



PRACTICA MULTIIDIOMA

DOCKER + APACHE



POR: ISABEL GONZÁLEZ ANZANO



Práctica Multiidioma

Creando el contenedor Docker e Instalaciones previas	1
1º Obtener la imagen del SO	1
2º Crear el contenedor uniendo volúmenes y exponiendo puertos	1
3º Instalaciones y actualizaciones necesarias	2
Actualizaciones	2
Instalamos apache y las librerías de php	2
Instalamos net-tools y editores de texto	2
Levantamos apache	2
Extra - guardamos una imagen del contenedor tal cual está	3
4º Instalamos la librería gettext	4
Modificar el fichero php.ini para que acepte la extensión gettext	4
Instalar la librería de php para usar gettext	4
Instalar los idiomas	4
5º Resumen de comandos más utilizados de Docker	5
Práctica Multiidioma	7
Descargar base de git	7
Estructura de ficheros	7
Crear los archivos .po automáticamente	7
Generar el fichero .mo a partir del .po	8
Instalar poedit o similares	9
Establecer un idioma por defecto en la aplicación	9
Establecer el valor con una variable de entorno	9
Activar los locales	10

Creando el contenedor Docker e Instalaciones previas

Partimos de la base de que se tiene instalado Docker en la máquina anfitriona, en mi caso Windows 10.

1º Obtener la imagen del SO

En este caso **ubuntu:latest**

Comando: `docker pull ubuntu:latest`

```
C:\Docker>docker pull ubuntu:latest
latest: Pulling from library/ubuntu
da7391352a9b: Pull complete
14428a6d4bcd: Pull complete
2c2d948710f2: Pull complete
Digest: sha256:c95a8e48bf88e9849f3e0f723d9f49fa12c5a00cfc6e60d2bc99d87555295e4c
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
```

2º Crear el contenedor uniendo volúmenes y exponiendo puertos

```
docker run -ti --name idiomas_d -v
"C:\Docker\public_html":/var/www/public_html -p 21080:80
ubuntu:latest
```

*Si public_html da problemas funciona tambien con html.

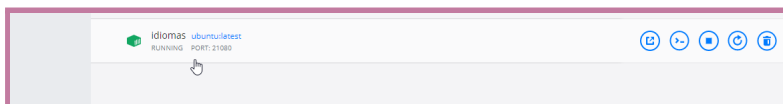
-ti: terminal interactiva

-v: volumen o carpeta compartida entre host y docker

-p: exponer el puerto del host 21080 al 80 del contenedor.

```
C:\Docker>docker run -ti --name idiomas -v "C:\Docker\public_html":/var/www/public_html -p 21080:80 ubuntu:latest
root@30f153c95147:/#
```

Vemos que ya está corriendo



y si vamos a la ruta /var/www/public_html:

```
C:\Docker>docker run -ti --name idiomas -v "C:\Docker\public_html":/var/www/public_html -p 21080:80 ubuntu:latest
root@30f153c95147:/# ls
bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@30f153c95147:/# cd /var/www/public_html
root@30f153c95147:/var/www/public_html# ls
'Creado en Host.txt'  Idiomas
root@30f153c95147:/var/www/public_html#
```

Podemos ver el archivo de prueba y las carpetas que están compartidas entre ambos volúmenes.

3º Instalaciones y actualizaciones necesarias

* Mejoramos los colores del bash:

```
alias ls='ls --color'
LS_COLORS='di=1:fi=0:ln=31:pi=5:so=5:bd=5:cd=5:or=31:mi=
0:ex=35:*.rpm=90'
export LS_COLORS
```

Actualizaciones

- apt-get update
- apt-get upgrade

Instalamos apache y las librerías de php

```
apt-get install apache2 php libapache2-mod-php
```

```
Please select the geographic area in which you live. Subsequent configuration questions will narrow this down by
presenting a list of cities, representing the time zones in which they are located.

 1. Africa    3. Antarctica  5. Arctic    7. Atlantic  9. Indian    11. SystemV  13. Etc
 2. America  4. Australia  6. Asia      8. Europe    10. Pacific  12. US
Geographic area: 8

Please select the city or region corresponding to your time zone.

 1. Amsterdam  10. Bucharest  19. Isle_of_Man  28. Luxembourg  37. Paris      46. Simferopol  55. Vaduz
 2. Andorra    11. Budapest   20. Istanbul    29. Madrid      38. Podgorica   47. Skopje      56. Vatican
 3. Astrakhan  12. Busingen  21. Jersey      30. Malta       39. Prague      48. Sofia       57. Vienna
 4. Athens     13. Chisinau   22. Kaliningrad 31. Mariehamn   40. Riga        49. Stockholm   58. Vilnius
 5. Belfast    14. Copenhagen 23. Kiev        32. Minsk       41. Rome        50. Tallinn     59. Volgograd
 6. Belgrade   15. Dublin     24. Kirov       33. Monaco      42. Samara      51. Tirane      60. Warsaw
 7. Berlin     16. Gibraltar 25. Lisbon      34. Moscow      43. San_Marino  52. Tiraspol    61. Zagreb
 8. Bratislava 17. Guernsey   26. Ljubljana   35. Nicosia     44. Sarajevo    53. Ulyanovsk   62. Zaporozhye
 9. Brussels   18. Helsinki   27. London      36. Oslo        45. Saratov     54. Uzhgorod    63. Zurich
Time zone: 29
```

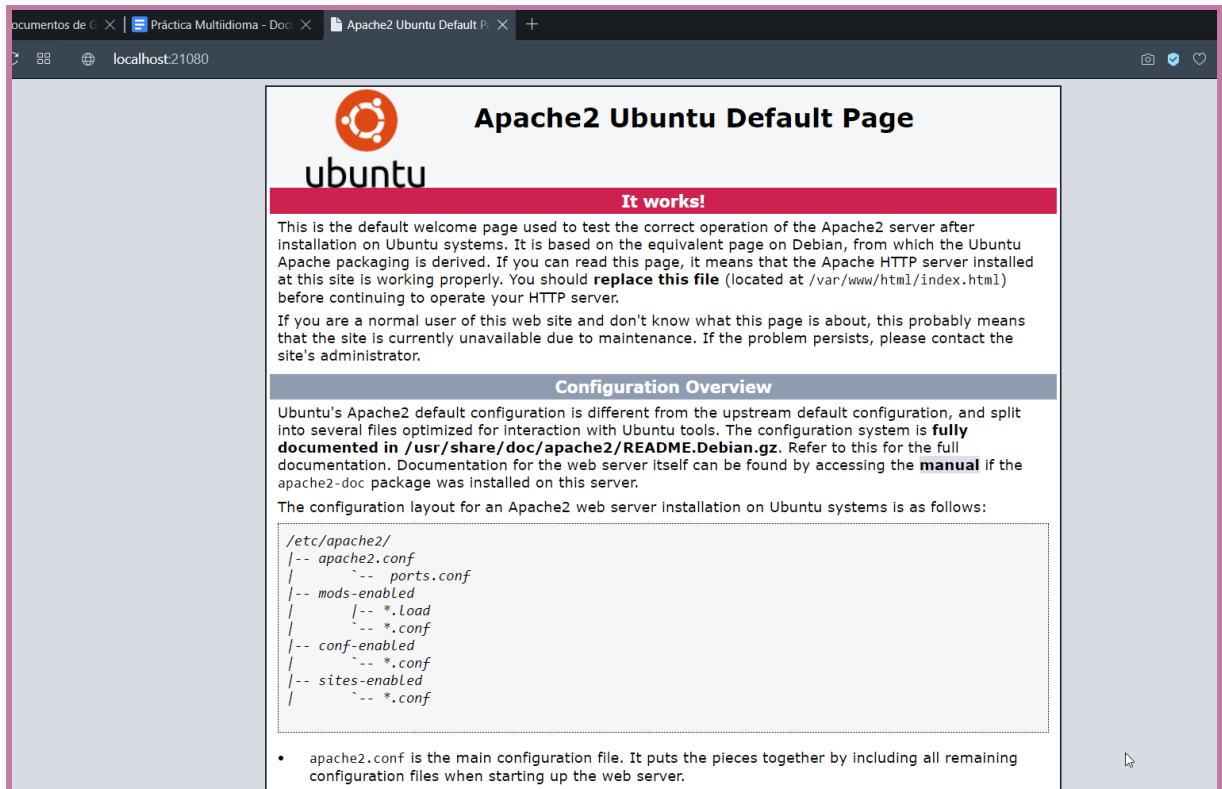
Instalamos net-tools y editores de texto

```
apt-get install net-tools vim nano
```

Levantamos apache

```
service apache2 start
```

Para comprobarlo abrimos una pestaña en el host, con localhost y el puerto que hemos elegido, y vemos como nos ha redirigido al puerto 80 del contenedor.



Extra - guardamos una imagen del contenedor tal cual está

```
docker commit -m "Imagen con apache y php en puerto 80" -a "Isabel Gonzalez" idiomas_d img_idioma
```

-a: usuario

```
C:\Docker\public_html>docker commit -m "Imagen con apache y php en puerto 80" -a "Isabel Gonzalez" idiomas_d img_idioma
sha256:cf0b4f8f3aab5d162ff0324e1efab086d70f5b89cf90dc6f010de7b474a940b
```

Y ahora nos aparece

		TAG	IMAGE ID	CREATED	SIZE
img_idioma		latest	cf0b4f8f3aab	less than a minute ago	294.42 MB
ubuntu	IN USE	latest	f643c72bc252	about 2 months ago	72.9 MB
jenkins/jenkins	IN USE	lts	ae4797bb5657	3 months ago	711.42 MB
ubuntu	IN USE	<none>	9140108b62dc	4 months ago	72.88 MB

o con docker images

```
C:\Docker\public_html>docker images
REPOSITORY          TAG          IMAGE ID          CREATED           SIZE
img_idioma          latest       cf0b4f8f3aab      About a minute ago 294MB
ubuntu              latest       f643c72bc252      7 weeks ago      72.9MB
jenkins/jenkins     lts         ae4797bb5657      2 months ago     711MB
ubuntu              <none>      9140108b62dc      3 months ago     72.9MB
```

4º Instalamos la librería gettext

```
apt-get install gettext
```

Modificar el fichero php.ini para que acepte la extensión gettext

ruta: /etc/php/7.4/apache2/php.ini

```
root@30f153c95147: /etc/php
GNU nano 4.8 /etc/php/7.4/apache2/php.ini

Notes for Windows environments :

- Many DLL files are located in the extensions/ (PHP 4) or ext/ (PHP 5+)
  extension folders as well as the separate PECL DLL download (PHP 5+).
  Be sure to appropriately set the extension_dir directive.

extension=bz2
extension=curl
extension=ffi
extension=ftp
extension=fileinfo
extension=gd2
extension=gettext
extension=gmp
extension=intl
extension=imap
extension=ldap
extension=mbstring
extension=exif      ; Must be after mbstring as it depends on it
extension=mysqli
extension=oci8_12c  ; Use with Oracle Database 12c Instant Client
extension=odbc
extension=openssl
extension=pdo_firebird
extension=pdo_mysql
```

Instalar la librería de php para usar gettext

```
sudo apt-get install php-gettext
```

```
service apache2 restart
```

Instalar los idiomas

```
sudo apt-get install locales
```

Comprobamos con locale -a

```
root@30f153c95147:/etc/php# locale -a
C
C.UTF-8
POSIX
root@30f153c95147:/etc/php#
```

Agregamos los idiomas que nos interesan con: `dpkg-reconfigure locales`

Para español, francés e inglés: 159 234 183

```
159. en_US.UTF-8 UTF-8      326. mg_MG.UTF-8 UTF-8      493. zh_HK.UTF-8 UTF-8
[More]
160. en_ZA ISO-8859-1      327. mhr_RU UTF-8          494. zh_SG GB2312
161. en_ZA.UTF-8 UTF-8    328. mi_NZ ISO-8859-13     495. zh_SG.GBK GBK
162. en_ZH UTF-8          329. mi_NZ.UTF-8 UTF-8    496. zh_SG.UTF-8 UTF-8
163. en_ZW ISO-8859-1      330. ml_IN UTF-8          497. zh_TW B1G5
164. en_ZW.UTF-8 UTF-8    331. mju_IN UTF-8        498. zh_TW.EUC-TW EUC-TW
165. eo UTF-8             332. mk_MK ISO-8859-5     499. zh_TW.UTF-8 UTF-8
166. eo_US.UTF-8 UTF-8    333. mk_MK.UTF-8 UTF-8   500. zu_ZA ISO-8859-1
167. es_AR ISO-8859-1      334. ml_IN UTF-8        501. zu_ZA.UTF-8 UTF-8

(Enter the items you want to select, separated by spaces.)
Locales to be generated: 159 234 183_
```

Establecemos por defecto el español: 4

```
(Enter the items you want to select, separated by spaces.)
Locales to be generated: 159 234 183

Many packages in Debian use locales to display text in the correct language for the user. You can choose a default locale for the system from the generated locales.

This will select the default language for the entire system. If this system is a multi-user system where not all users are able to speak the default language, they will experience difficulties.

1. None 2. C.UTF-8 3. en_US.UTF-8 4. es_ES.UTF-8 5. fr_FR.UTF-8
Default locale for the system environment: 4
```

Hacemos un `locale -a` y vemos que ya aparecen los idiomas instalados:

```
root@30f153c95147:/etc/php# locale -a
C
C.UTF-8
POSIX
en_US.utf8
es_ES.utf8
fr_FR.utf8
root@30f153c95147:/etc/php#
```

5º Resumen de comandos más utilizados de Docker

`docker pull nombre_imagen` (descargar imagen)

`docker search ubuntu` (buscar algo concreto)

`docker images` (listar imagenes)

`docker ps -a` (ver contenedores)

Crear contenedor:

`docker run --name nombre -ti -p 80:80 -v ruta:ruta nombre_imagen:tar`

Crear una imagen a partir de un contenedor:

`docker commit -m "mensaje" nombre_contenedor imagen:tag`

Exportar:

`docker export nombre_contenedor > file.tar`

importar: (imagen)

`docker import file.tar imagen:tag`

Encender un contenedor:

`docker start nombreContenedor`

Ejecutar comandos:

`docker exec nombreContenedor`

(`docker exec -ti idiomas_d bash`)

Eliminar un contenedor:

`docker rm nombreContenedor`

Eliminar imagenes: (si hay contenedores con la imagen, primero se borran los contenedores)

```
docker rmi nombreImagen
```

Exportar contenedores:

```
docker export -o nombreContenedor.tar nombreContenedor
```


Práctica Multiidioma

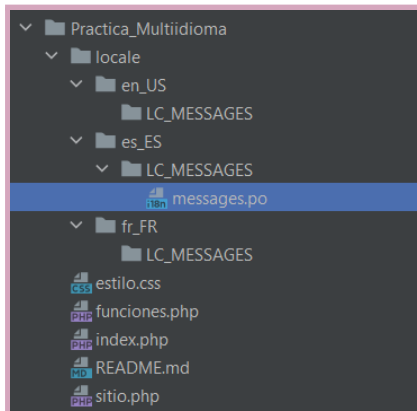
Descargar base de git

Siguiendo indicaciones de la práctica:

```
git clone https://github.com/MAlejandroR/Practica_Multiidioma/
```

Estructura de ficheros

Creamos la carpeta locale que contendrá los archivos de traducciones de cada idioma



A partir de ahora si escribimos algún texto que deba ser traducido simplemente hay que utilizar la siguiente sintaxis:

```
<h1><?php echo gettext("Bienvenido a este sitio web") ?></h1>
```

o la forma abreviada:

```
<title><?php echo _("WELLCOME") ?></title>
```

Crear los archivos .po automáticamente

Debemos crear el archivo messages.po en cada carpeta de idioma.

Nos ubicamos en el directorio del idioma:

```
root@30f153c95147:/var/www/public_html/PracticaMultiidioma/Practica_Multiidioma/locale/es_ES/LC_MESSAGES#
root@30f153c95147:/var/www/public_html/PracticaMultiidioma/Practica_Multiidioma/locale/es_ES/LC_MESSAGES# xgettext --language=PHP --from-code=UTF-8 -o messages.po -j messages.po /var/www/public_html/PracticaMultiidioma/Practica_Multiidioma/index.php
root@30f153c95147:/var/www/public_html/PracticaMultiidioma/Practica_Multiidioma/locale/es_ES/LC_MESSAGES#
```

```
xgettext --language=PHP --from-code=UTF-8 -o messages.po -j
messages.po /var/www/html/Practica_Multiidioma/index.php
/var/www/html/Practica_Multiidioma/sitio.php
```

```

1  # SOME DESCRIPTIVE TITLE.
2  # Copyright (C) YEAR THE PACKAGE'S COPYRIGHT HOLDER
3  # This file is distributed under the same license as the PACKAGE package.
4  # FIRST AUTHOR <EMAIL@ADDRESS>, YEAR.
5  #
6  #, fuzzy
7  msgid ""
8  msgstr ""
9  "Project-Id-Version: PACKAGE VERSION\n"
10 "Report-Msgid-Bugs-To: \n"
11 "POT-Creation-Date: 2021-01-18 12:40+0100\n"
12 "PO-Revision-Date: YEAR-MO-DA HO:MI+ZONE\n"
13 "Last-Translator: FULL NAME <EMAIL@ADDRESS>\n"
14 "Language-Team: LANGUAGE <LL@li.org>\n"
15 "Language: \n"
16 "MIME-Version: 1.0\n"
17 "Content-Type: text/plain; charset=CHARSET\n"
18 "Content-Transfer-Encoding: 8bit\n"
19
20 #: /var/www/public_html/PracticaMultiidioma/Practica_Multiidioma/index.php:75
21 msgid "FRANCES"
22 msgstr ""
23
24 #: /var/www/public_html/PracticaMultiidioma/Practica_Multiidioma/index.php:76
25 msgid "INGLES"
26 msgstr ""
27
28 #: /var/www/public_html/PracticaMultiidioma/Practica_Multiidioma/index.php:77
29 msgid "CASTELLANO"
30 msgstr ""

```

Esto habrá que repetirlo en cada archivo de idiomas, es decir, en francés e inglés también habrá que ejecutar este comando.



Importante:

Puede ser que nos salga un warning:

*messages.po: warning: Charset "CHARSET" is not a portable encoding name.
Message conversion to user's charset might not work.*

Esto se arregla editando el fichero creado dónde aparece charset y cambiarlo a UTF-8:

```

"Project-Id-Version: PACKAGE VERSION\n"
"Report-Msgid-Bugs-To: \n"
"POT-Creation-Date: 2021-01-18 12:45+0100\n"
"PO-Revision-Date: YEAR-MO-DA HO:MI+ZONE\n"
"Last-Translator: FULL NAME <EMAIL@ADDRESS>\n"
"Language-Team: LANGUAGE <LL@li.org>\n"
"Language: \n"
"MIME-Version: 1.0\n"
"Content-Type: text/plain; charset=UTF-8\n"
"Content-Transfer-Encoding: 8bit\n"

```

Con esto hecho, simplemente queda completar las traducciones en cada fichero messages.po para cada idioma.

Generar el fichero .mo a partir del .po

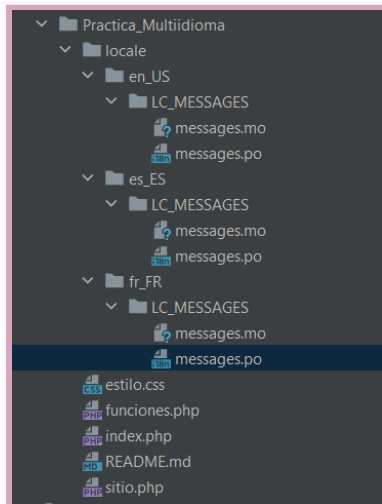
```
msgfmt -o messages.mo messages.po
```

```

root@30f153c95147:/var/www/public_html/PracticaMultiidioma/Practica_Multiidioma/locale/fr_FR/LC_MESSAGES# ls
messages.mo  messages.po
root@30f153c95147:/var/www/public_html/PracticaMultiidioma/Practica_Multiidioma/locale/fr_FR/LC_MESSAGES#

```

De esta forma ya tenemos todos los archivos con los datos que necesitamos:



Instalar poedit o similares

```
apt-get install poedit
```

Nos pedirá establecer el idioma del teclado 83



Establecer un idioma por defecto en la aplicación

Establecer el valor con una variable de entorno

```
putenv("LC_ALL=es_ES.utf8");
```

En mi caso todo esto lo voy a gestionar con funciones, en el archivo funciones.php

```
// ** Funciones para locales- Idiomas
function asigna_idioma($idioma){

    if (!putenv( setting: "LC_ALL=$idioma"))
        $msj="No se ha podido establecer el idioma $idioma en variables de entorno<br />";
    if (setlocale( category: LC_ALL, $idioma)=="")
        $msj.="No se ha podido cargar el idioma $idioma como categoría LC_ALL";
    echo setlocale ( category: LC_ALL, locale: 0);
    $dominio = "messages";
    echo bindtextdomain($dominio, directory: "locales");
}
```

Activar los locales

```
locale-gen fr_FR fr_FR.utf8
```

```
locale-gen en_US en_US.utf8
```

```
locale-gen es_ES es_ES.utf8
```

```
root@30f153c95147:/var/www/public_html/PracticaMultiidioma/Practica_Multiidioma# locale -a
C
C.UTF-8
POSIX
en_US
en_US.iso88591
en_US.utf8
es_ES
es_ES.iso88591
es_ES.utf8
fr_FR
fr_FR.iso88591
fr_FR.utf8
french
spanish
```