Mac Development System With Homebrew

How to install Apache, PHP (with xdebug) and MySQL on a Mac using Homebrew, a software package manager for MacOS X. The list of available software packages can be found at braumeister.org, although there are more "taps" available at other repositories.

The Mac already comes with Apache and PHP installed, but if you want to be able to upgrade to more recent versions you can use Homebrew to install these. Also, starting with the OS Sierra, the built-in Apache is more difficult to use due to some missing scripts. MacOS X doesn't include MySQL.

Before Installing Homebrew

1. Remove MacPorts or Fink, if these are installed
   - Macports
     ```
     sudo port -f uninstall installed
     sudo rm -rf /opt/local
     sudo rm -rf /Applications/DarwinPorts
     ```
   - Fink (untested)
     ```
     fink remove --recursive daemonic xinitrc
     sudo rm -rf /sw
     ```

2. Make sure Xcode and the Command Line Tools of Xcode are installed
   - Depending on your version, command line tools can be installed in two ways:
     1. In Xcode, go to Preferences > Downloads > Command Line Tools and click install
     2. In Terminal, type `xcode-select --install` and then click install on the pop-up

Install Homebrew

1. Type the following in Terminal to install Homebrew:
   ```
   /usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
   ```

2. Run the command to check your installation:
   ```
   brew doctor
   ```
   - There may be several errors but Homebrew has pretty good explanations of how to fix them

3. Make sure brew is up to date (if brew was already installed):
   ```
   brew update
   ```
Install Apache

Enter the following in Terminal:

1. Turn off built-in Apache

```
sudo apachectl stop
sudo launchctl unload -w /System/Library/LaunchDaemons/org.apache.httpd.plist
```

2. Install Homebrew Apache

```
brew install httpd
```

3. Set up Apache to autostart upon reboot

```
brew services start httpd
```

4. Test that Apache is working by pointing your browser to [http://localhost:8080](http://localhost:8080). You should see a simple header that says "It works!".

5. Troubleshooting tips if the browser can't connect

```
#check if server is up
ps -aef | grep httpd
#restart Apache
brew services restart httpd
# watch Apache log
tail -f /usr/local/var/log/httpd/error_log
```

Configure Apache

Make the following changes in `/usr/local/etc/httpd/httpd.conf`:

1. Change the listen port from the brew default of `Listen 8080` to `Listen 80`

```
# if using the global root
DocumentRoot "/Library/WebServer/Documents"
# if using the user home directory
DocumentRoot "/Users/username/Sites"
```
3. Set the Directory tag to the same root

```bash
# if using the global root
<Directory "/Library/WebServer/Documents"
# if using the user home directory
<Directory "/Users/username/Sites"
```

4. Set AllowOverride within the same Directory tag to All

```bash
AllowOverride All
```

5. Enable rewrite module (disabled by default)
   - Uncomment the line that says `LoadModule rewrite_module libexec/mod_rewrite.so`

6. Change User and Group
   - By default these are set to `daemon`. Change to `[www]` or actual user and group names to avoid permission problems.

7. Set the server name: `ServerName localhost`

Install PHP

1. In Terminal, enter the following commands:

   ```bash
   # install php (dependencies will also be installed)
   # this will install latest version
   # use @ to install a specific version, like php@7.2
   brew install php
   # if using imagick php extension, first install imagemagick itself using homebrew, then install the php extension using pecl
   brew install imagemagick
   pecl uninstall -r imagick && pecl install imagick
   # see below regarding installing xdebug
   ```

   *Note that "php@7.2" is just an example for PHP version 7.2. Change according to the version you intend to install.*

2. Make changes to Apache configuration at `/usr/local/etc/httpd/httpd.conf` to use newly installed PHP version

   ```bash
   # Add php load module command
   LoadModule php7_module /usr/local/opt/php@7.2/lib/httpd/modules/libphp7.so
   # Set directory indexes to PHP by changing the DirectoryIndex options
   <IfModule dir_module>
     DirectoryIndex index.php index.html
   </IfModule>
   ```
Install Xdebug PHP Extension

This section assumes you've installed PHP in the previous step.

1. In Terminal, enter the following command:
   ```shell
type
```
   `pecl uninstall -r xdebug && pecl install xdebug`
   ```shell
```

2. In the php.ini file, add the following lines:
   (look at phpinfo to see where the file is located)
   ```ini
   xdebug.remote_enable=1
   xdebug.remote_port=9000
   xdebug.remote_handler="dbgp"
   xdebug.remote_host="localhost"
   ```

3. Restart Apache by entering the following command in Terminal (you may be prompted for a password):
   ```shell
   brew services restart httpd
   ```

Install MySQL

1. In Terminal, enter the following command:
   ```shell
   brew install mysql
   ```

2. Follow Homebrew instructions for launching
   - Enter `brew services restart mysql` following the other Homebrew instructions (but before setting running `mysql_secure_installation`):
   - If you have problems restarting the mysql server, try restarting your computer.

3. Set a password to avoid errors. Enter the following and follow the prompts:
   ```shell
   mysql_secure_installation
   ```

Check versions

1. In Terminal, enter one of the following commands (lines beginning with # are just descriptions):
   ```shell
   brew install mysql
   ```
# list versions of all programs installed by home-brew
brew list --versions

or

# show the version of a single program
brew list --versions mysql

## Update programs

1. In Terminal, enter the following command choices (the purpose for each command follows the # symbol - you only need to type the part before the #):

   ```
   brew update  # first update homebrew itself
   brew outdated # list what is outdated
   brew upgrade # upgrade everything
   brew upgrade mysql # or upgrade a specific program
   ```

## Remove old versions of programs

1. By default Homebrew leaves the old version intact when updating program. To remove them, type the following in Terminal:

   ```
   # to see what would be removed (without removing anything)
   brew cleanup -n

   # to remove all old versions of all programs
   brew cleanup

   # to remove the old versions for just one program
   brew cleanup mysql
   ```

## Uninstall programs

1. For php, each new version branch (moving from 7.0 to 7.1, for example) is a separate program in homebrew. Unlinking the program as shown above still keeps it on your system so you need to perform an uninstall to remove it

   ```
   brew uninstall php@7.0
   ```