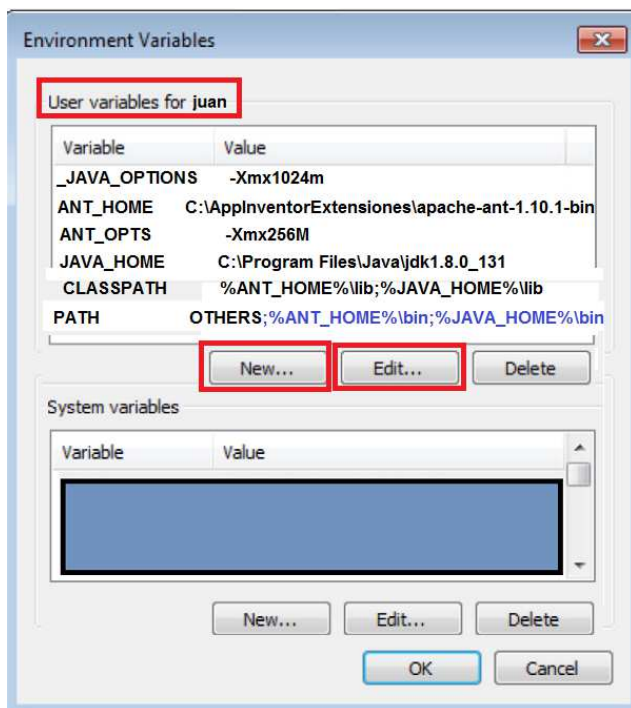
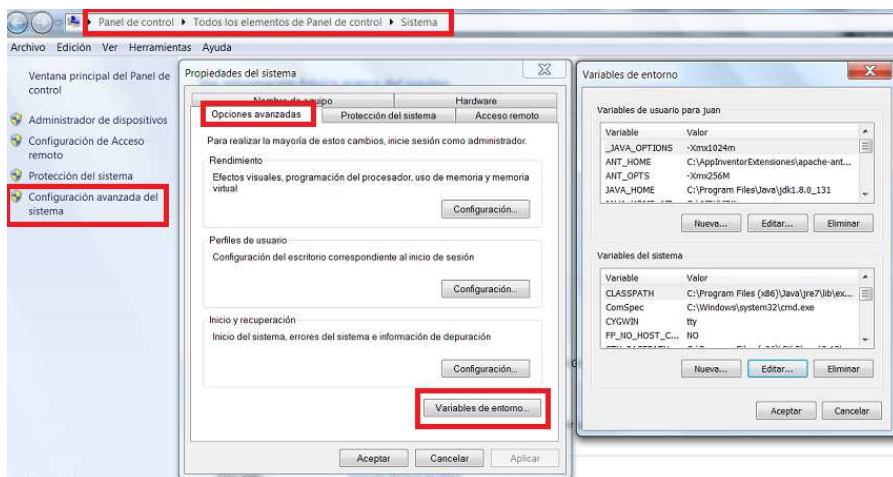
6.- Install **Java SE Development Kit**. Perhaps is necessary restart PC for installation.

- Environment Variables.

- Lets with Environment Variables:

Control Panel / System / Advanced System Settings / Advanced Options / Environment Variables...





- We can use "Edit..." or "New..." button.

- In **User variables for yourname**, by "New..." button we create this variables...

_JAVA_OPTIONS -----> -Xmx1024m

ANT_HOME -----> C:\AppInventorExtensiones\apache-ant-1.10.1-bin [folder unzip apache-ant]

ANT_OPTS -----> -Xmx256M

JAVA_HOME -----> C:\Program Files\Java\jdk1.8.0_131 [Look! is jdk NOT jdr]

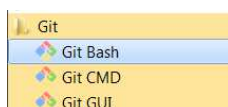
CLASSPATH -----> %ANT_HOME%\lib;%JAVA_HOME%\lib

En PATH ----- Edit -----> ;%ANT_HOME%\bin;%JAVA_HOME%\bin [Look! ;start with semicolon;]

- Create a clone of App Inventor in our PC.

- We create a clone of App Inventor direct from Internet.

- Run **Git Bash**



- write:

git clone <https://github.com/mit-cml/appinventor-sources.git>

```

MINGW32/c/Users/juan
juan@juan-PC MINGW32 ~
$ git clone https://github.com/mit-cml/appinventor-sources.git
Cloning into 'appinventor-sources'...
remote: Counting objects: 41191, done.
remote: Total 41191 (delta 0), reused 0 (delta 0), pack-reused 41190
Receiving objects: 100% (41191/41191), 553.30 MiB | 494.00 KiB/s, done.
Resolving deltas: 100% (21999/21999), done.
Checking out files: 100% (1758/1758), done.

juan@juan-PC MINGW32 ~
$

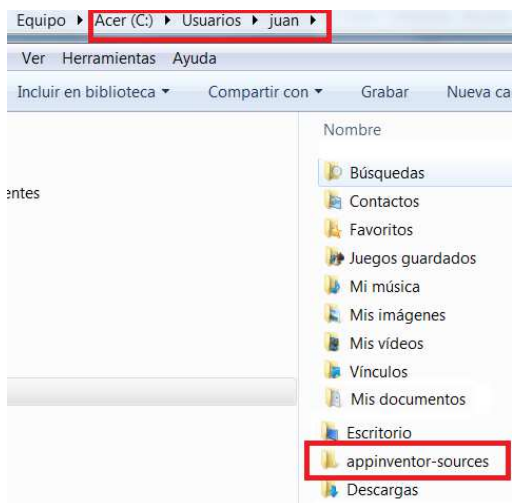
```

- In **5 or 10 minutes**, we get a folder: **appinventor-sources** with sources files of App Inventor.

- **Where is image App Inventor source?**

- Disk **C:**, folder **Users**, your name user, in my case **juan**, look this folder:

C:\Users\juan\appinventor-sources



- You can **pry** this sub-folders, look **many files with .java**

..\appinventor\components\src\com\google\appinventor\components\runtime

Nombre	Tipo	Tamaño
Button.java	Archivo JAVA	3 KB
ButtonBase.java	Archivo JAVA	23 KB
Camcorder.java	Archivo JAVA	5 KB
Camera.java	Archivo JAVA	9 KB
Canvas.java	Archivo JAVA	56 KB
CheckBox.java	Archivo JAVA	11 KB
Clock.java	Archivo JAVA	18 KB
Component.java	Archivo JAVA	6 KB
ComponentContainer.java	Archivo JAVA	2 KB
ContactPicker.java	Archivo JAVA	18 KB
DatePicker.java	Archivo JAVA	8 KB
Deleteable.java	Archivo JAVA	1 KB
EmailAddressAdapter.java	Archivo JAVA	6 KB
EmailPicker.java	Archivo JAVA	4 KB
Ev3ColorSensor.java	Archivo JAVA	16 KB
Ev3Commands.java	Archivo JAVA	11 KB
Ev3GyroSensor.java	Archivo JAVA	7 KB

- **Create extension. Pythagoras theorem.**

- **Careful with upper and lower case, it isn't equal Cathetus what cathetus.**

- Our first extension: **Pythagoras theorem.**

Pitagoras.java

```

package com.Pitagoras;
// package es.kio4.Pitagoras; // New version.

// @ Juan Antonio Villalpando
// kio4.com
// Creacion de extensiones. Junio 2017.
// Esta extension calcula la hipotenusa mediante el Teorema de Pitagoras.

import com.google.appinventor.components.annotations.DesignerComponent;
import com.google.appinventor.components.annotations.DesignerProperty;
import com.google.appinventor.components.annotations.PropertyCategory;
import com.google.appinventor.components.annotations.SimpleEvent;
import com.google.appinventor.components.annotations.SimpleFunction;
import com.google.appinventor.components.annotations.SimpleObject;
import com.google.appinventor.components.annotations.SimpleProperty;
import com.google.appinventor.components.common.ComponentCategory;
import com.google.appinventor.components.common.PropertyTypeConstants;
import com.google.appinventor.components.runtime.util.MediaUtil;
import com.google.appinventor.components.runtime.*;

@DesignerComponent(version = Pitagoras.VERSION,
    description = "Teorema de Pitagoras. " + "Juan Antonio Villalpando - KIO4.COM ",
    category = ComponentCategory.EXTENSION,
    nonVisible = true,
    iconName = "")
@SimpleObject(external = true)
public class Pitagoras extends AndroidNonvisibleComponent implements Component {

    public static final int VERSION = 1;
    public static final float DEFAULT_CATETO_A = 3f;
    public static final float DEFAULT_CATETO_B = 4f;
    private ComponentContainer container;
    private double cateto_a = 0;
    private double cateto_b = 0;

    public Pitagoras(ComponentContainer container) {
        super(container.$form());
        this.container = container;
        Cateto_A(DEFAULT_CATETO_A);
        Cateto_B(DEFAULT_CATETO_B);
    }

    // Obtener el valor.
    @SimpleProperty(category = PropertyCategory.BEHAVIOR)
    public double Cateto_A() {
        return cateto_a;
    }

    @SimpleProperty(category = PropertyCategory.BEHAVIOR)
    public double Cateto_B() {
        return cateto_b;
    }

    // Establecer el valor.
    @DesignerProperty(editorType = PropertyTypeConstants.PROPERTY_TYPE_NON_NEGATIVE_FLOAT, defaultValue = Pitagoras.DEFAULT_CATETO_A + "")
    @SimpleProperty(description = "Asigna el valor del cateto A. " + "El separador decimal es el punto.")
    public void Cateto A(double nuevoCateto_A) {
        this.cateto_a = nuevoCateto_A;
    }

    @DesignerProperty(editorType = PropertyTypeConstants.PROPERTY_TYPE_NON_NEGATIVE_FLOAT, defaultValue = Pitagoras.DEFAULT_CATETO_B + "")
    @SimpleProperty(description = "Asigna el valor del cateto B. " + "El separador decimal es el punto.")
    public void Cateto B(double nuevoCateto_B) {
        this.cateto_b = nuevoCateto_B;
    }

    // Funcion para calcular la hipotenusa.
    @SimpleFunction(description = "Introduce los dos catetos y obtendras la hipotenusa.")
    public double Pitagoras(double catetoA, double catetoB) {
        double hipotenusa, cuadrado;

        hipotenusa = Math.sqrt((catetoA*catetoA)+(catetoB*catetoB));
        cuadrado = hipotenusa * hipotenusa; // Calcula el cuadrado de la hipotenusa.

        YaCalculada(cuadrado);
        return hipotenusa;
    }

    // Bloque disponible despues de calcular la hipotenusa.
    @SimpleEvent(description = "Muestra la hipotenusa al cuadrado.")
    public void YaCalculada(double sucuadrado){
        EventDispatcher.dispatchEvent(this, "YaCalculada", sucuadrado);
    }
}

```

- Copy file **Pitagoras.java** in:

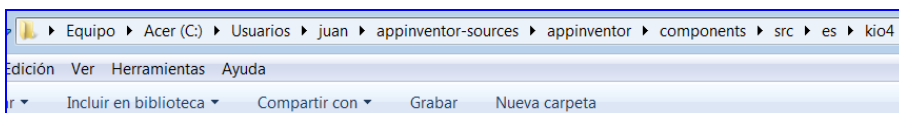
C:\Users\juan\appinventor-sources\appinventor\components\src\com\google\appinventor\components\runtime

Update.

[In new version Pitagoras.java is copied in:

C:\Users\juan\appinventor-sources\appinventor\components\src\es\kio4

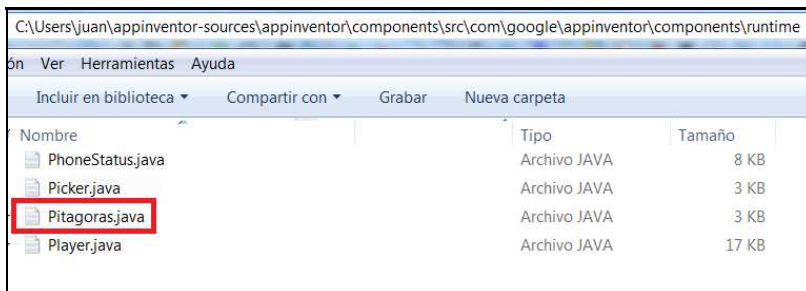
(Read: <http://ai2.appinventor.mit.edu/reference/other/extensions.html>) Look sección 3.3.1



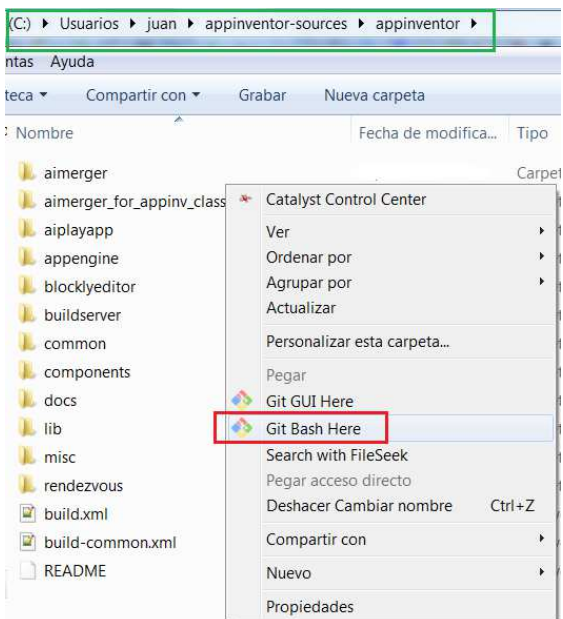
Usamos cookies propias y de terceros que entre otras cosas recogen datos sobre sus hábitos de navegación y realizan análisis de uso de nuestro sitio.

Si continúa navegando consideramos que acepta su uso. **Acepto** Más información

- 1.- In `C:\Users\juan\appinventor-sources\appinventor\components\src` create folder `es`:
`C:\Users\juan\appinventor-sources\appinventor\components\src\es`
- 2.- In `es`, create folder `kio4`:
`C:\Users\juan\appinventor-sources\appinventor\components\src\es\kio4`
- 3.- In `kio4` copy `Pitagoras.java` file.
- 4.- Look this line in `Pitagoras.java`
`package es.kio4.Pitagoras;`
- 5.- Add `import`, with this line:
`import com.google.appinventor.components.runtime.*;`
- 6.- También en la carpeta `kio4` creamos otra carpeta llamada `aiwebres` y en esa carpeta ponemos la imagen del icono que debe ser de 16x16
`iconName = "aiwebres/iconname.png"`

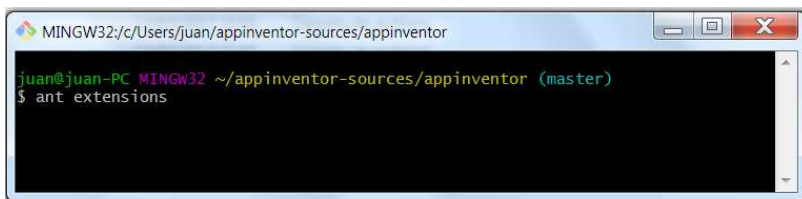


- Let's go to folder with App Inventor sources (`C:\Users\juan\appinventor-sources\appinventor`)
- Click with **Right button mouse**, then click **Git Bash Here**.



- **Git bash** get directory...

`C:\Users\juan\appinventor-sources\appinventor (master)`



[In Git bash, another way to get that directory is write: `cd c:/Users/juan/appinventor-sources/appinventor`]

- In **Git Bash** write:

`ant extensions`

C:\Users\juan\appinventor-sources\appinventor\components\build\extensions

```

MINGW32/c/Users/juan/appinventor-sources/appinventor
dexAllExtensions:
dexExtension:
[java] Picked up _JAVA_OPTIONS: -Xmx1024m
[echo] Dexing extension: com.Pitagoras

extensions:
[mkdir] Created dir: C:\Users\juan\appinventor-sources\appinventor\components\build\extensions

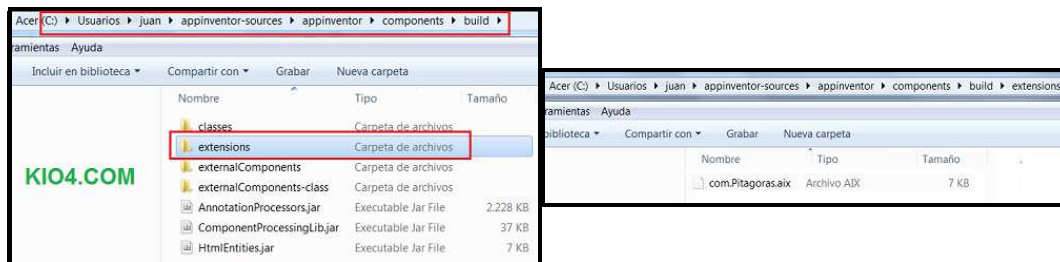
packExtension:
[zip] Building zip: C:\Users\juan\appinventor-sources\appinventor\components\build\extensions\com.Pitagoras.aix

BUILD SUCCESSFUL
Total time: 25 seconds
Picked up _JAVA_OPTIONS: -Xmx1024m

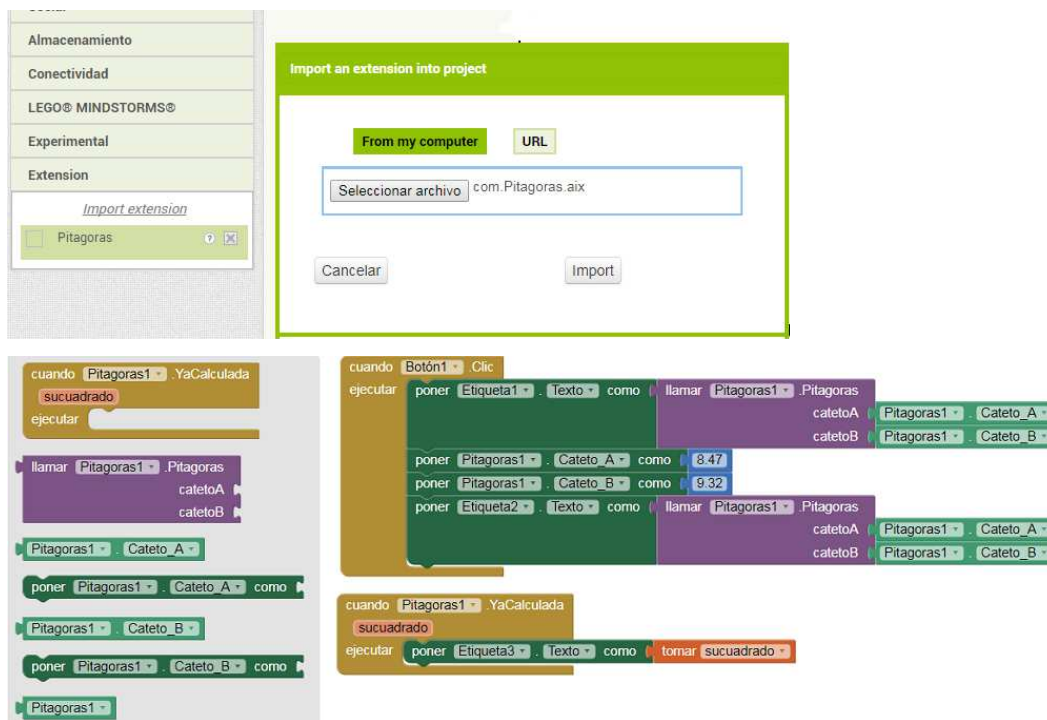
juan@juan-PC MINGW32 ~/appinventor-sources/appinventor (master)
$

```

- This is our extension: [com.Pitagoras.aix](#)



- Now in App Inventor. *Import extension* [com.Pitagoras.aix](#) and look as works.



- Important.

- When write your extension, change name package with your name, file name,... change this word in all file .java

```
package com.Pitagoras;
```

```
package com.Juan;
```

- This mean we build an **EXTENSION**, extensions aren't visible in Screen.

```
category = ComponentCategory.EXTENSION, nonVisible = true,
```

- This are DEFAULTS valors, you can look extension **Properties**. Letter f is float (also is possible: int, String, double,...)

```
public static final float DEFAULT_CATETO_A = 3f; public static final float DEFAULT_CATETO_B = 4f;
```




- It is **highly recommended to back up** the file Pitagoras.java in other folder.

- Pitagoras.aix **can not convert to** Pitagoras.java

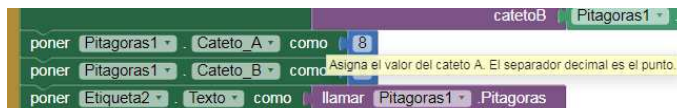
- You can put an icon image of 16x16 pixels:

iconName = "imagenes/extension.png")

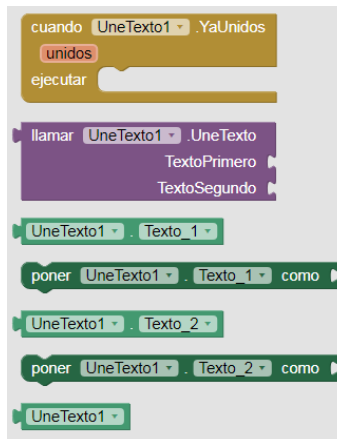
- Also external link:

iconName = "http://kio4.com/appinventor/imagenes/extension.png")

- Look at the comments:



- Another example similar to the previous one. Join 2 texts.



- This code is an adaptation to Pitagoras.java. This code join 2 texts.

- Example, in TextoPrimero set "Hola" and TextoSegundo send "Adios", result is "HolaAdios".

unidos = TextoPrimero + TextoSegundo;

- Fíjate que en la aplicación de Pitágoras las variables eran numéricas **double**, ahora son de cadena de texto **String**.

- Cuidado con las mayúsculas, no es igual String que string.

UnaTexto.java

```
package com.UnaTexto;
// @ Juan Antonio Villalpando
// kio4.com
// Creacion de extensiones. Junio 2017.
// Esta extension se utiliza para unir dos textos.

import com.google.appinventor.components.annotations.DesignerComponent;
import com.google.appinventor.components.annotations.DesignerProperty;
import com.google.appinventor.components.annotations.PropertyCategory;
import com.google.appinventor.components.annotations.SimpleEvent;
import com.google.appinventor.components.annotations.SimpleFunction;
import com.google.appinventor.components.annotations.SimpleObject;
```

Usamos cookies propias y de terceros que entre otras cosas recogen datos sobre sus hábitos de navegación y realizan análisis de uso de nuestro sitio.

Si continúa navegando consideramos que acepta su uso. **Acepto** Más información

```

import com.google.appinventor.components.runtime.*;

@DesignerComponent(version = UneTexto.VERSION,
    description = "Une dos textos. " + "Juan Antonio Villalpando - KIO4.COM ",
    category = ComponentCategory.EXTENSION,
    nonVisible = true,
    iconName = "")
@SimpleObject(external = true)
public class UneTexto extends AndroidNonvisibleComponent implements Component {

    public static final int VERSION = 1;
    public static final String DEFAULT_TEXTO_1 = "Hola";
    public static final String DEFAULT_TEXTO_2 = "Adios";
    private ComponentContainer container;
    private String texto_1 = "";
    private String texto_2 = "";

    public UneTexto(ComponentContainer container) {
        super(container.$form());
        this.container = container;
        Texto_1(DEFAULT_TEXTO_1);
        Texto_2(DEFAULT_TEXTO_2);
    }

    // Obtener el valor.
    @SimpleProperty(category = PropertyCategory.BEHAVIOR)
    public String Texto_1() {
        return texto_1;
    }

    @SimpleProperty(category = PropertyCategory.BEHAVIOR)
    public String Texto_2() {
        return texto_2;
    }

    // Establecer el valor.
    @DesignerProperty(editorType = PropertyTypeConstants.PROPERTY_TYPE_STRING, defaultValue = UneTexto.DEFAULT_TEXTO_1 + "")
    @SimpleProperty(description = "Asigna el valor del texto 1. ")
    public void Texto_1(String nuevoTexto_1) {
        this.texto_1 = nuevoTexto_1;
    }

    @DesignerProperty(editorType = PropertyTypeConstants.PROPERTY_TYPE_STRING, defaultValue = UneTexto.DEFAULT_TEXTO_2 + "")
    @SimpleProperty(description = "Asigna el valor del texto 2. ")
    public void Texto_2(String nuevoTexto_2) {
        this.texto_2 = nuevoTexto_2;
    }

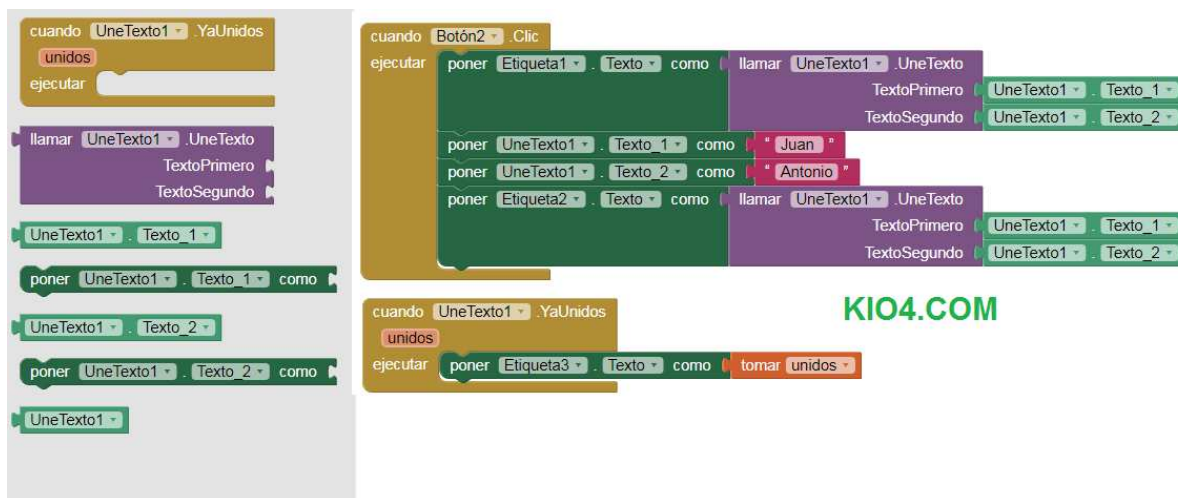
    // Funcion que une los dos textos.
    @SimpleFunction(description = "Une dos textos.")
    public String UneTexto(String TextoPrimero, String TextoSegundo) {
        String unidos;
        unidos = TextoPrimero + TextoSegundo;

        YaUnidos(unidos);
        return unidos;
    }

    // Bloque disponible despues de la union de los textos.
    @SimpleEvent(description = "Esto sale despues de unir los textos.")
    public void YaUnidos(String unidos){
        EventDispatcher.dispatchEvent(this, "YaUnidos", unidos);
    }
}

```

- Ejemplo.



- Every time you make a .java file, make a backup copy of it, copy it to another folder since at any moment you can block the clone of app-inventor-source and lose what you have.