

IoTとデータを分析・可視化する ハンズオン、Q&A会

2016年8月27日

先端IT活用推進コンソーシアム
クラウド・テクノロジー活用部会

今期（2015/04～2016/08）の活動概要

- 4月16日（土）データ分析を知る：統計基礎＋R入門
 - <http://www.slideshare.net/natsumiirimura/ss-61013742>
 - <http://www.slideshare.net/natsuwatanabel/r-61005514>
 - <http://www.slideshare.net/natsuwatanabel/r-61005486>
- 5月21日（土）今さら聞けないWebサイト開発
 - <http://www.slideshare.net/ssuser5bbd5a/aitc-web-vol1>
- 6月18日（土）Webで可視化＋可視化ライブラリさわってみよう[1] D3.js
 - http://aramoto.sakura.ne.jp/20160618_D3js/20160618_IT%8f%97%8eq%89%ef_D3js.pdf
- 7月16日（土）Webで可視化＋可視化ライブラリさわってみよう[2] NVD3.js
 - http://aramoto.sakura.ne.jp/20160716_nv3/20160716_IT%8f%97%8eq%89%ef_nv3.pdf
- 8月27日（土）IoTのデータを分析・可視化するハンズオン、Q&A会

これで全部かと言うと

- ハンズオンでやったのは、最近流行ったものだけ
 - Rによる統計
 - HTML5/JavaScript/CSS
 - データの可視化 × 2
- システム全体を作るためには、古い技術も必要
 - Webサーバ側で何か処理をする
 - サーバ側で実行する言語（PHP,Perl,Java,Ruby on Rails, e.t.c....）
 - データの保存と取り出し
 - HTTP通信,RDB, e.t.c...

ということで、今日の予定


- 14:00～16:00：データ可視化（NVD3.js）の復習
 - ざっくりとリマインド
 - もう1度、動かしてみる
 - Q&A、リクエストなどあれば
- 16:00～18:00：システムをPC上に構築してみる
 - XAMPPをインストール
 - <https://ja.wikipedia.org/wiki/XAMPP>
 - Windows/Mac/Linuxに対応
 - 簡単に導入&破棄できる
 - レンタルサーバを借りると、ほぼ同じ環境が手に入る
 - 既存の可視化システムを載せてみる
 - <http://aramoto.sakura.ne.jp/loT/>

前半

データ可視化(NVD3js) の復習 + Q&A会

前回の資料

http://aramoto.sakura.ne.jp/20160716_nvd3/20160716_IT%8f%97%8eq%89%ef_nvd3.pdf



後半

**XAMPP
を動かしてみる**

XAMPPとは

- <https://ja.wikipedia.org/wiki/XAMPP>



The screenshot shows the Japanese Wikipedia page for XAMPP. The browser window title is "W XAMPP - Wikipedia". The address bar shows the URL "https://ja.wikipedia.org/wiki/XAMPP". The page content includes the Wikipedia logo, navigation tabs (ページ, ノート, 閲覧, 編集, 履歴表示), and a search box. The main article text describes XAMPP as a package of free software for running web applications, developed by Apache Friends. A table on the right provides details about XAMPP, including its developer (Apache Friends), latest version (7.0.8), supported OS (Linux, Windows, Mac OS X, Solaris), type (WAMP), license (GPL), and official website (www.apachefriends.org).

W XAMPP - Wikipedia

https://ja.wikipedia.org/wiki/XAMPP

ログインしていません トーク 投稿記録 アカウント作成 ログイン

ページ ノート 閲覧 編集 履歴表示 検索

XAMPP

XAMPP (ザンプ)とは、ウェブアプリケーションの実行に必要なフリーソフトウェアをパッケージとしてまとめたもので、apachefriends.orgから提供されている。主として開発用あるいは学習用ではあるが、イントラネットなどにおいて実運用環境として使われることもある。

開発元	Apache Friends
最新版	7.0.8 / 2016年6月28日 (44日前)
対応OS	クロスプラットフォーム (Linux / Windows / Mac OS X / Solaris)
種別	WAMP
ライセンス	GPL
公式サイト	www.apachefriends.org

テンプレートを表示

概要 [編集]

本番運用で使うには

- 社内：

- XAMPPは入門用の環境なので、セキュリティが甘い
 - 本番運用には、セキュリティ設定を強化するなどの対策が必要

- 社外

- クラウドで、同様の環境が簡単に手に入る
 - 例：さくらインターネットのレンタルサーバ：515円／月

<http://www.sakura.ne.jp/standard.html>

↑サーバを1カ月動かした時の電気代より安いかも？

The image shows a screenshot of the Sakura Internet server configuration interface. The interface is organized into several categories on the left: 'サイト・DB・プログラム' (Site, DB, Program), 'ドメイン' (Domain), 'セキュリティ' (Security), 'メール' (Email), and 'サポート・運用' (Support & Operation). Under 'サイト・DB・プログラム', there are options for CMS (WordPress, EC-CUBE, concrete5, XOOPS Cube), PHP (Ver. 4/5 selectable), MySQL (Ver 5.5 upgradeable), 独自CGI (Perl, Ruby, Python), クイックインストール (独自CGI), and さくらのブログ. The 'PHP' and 'MySQL' options are highlighted with a red box. Other options include 独自ドメイン, サブドメイン, マルチドメイン (20個), 共有SSL (データ暗号化), WAF (脆弱性対策), SFTP, 独自SSL (SNI SSL), 国外IPアドレスフィルタ, Web改ざん検知サービス, ウェブメール, メールアドレス無制限, メーリングリスト (10個), 迷惑メールフィルタ, 電話サポート無料, サーバコントロールパネル, アクセス解析, シェルログイン, 複数人管理, WebDAV (ファイル共有), and モニタリングツール.

開発環境の準備 - 1

- XAMPPをダウンロード
 - <https://www.apachefriends.org/jp/index.html>
 - ダウンロードが遅い場合は、USBメモリで回覧します



The image shows a browser window displaying the XAMPP website. The browser's address bar shows the URL <https://www.apachefriends.org/jp/index.html>. The page title is "XAMPP Apache + MariaDB + PHP + Perl". Below the title, there is a section titled "XAMPP とは？" (What is XAMPP?). The text describes XAMPP as a popular PHP development environment that is free and includes Apache, MariaDB, PHP, and Perl. A large video player with the XAMPP logo and a play button is centered on the page. At the bottom, there are three download buttons: "ダウンロード" (Download) with a green arrow, "Windows 向け XAMPP v5.6.23 (PHP 5.6.23)", "Linux 向け XAMPP v5.6.23 (PHP 5.6.23)", and "OS X 向け XAMPP v5.6.23 (PHP 5.6.23)".

開発環境の準備 - 2

- テキストエディタ
 - いつも使っているテキストエディタでOK
- 注意：Macのテキストエディットは要設定変更
 - Mavericks以降、ダブルクォートが自動的に変換される
 - この機能を無効にしておくこと



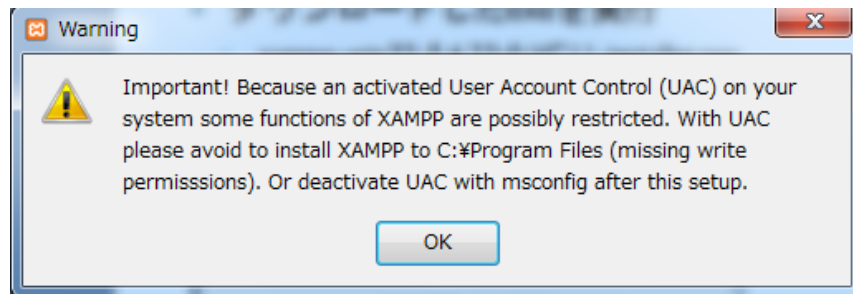
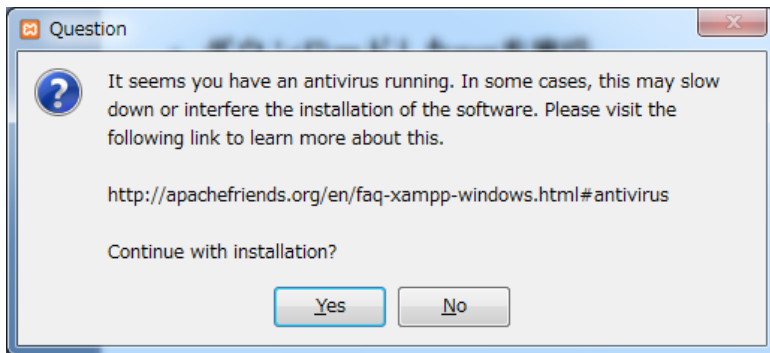
開発環境の準備 - 3

- ブラウザ

- [F-12]を押せば、開発ツールが起動する
- **Firefox ← 今日はこちらを使用してください**
 - オススメアドオン：Firebug
 - ・ 「ツール」 → 「アドオン」 → 「Firebug」で検索
 - ・ 「ツール」 → 「Web開発」 → 「Firebug」 → 「Firebugを開く」
 - **Safari**
 - ・ 「環境設定」 → 「詳細」 → 「メニューバーに”開発”メニューを表示」
 - ・ ローカルファイルにアクセスできない場合
 - ・ メニュー「開発」 → 「ローカルファイルの制限を無効にする」をチェック
 - **IE (9以降)**
 - ・ ローカルだとうまく動作しない
 - ・ データにアクセスしている部分をjQueryにすれば、動くらしい
 - ・ PCにApacheを導入した方が早い
 - **Chrome**： 「ツール」 → 「デベロッパーツール」
 - ・ ローカルファイルを参照するため、起動オプションを追加
「--allow-file-access-from-files」

インストール：Windows版 - 1

- ダウンロードしたexeを実行
 - xampp-win32-5.6.23-0-VC11-installer.exe



It seems you have an antivirus running. In some cases, this may slow down or interface the installation of the software. please visit the following link to lean more about this.

Google翻訳結果

あなたがウイルス対策を実行しているようです。いくつかのケースでは、これは減速またはソフトウェアのインストールをインターフェースすることができます。これについての詳細を傾くするには次のリンクをご覧ください。

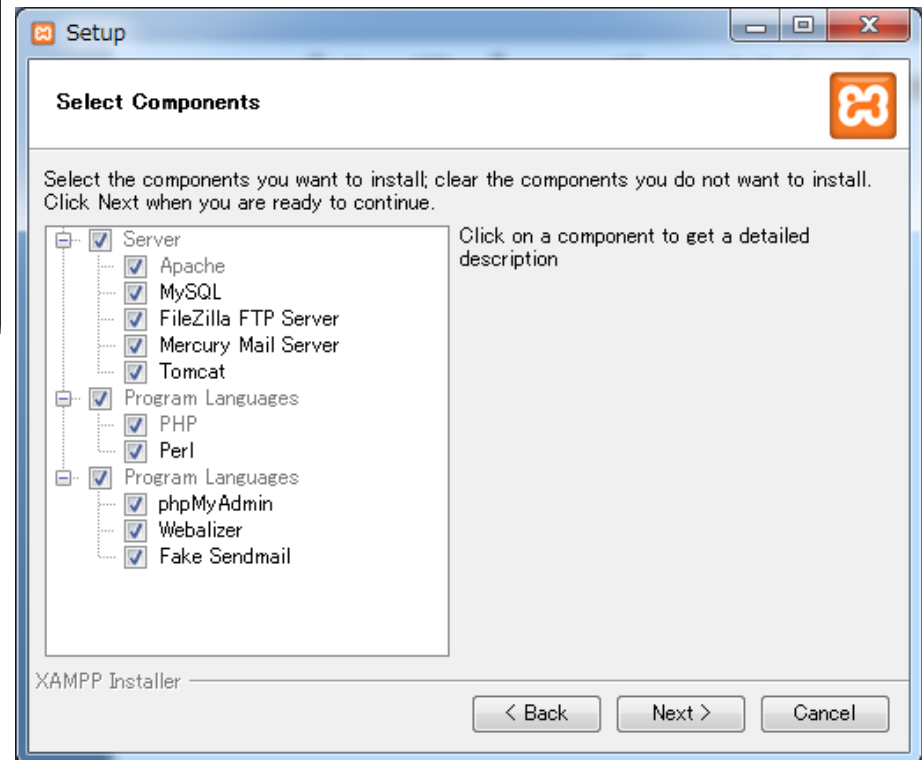
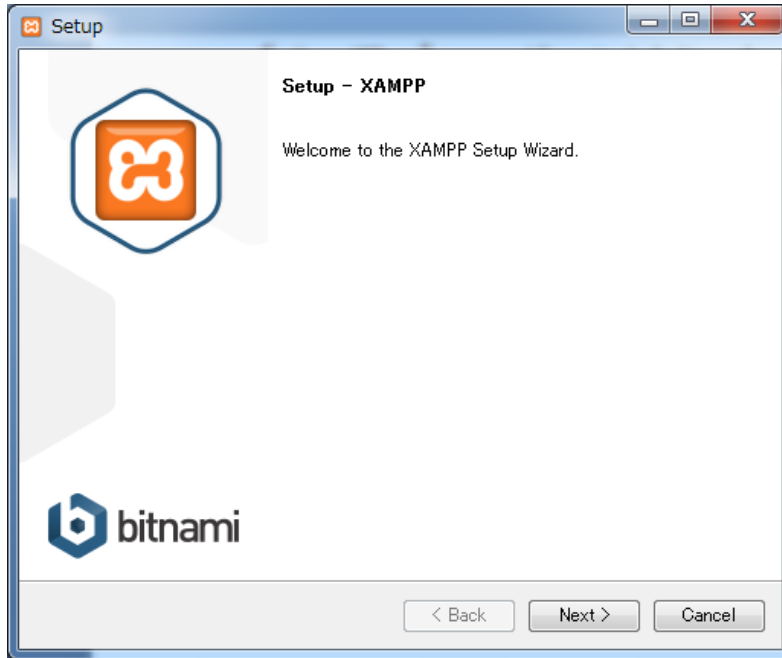
Important! because activated User Account Control(UAC) on your system some functions of XAMPP are possibly restricted. With UAC please avoid to install XAMPP to C:\Program Files(missing write permissions). Or deactivate UAC with msconfig after this setup.

Google翻訳結果

重要！ので、あなたのシステム上の活性化、ユーザーアカウント制御（UAC）は、XAMPPの一部の機能は、おそらく制限されています。UACでXAMPPをインストールするには避けてください C:\Program Files（書き込み権限が不足して）。またはこのセットアップ後のMSCONFIGでUACを無効にします。

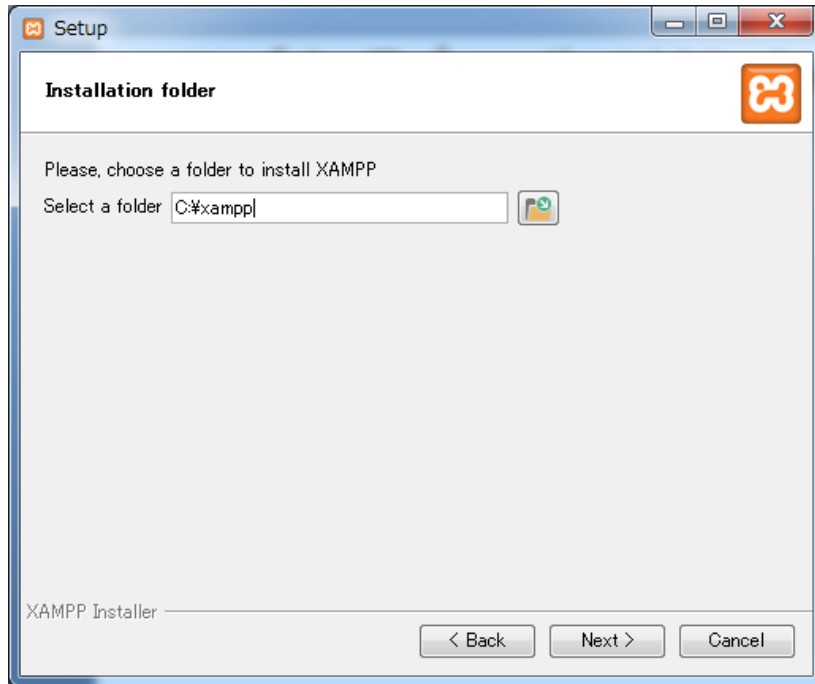
インストーラ：Windows版－2

- インストールするコンポーネントを選択



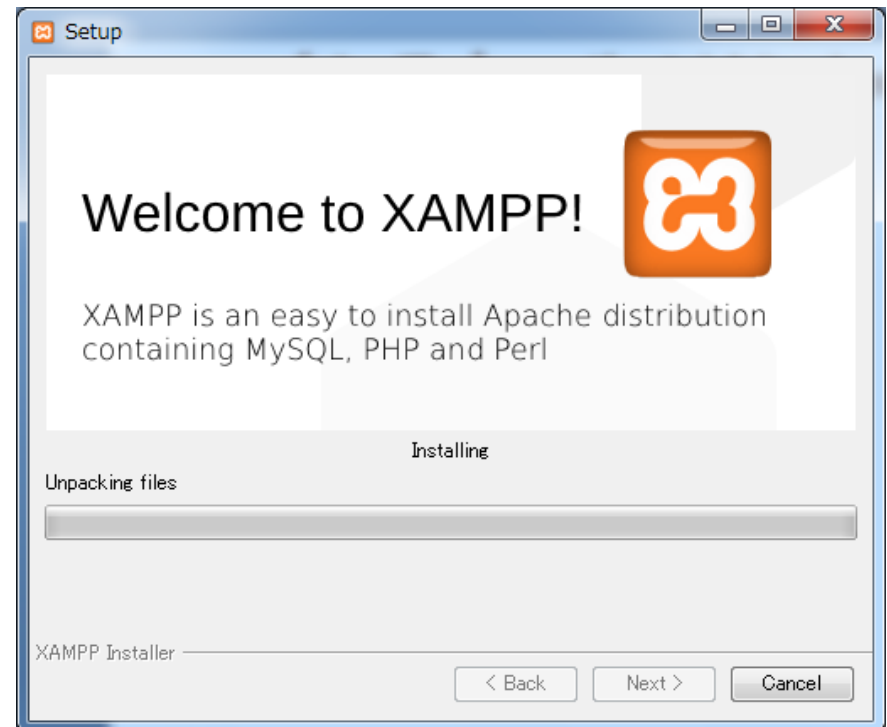
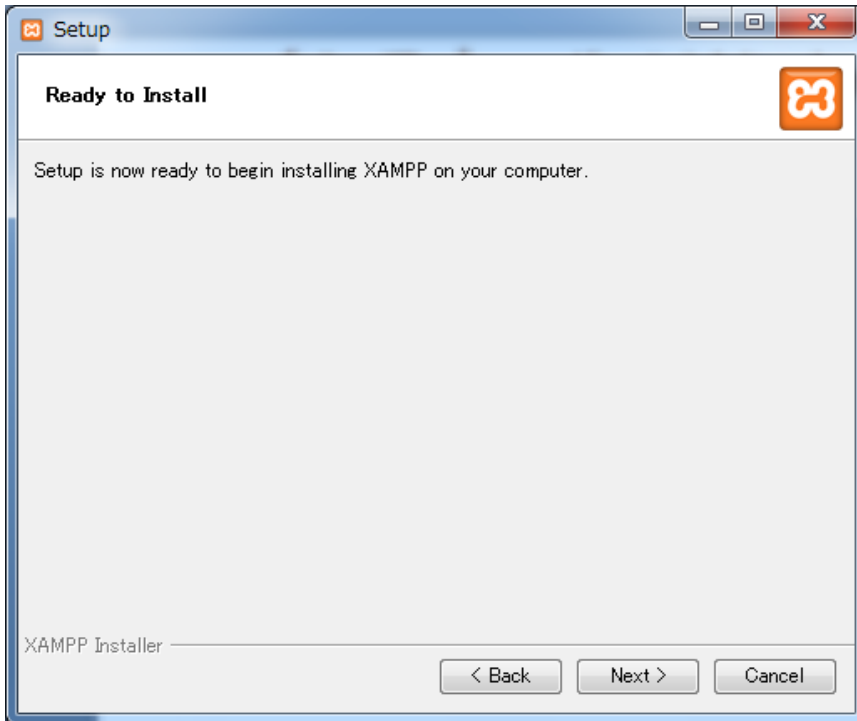
インストール : Windows版 - 3

- インストール先のディレクトリを選択



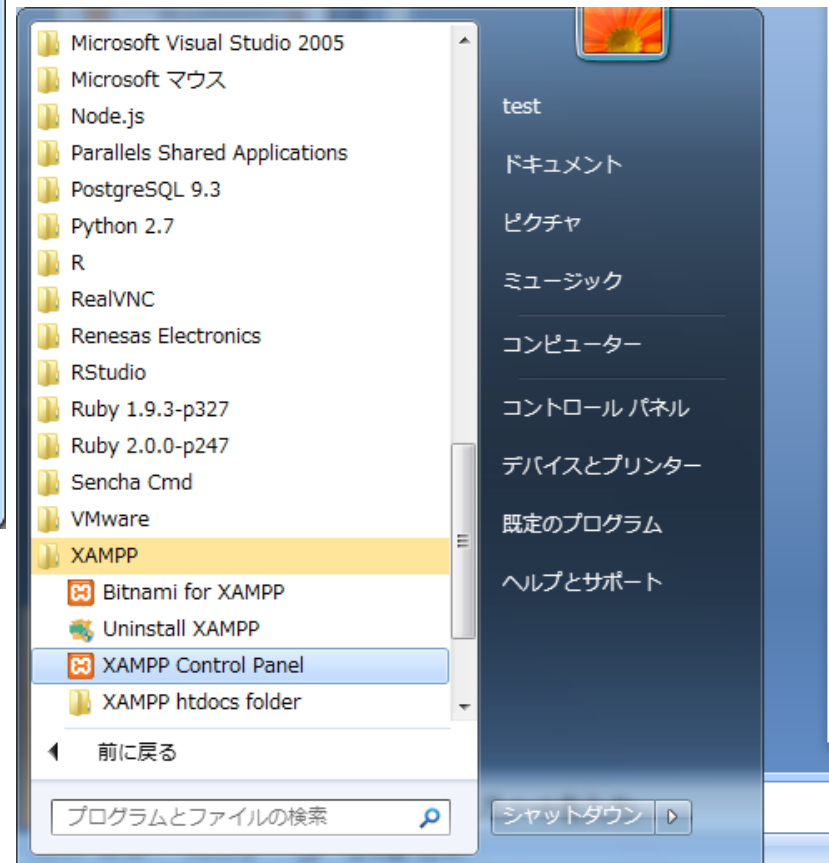
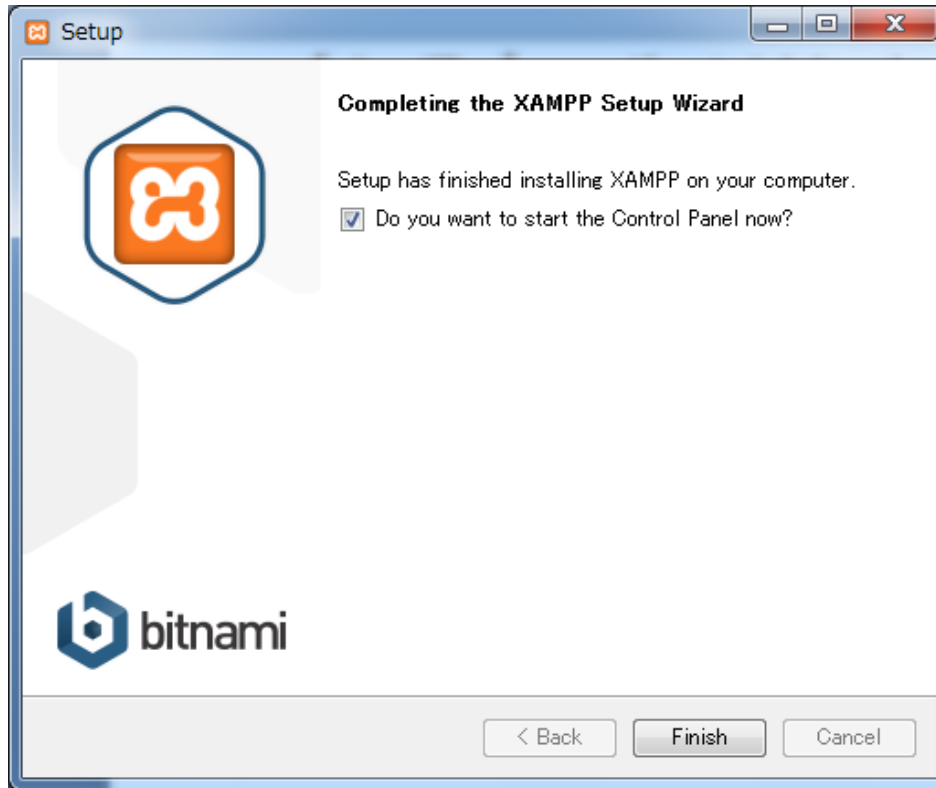
インストーラ：Windows版－4

- インストールを実行



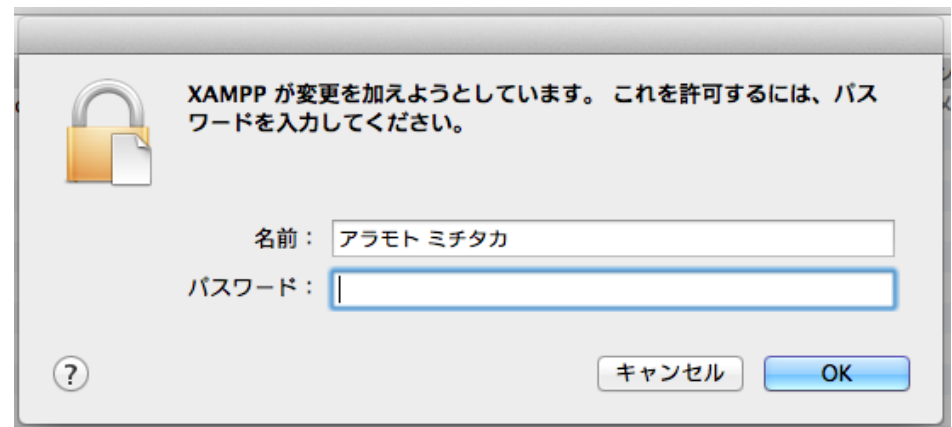
インストール：Windows版 - 5

- インストール終了後に、コントロールパネルを起動



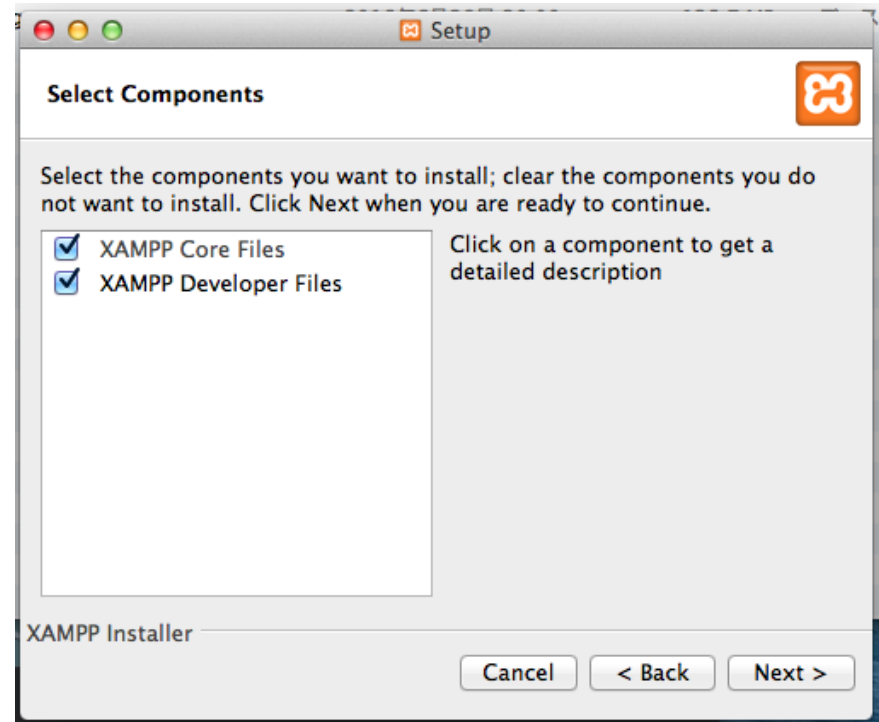
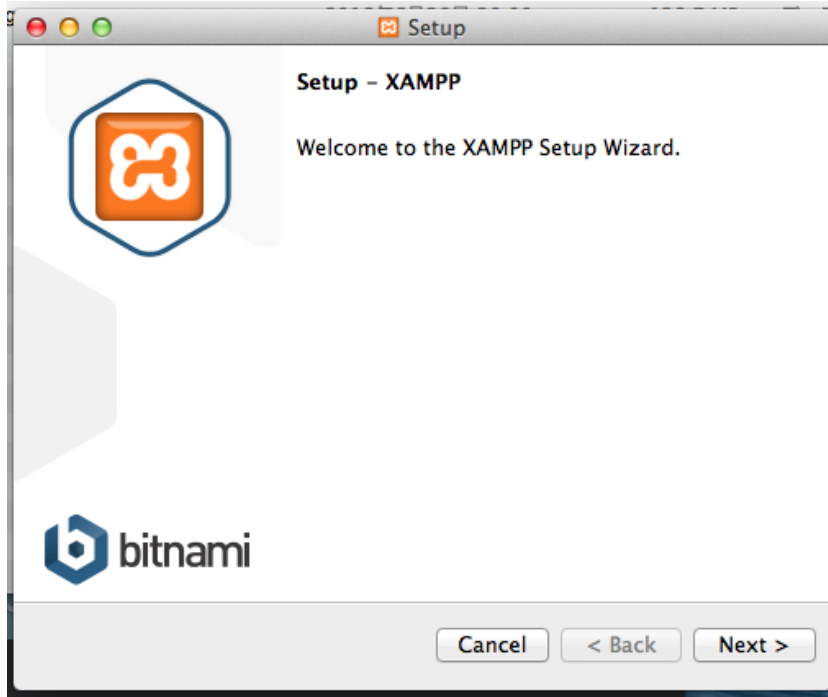
インストール：Mac版ー 1

- ダウンロードしたdmgを開き、インストーラを実行
 - xampp-osx-5.6.23-0-installer.dmg



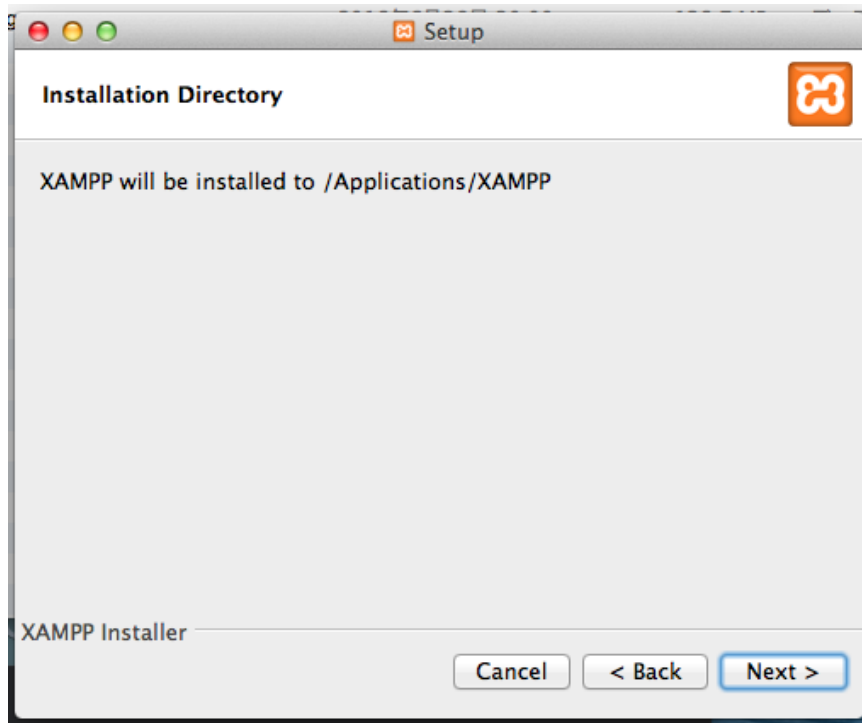
インストール：Mac版ー 2

- インストールするパッケージを選択



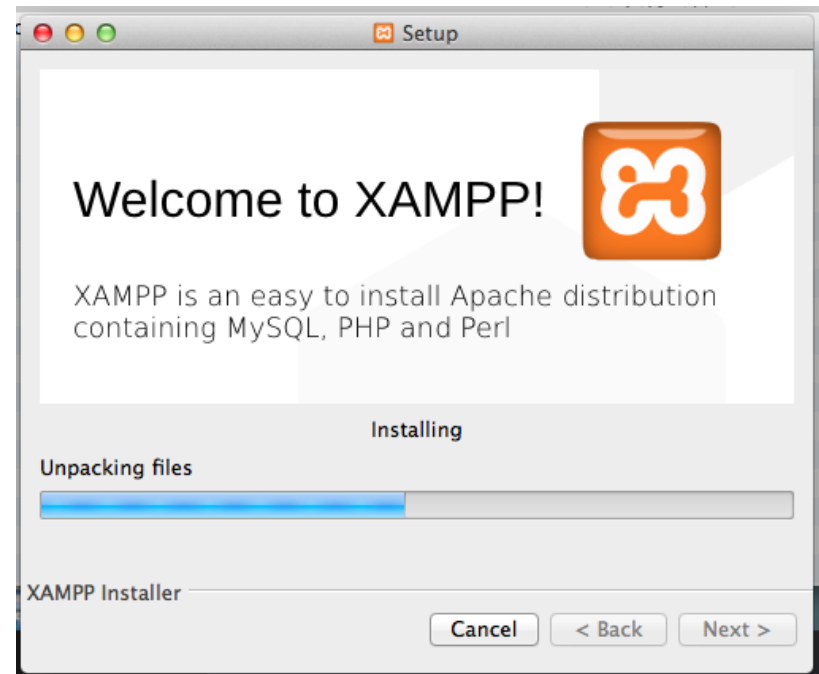
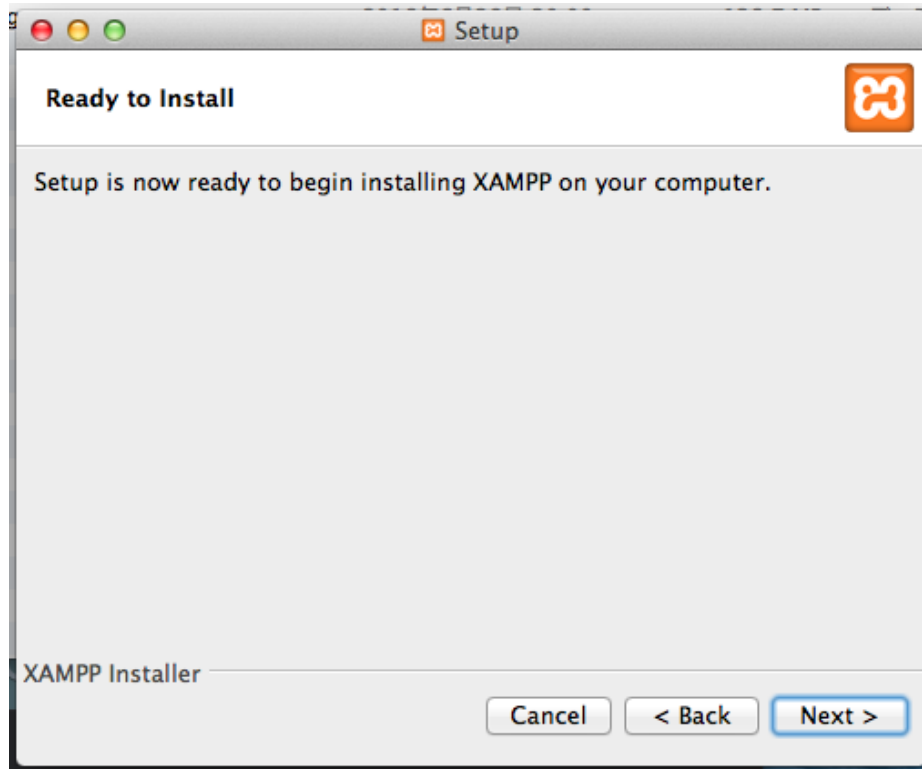
インストーラ : Mac版 - 3

- インストール先を確認



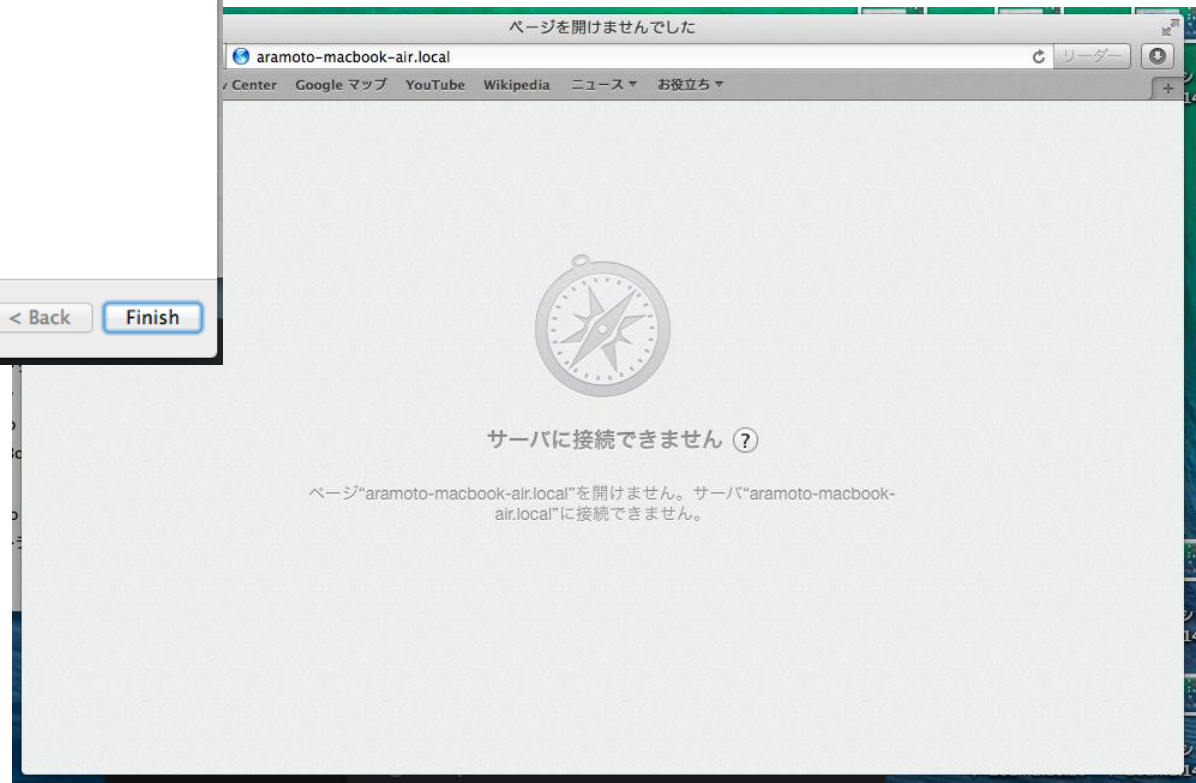
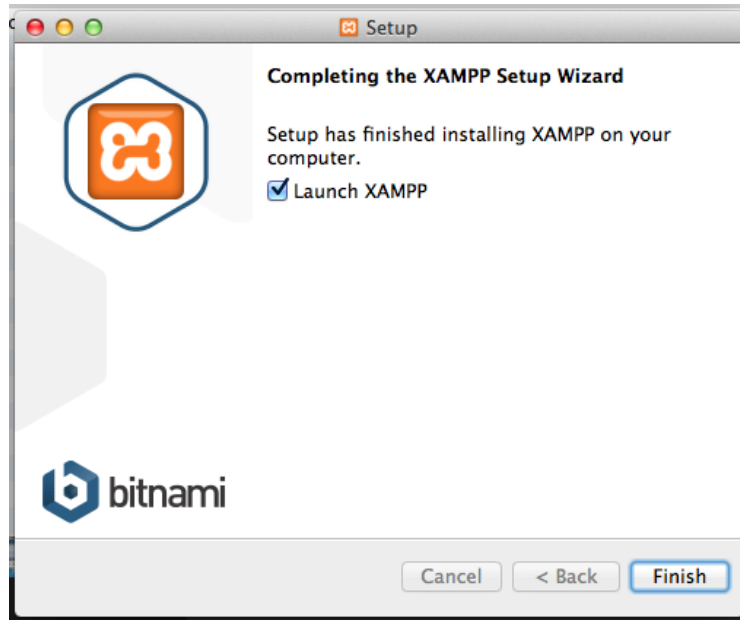
インストーラ : Mac版 - 4

- インストールを実行



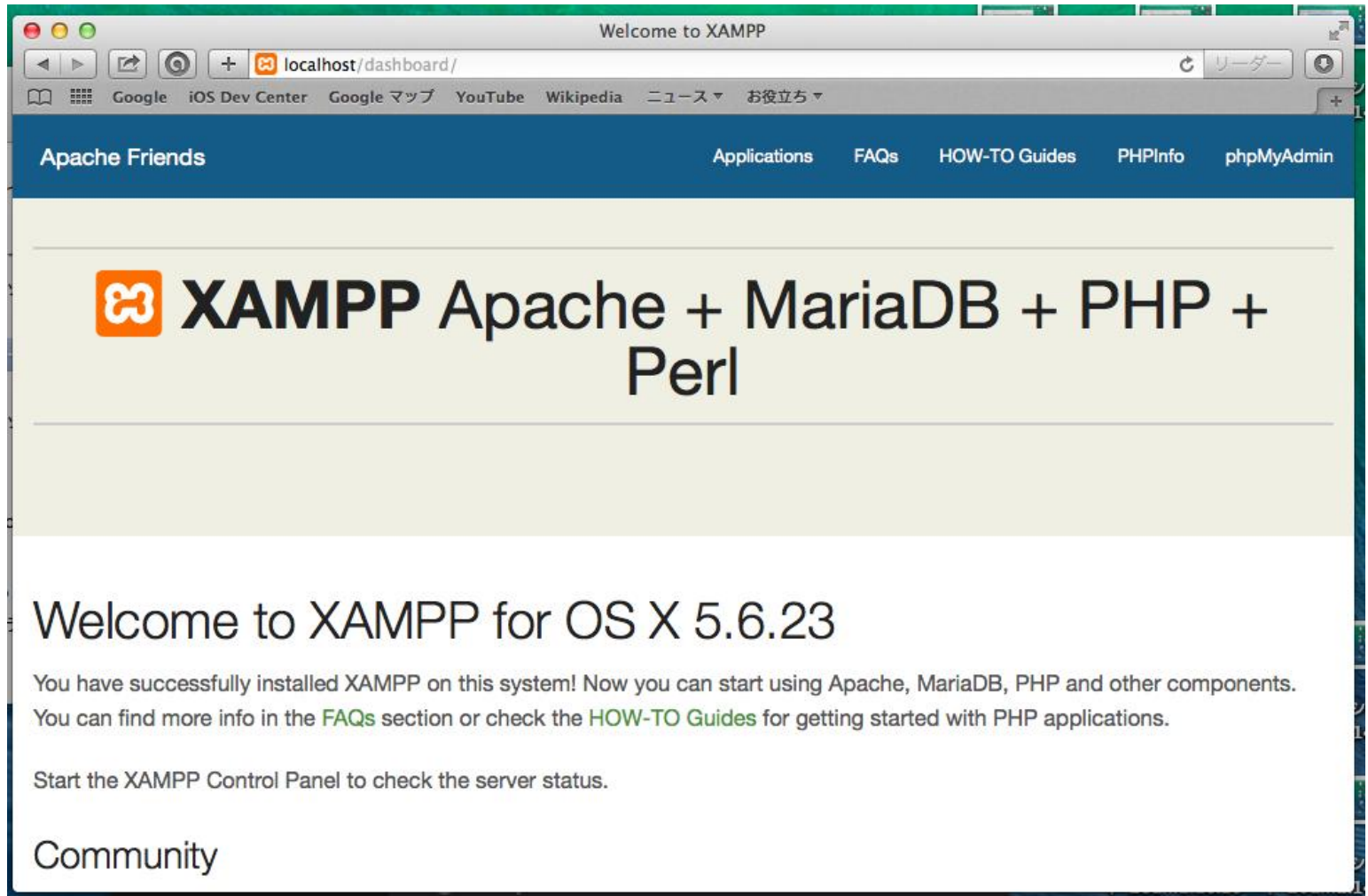
インストール：Mac版ー5

- インストール後に、確認画面が開くけど
 - ホスト名の問題でエラーになる



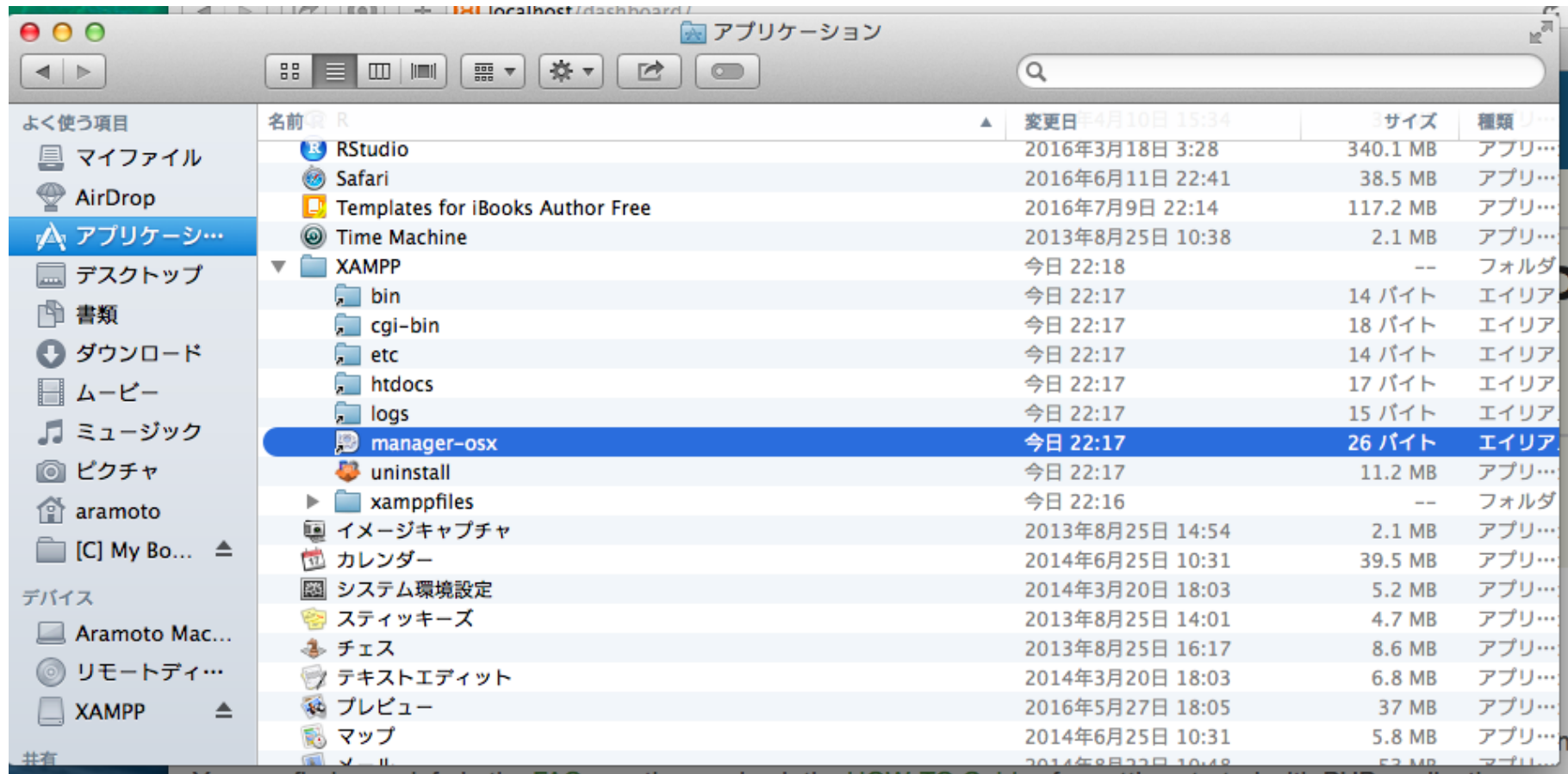
インストール：Mac版－6


- 手動でURLを入力し、この画面が出たらOK
 - <http://localhost>



インストール：Mac版ー 7

- 管理ツールのありか
 - manager-osx

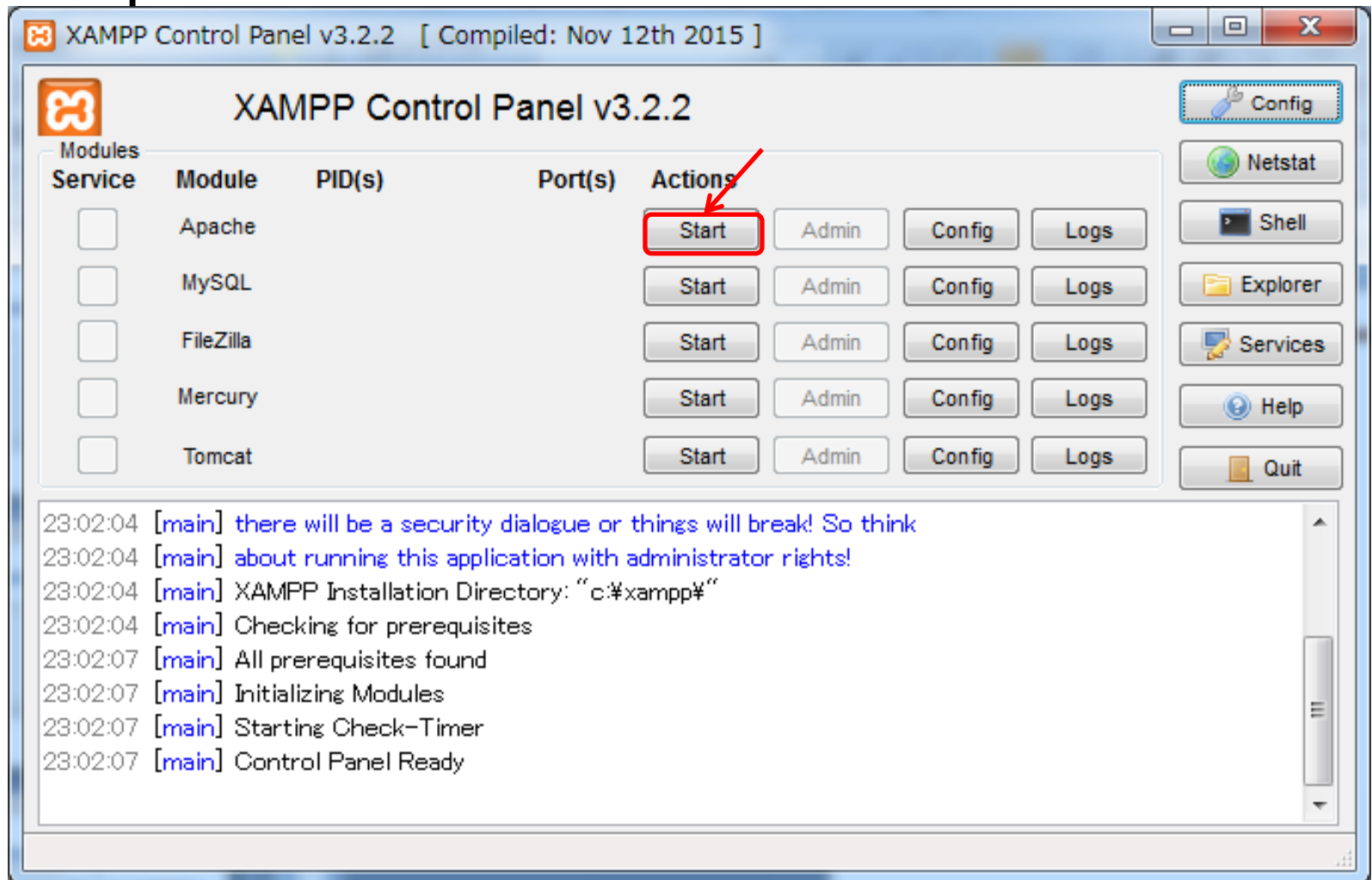




**まずは、
Webサーバを使ってみる**

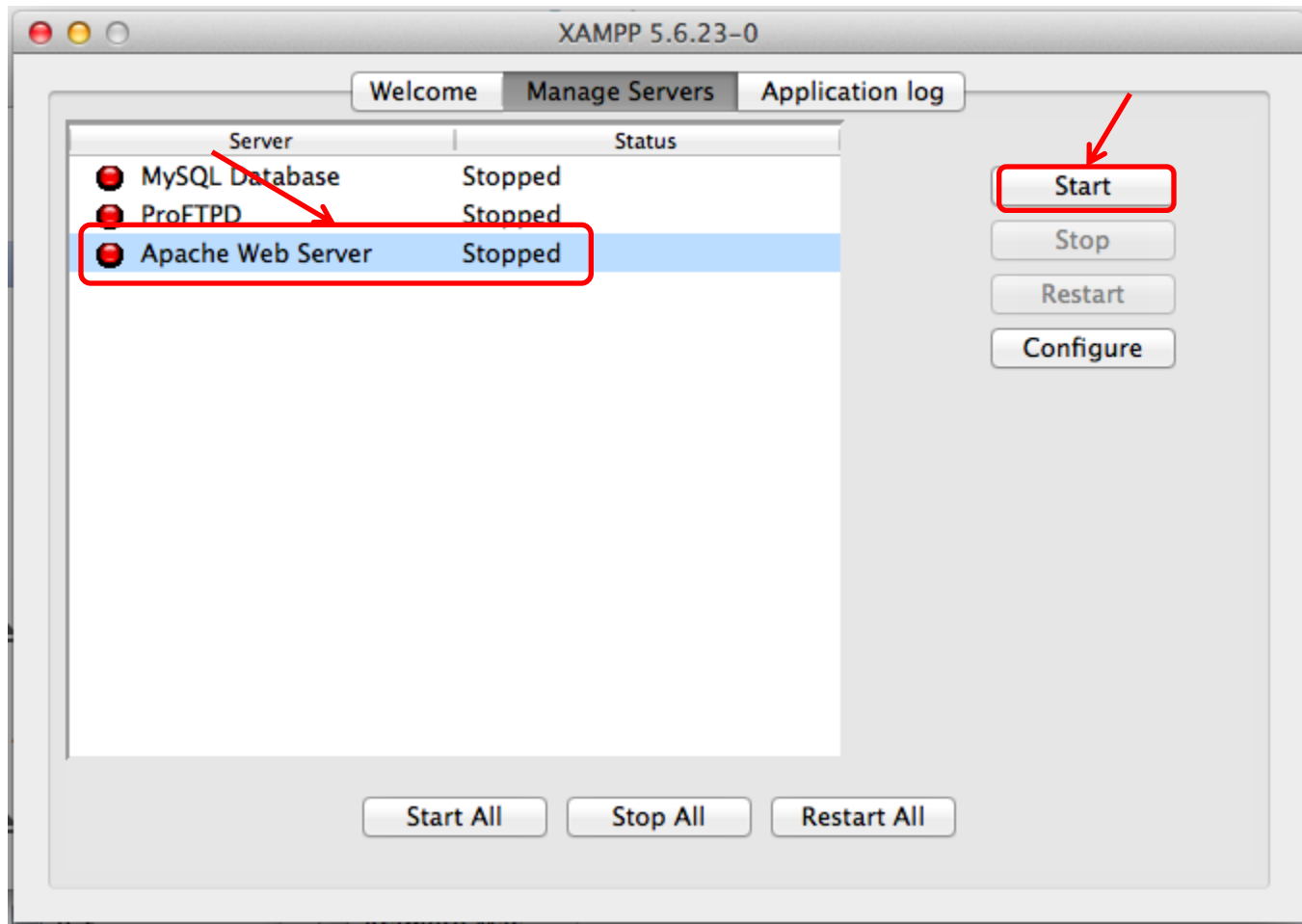
Webサーバーの起動：Windows版

- Apacheを起動する
 - Apache：世界で最も有名なWebサーバ



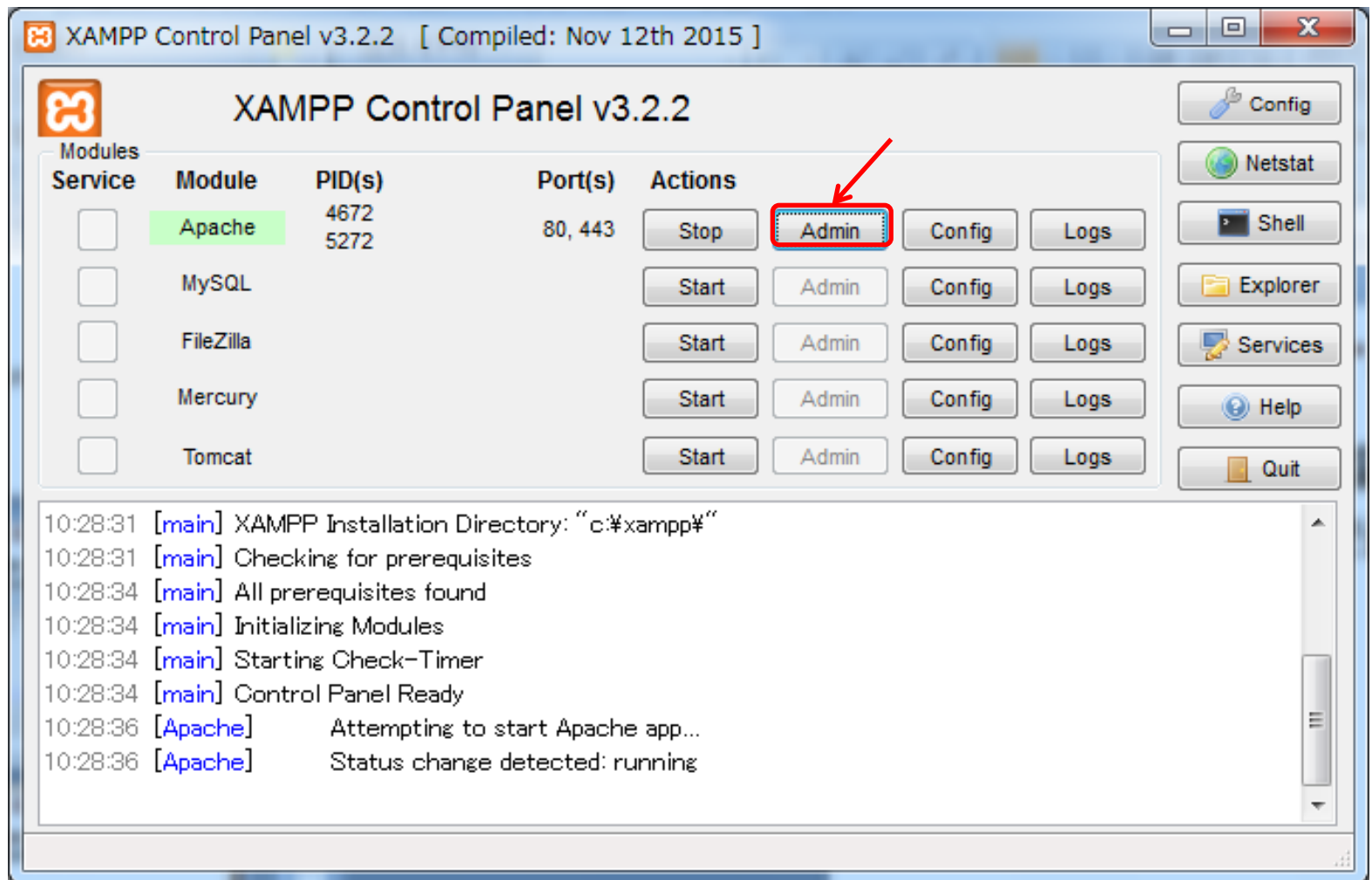
Webサーバーの起動：Mac版

- Apacheを起動する
 - Apache：世界で最も有名なWebサーバ



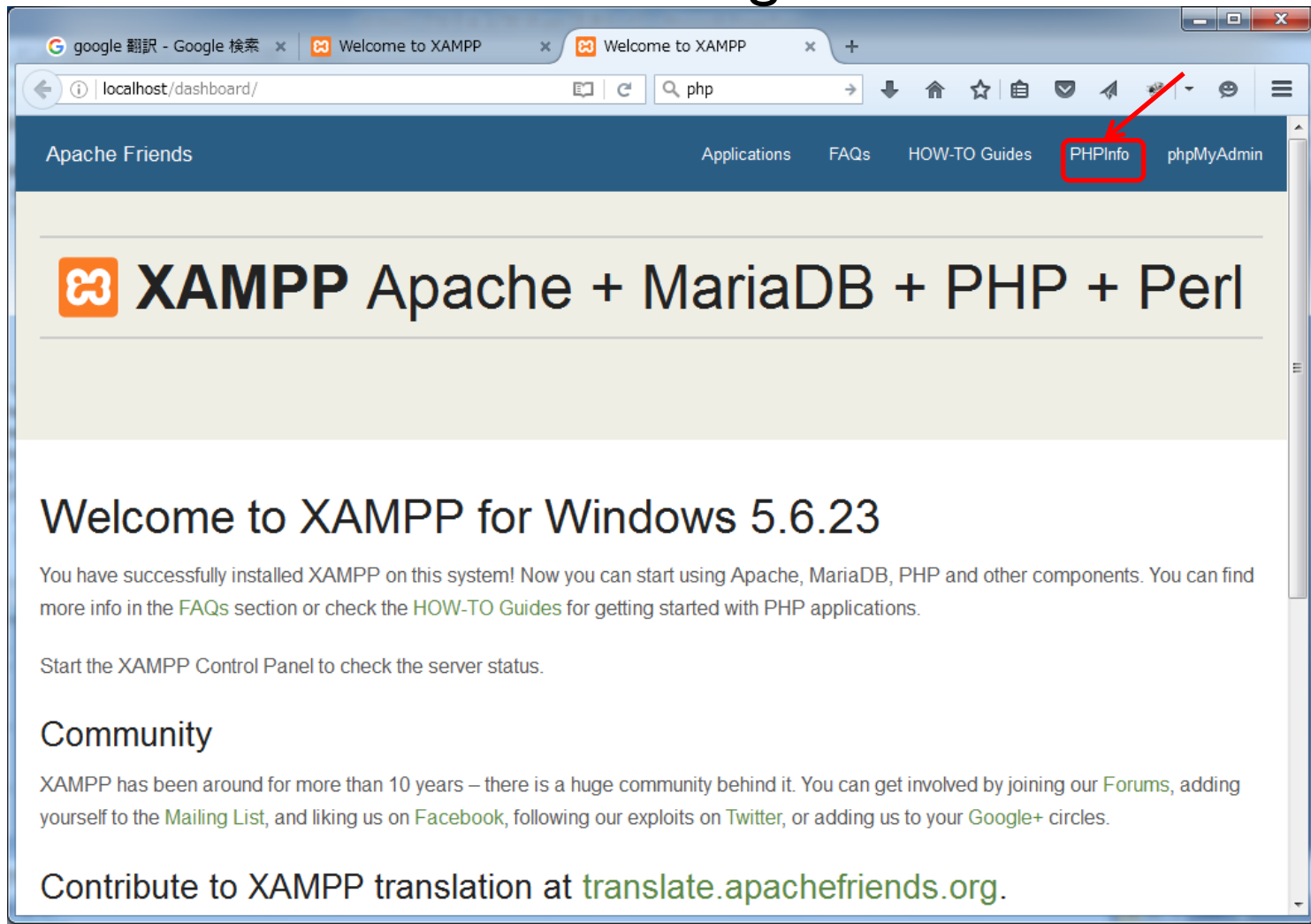
Webサーバの動作確認 - 1

- Apacheの動作を確認する
 - <http://localhost> をブラウザで開く



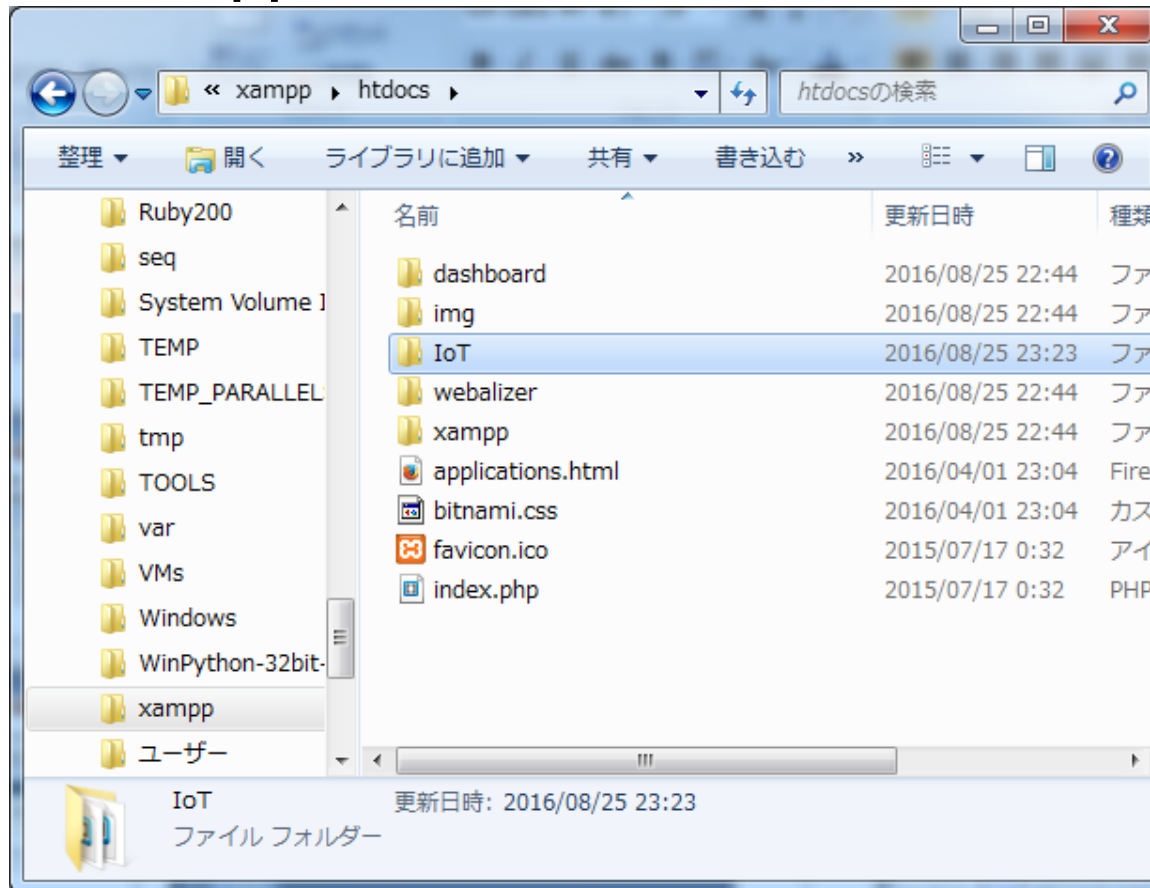
Webサーバの動作確認ー 2

- ApacheでPHPが動いていることを確認する
 - PHP：Personal Home Pageの略



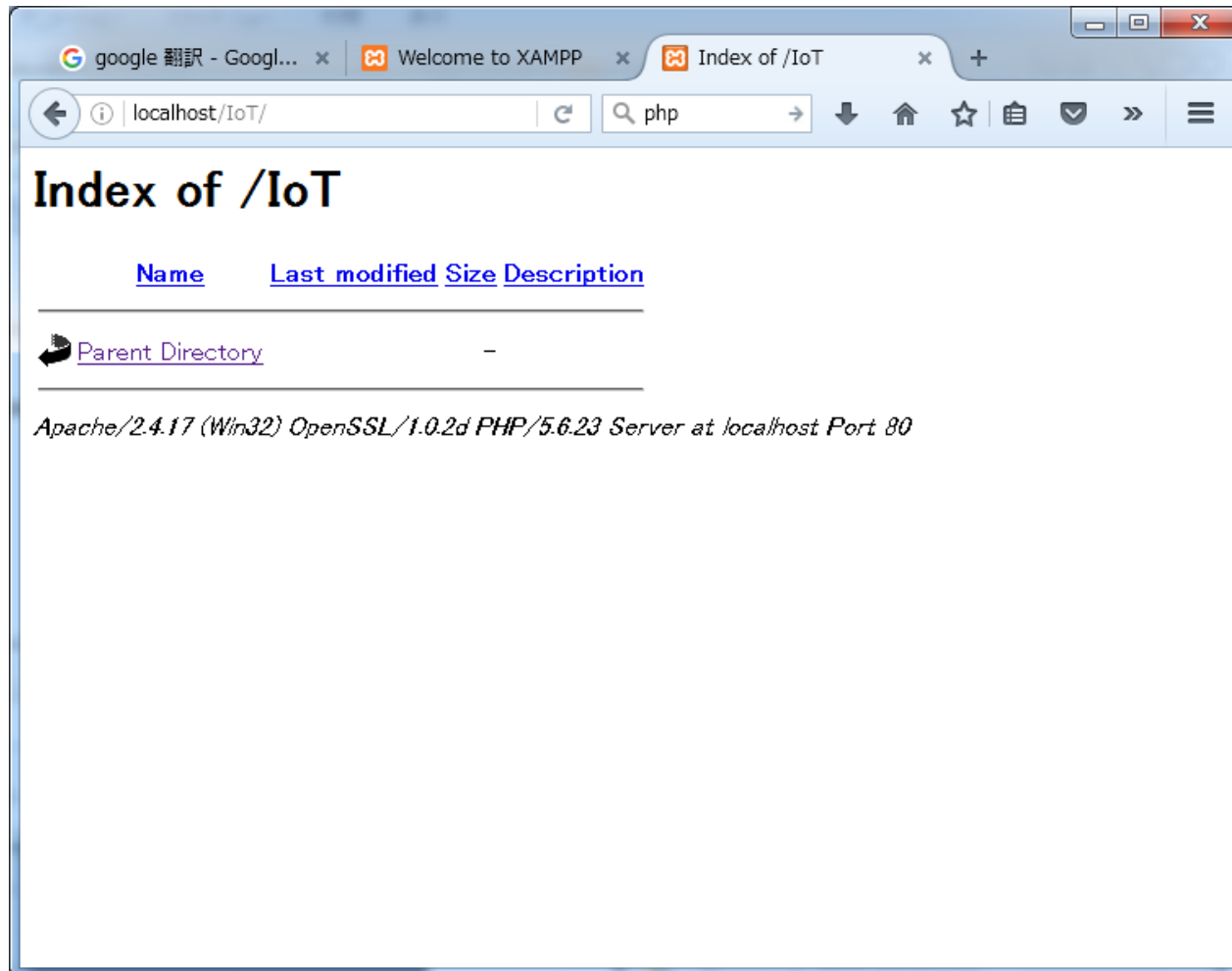
Webサーバにコンテンツを設置 - 1


- 「IoT」というディレクトリを作成
 - Windows : C:\xampp\htdocs の下
 - Mac : /Application/XAMPP/htdocs/ の下



Webサーバにコンテンツを設置 - 2

- ブラウザにアクセスして、動作を確認
 - <http://localhost/IoT/>





**次に、
RDBを使ってみる**

RDB とは

関係モデル [編集]

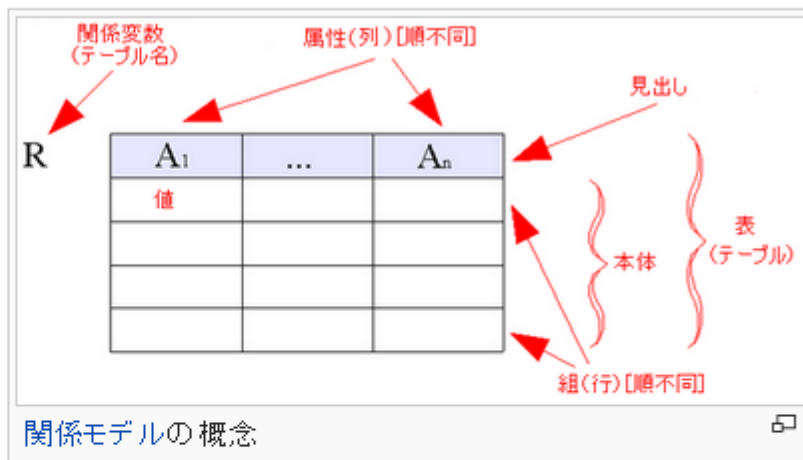
詳細は「[関係モデル](#)」を参照

関係モデルはIBMのエドガー・F・コードによって考案された^[2]現在もっとも広く用いられているデータモデルである。データベースの利用者は、クエリ(問い掛け)をデータベースに与え、データを検索したり、変更することができる。

データは表に似た構造で管理されるが、関係と呼ぶ概念でモデル化される。関係は組(タプル、表における行

に相当する)、属性(アトリビュート、表における列に相当する)、定義域(ドメイン)、候補キー(主キー)、外部キーなどによって構成される。SQLなどに代表されるデータベース言語(問い合わせ言語)を用いて、関係に対して制限・射影・結合・和・差・交わりなどの関係代数演算(集合演算を含む)ないし関係論理演算を行うことで結果を取り出す。

関係を複数持つことも可能で、互いを関連させることも可能である。



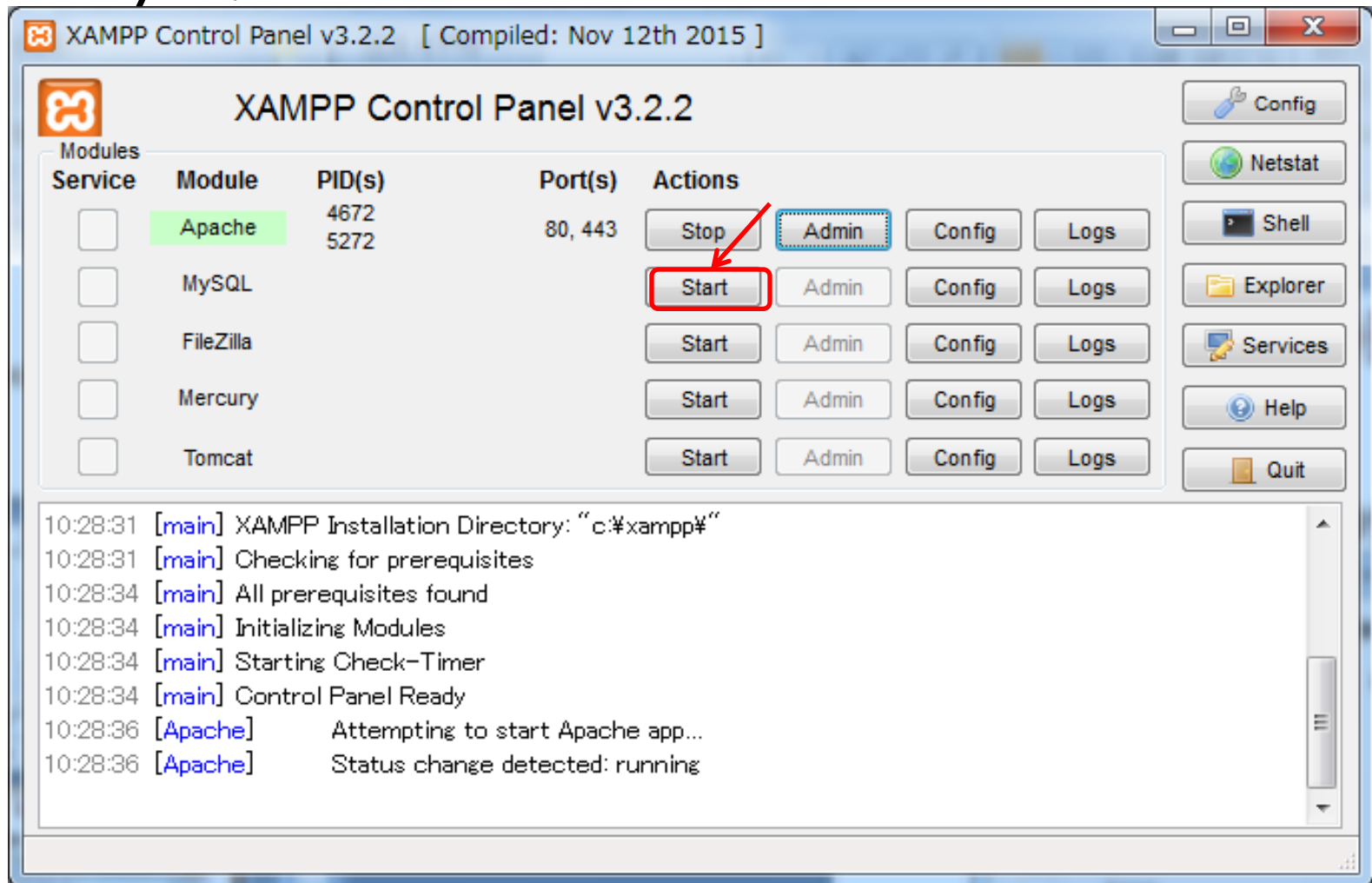
● 2言で説明すると

[ウィキペディア](#)より

- Excel表をサーバに格納し、プログラムでアクセスできる
- SQLで、集計 (sum) や平均(agv)などが簡単に出せる

RDBの起動：Windows版

- MySQLを起動する
 - MySQL：RDBの世界では、かなり有名



XAMPP Control Panel v3.2.2 [Compiled: Nov 12th 2015]

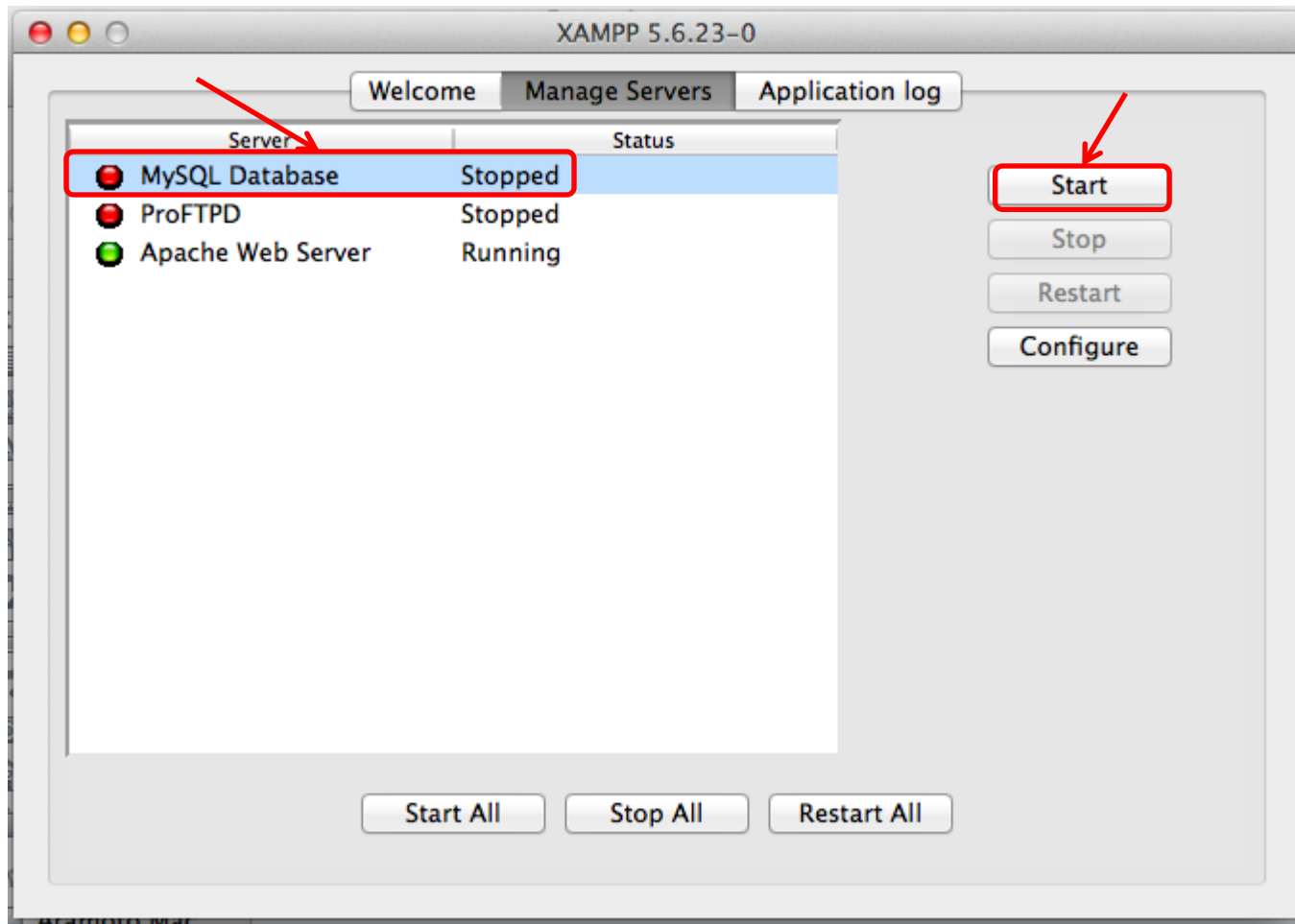
Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache	4672 5272	80, 443	Stop Admin Config Logs
<input type="checkbox"/>	MySQL			Start Admin Config Logs
<input type="checkbox"/>	FileZilla			Start Admin Config Logs
<input type="checkbox"/>	Mercury			Start Admin Config Logs
<input type="checkbox"/>	Tomcat			Start Admin Config Logs

Log window output:

```
10:28:31 [main] XAMPP Installation Directory: "c:\xampp%"  
10:28:31 [main] Checking for prerequisites  
10:28:34 [main] All prerequisites found  
10:28:34 [main] Initializing Modules  
10:28:34 [main] Starting Check-Timer  
10:28:34 [main] Control Panel Ready  
10:28:36 [Apache] Attempting to start Apache app...  
10:28:36 [Apache] Status change detected: running
```

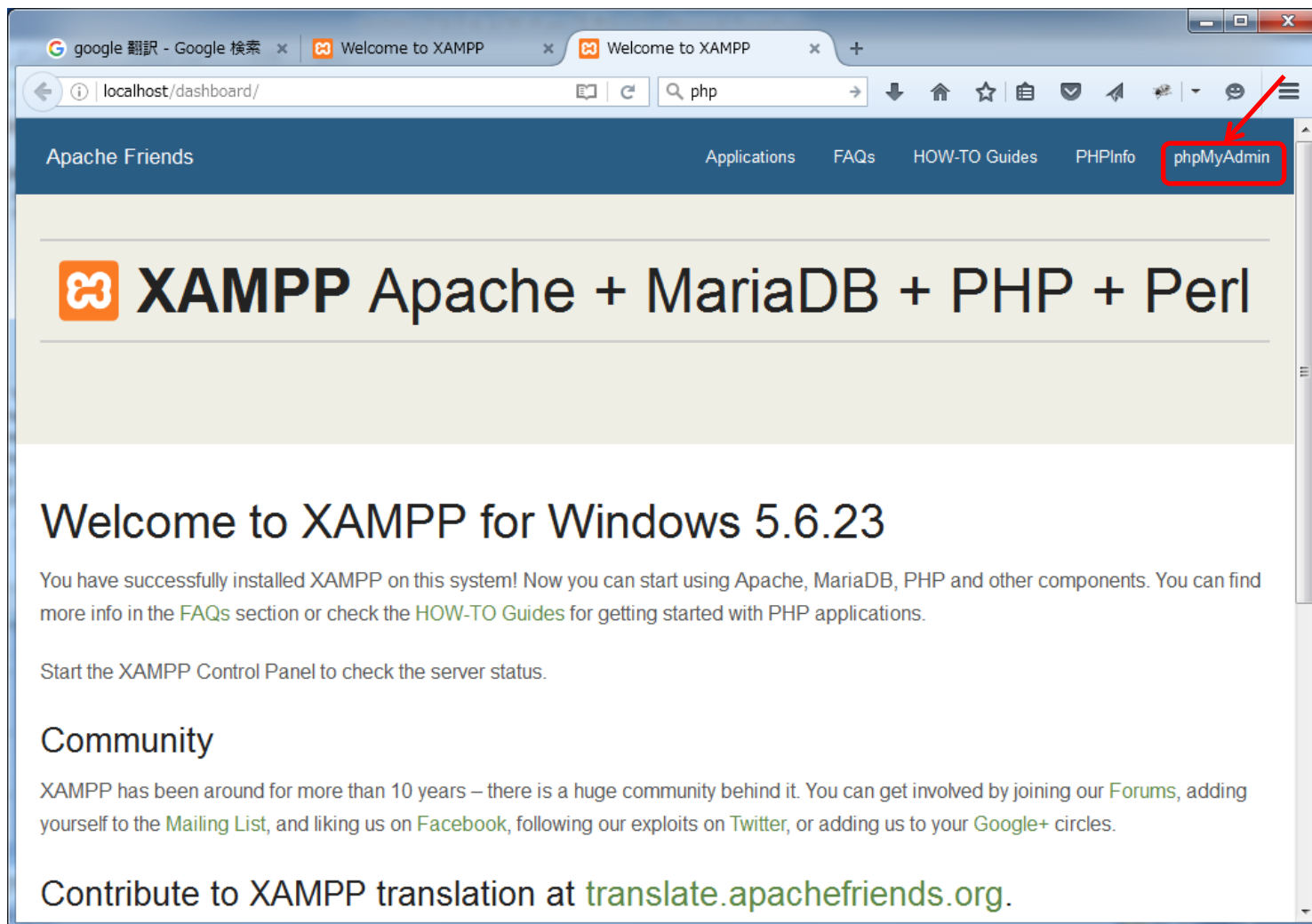
RDBの起動：Mac版

- MySQLを起動する
 - MySQL：RDBの世界では、かなり有名



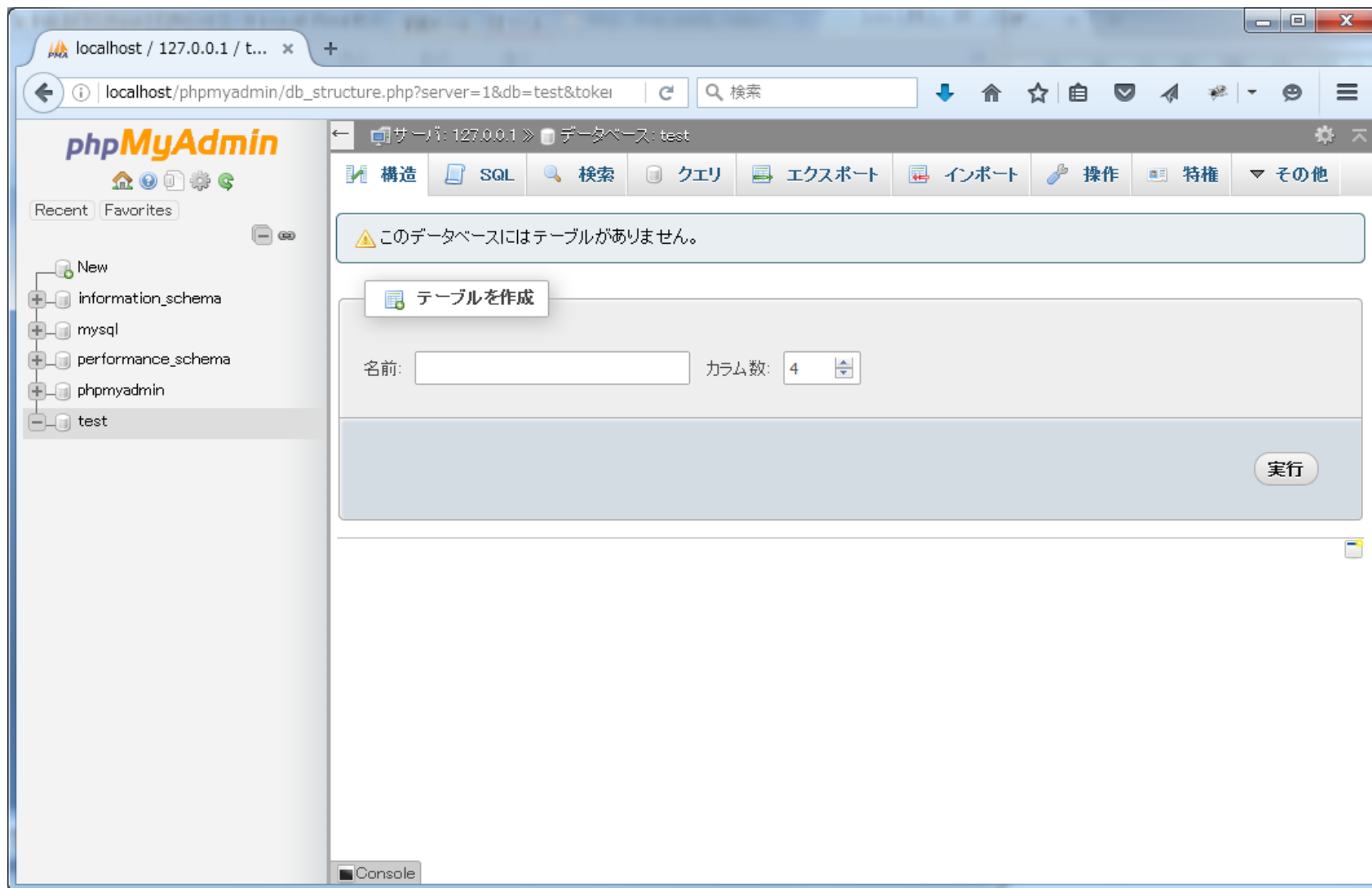
RDBの動作確認 - 1

- Apacheの動作確認画面から、管理画面を開く



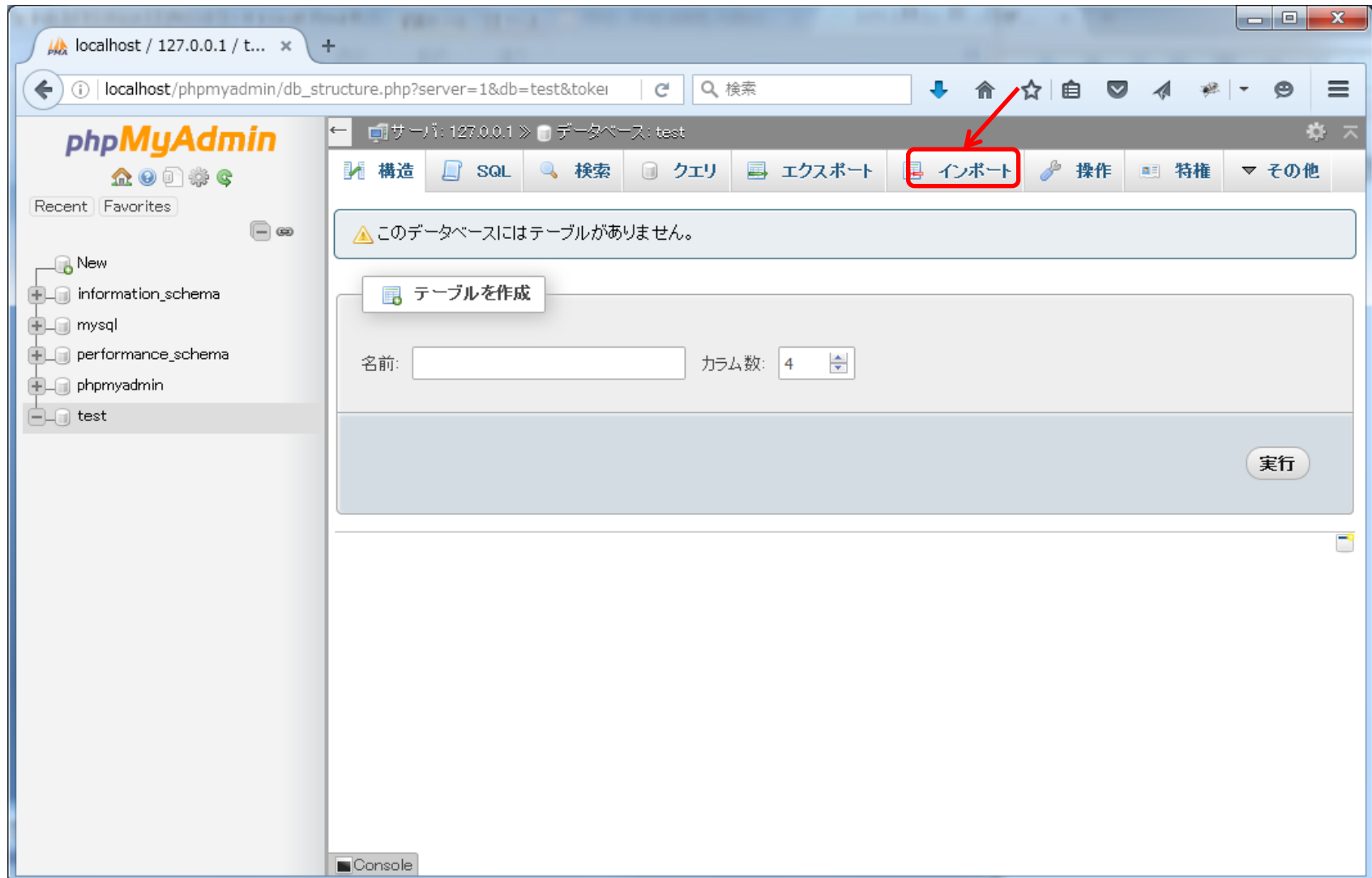
RDBの動作確認 - 2

- 管理画面



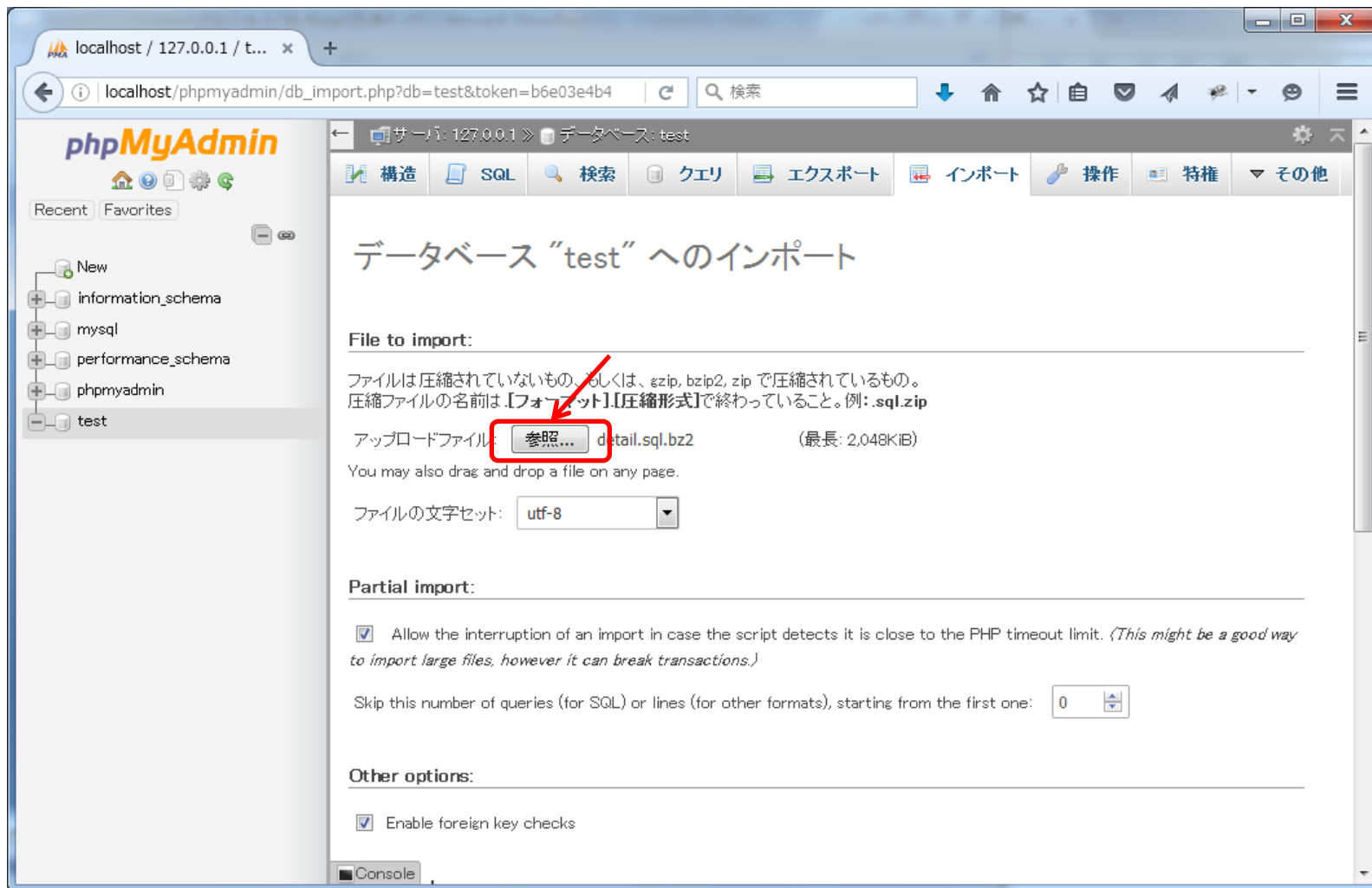
RDBにデータを登録－1

- 既存データのインポートをする



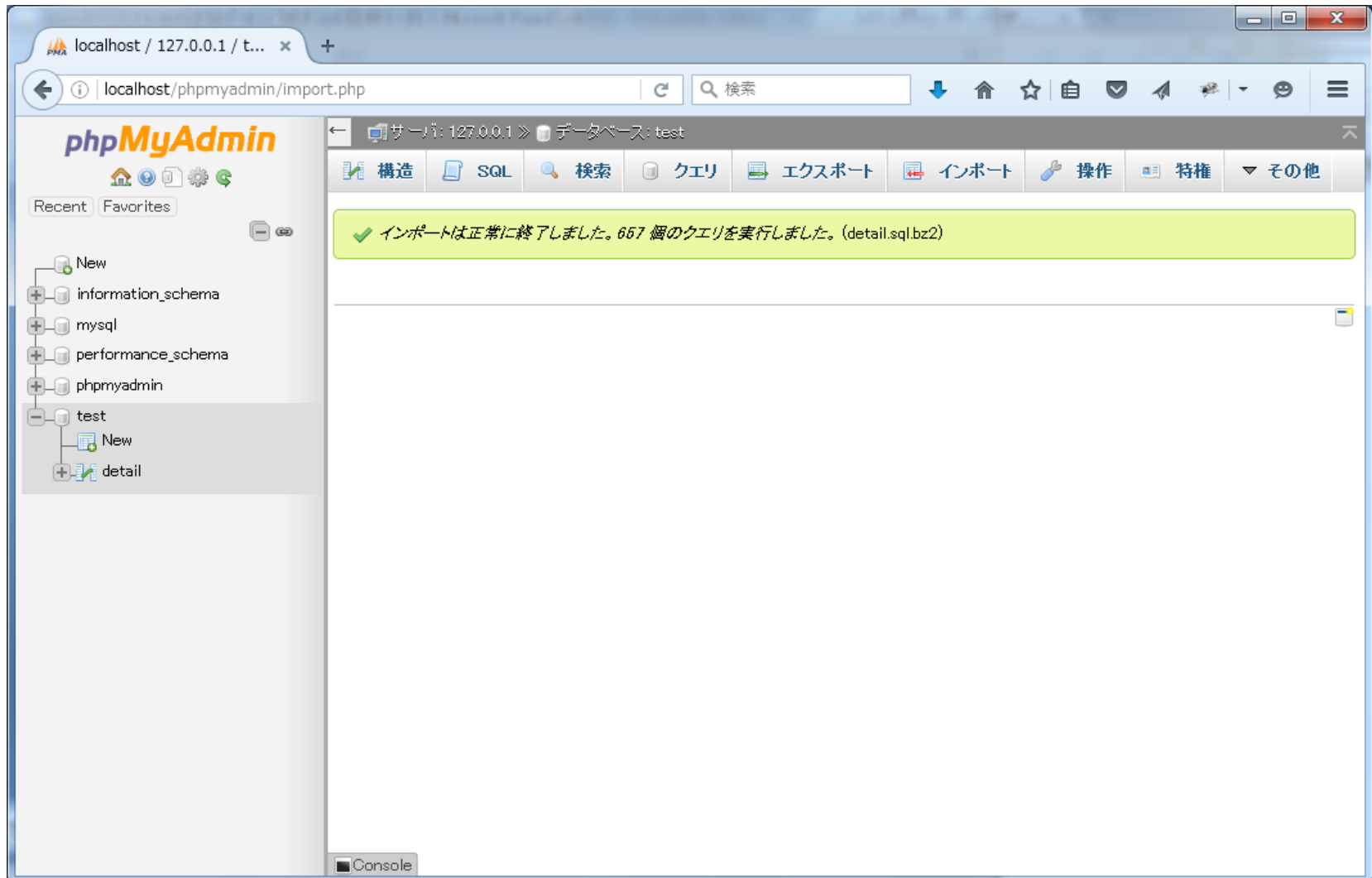
RDBにデータを登録 - 2

- detail.sql.bz2 を指定し、一番下の「実行」を押す
 - 私の部屋の、1カ月分の温度、加速度、動きのデータ



RDBにデータを登録－3

- しばらく～く待っていると、登録が完了



SQLを実行してみる - 1

- SQLの実行画面を呼び出す

The screenshot shows the phpMyAdmin interface for a database named 'test'. The 'SQL' tab is highlighted with a red box and a red arrow. The main content area displays a table with the following data:

テーブル	操作	行	データ型	照合順序	サイズ	オーバーヘッド
detail	表示 構造 検索 挿入 空にする 削除	444,202	MyISAM	ujis_japanese_ci	36.4 MiB	-
1 テーブル 合計		444,202	InnoDB	latin1_swedish_ci	36.4 MiB	0 バイト

Below the table, there is a 'テーブルを作成' (Create Table) form with the following fields:

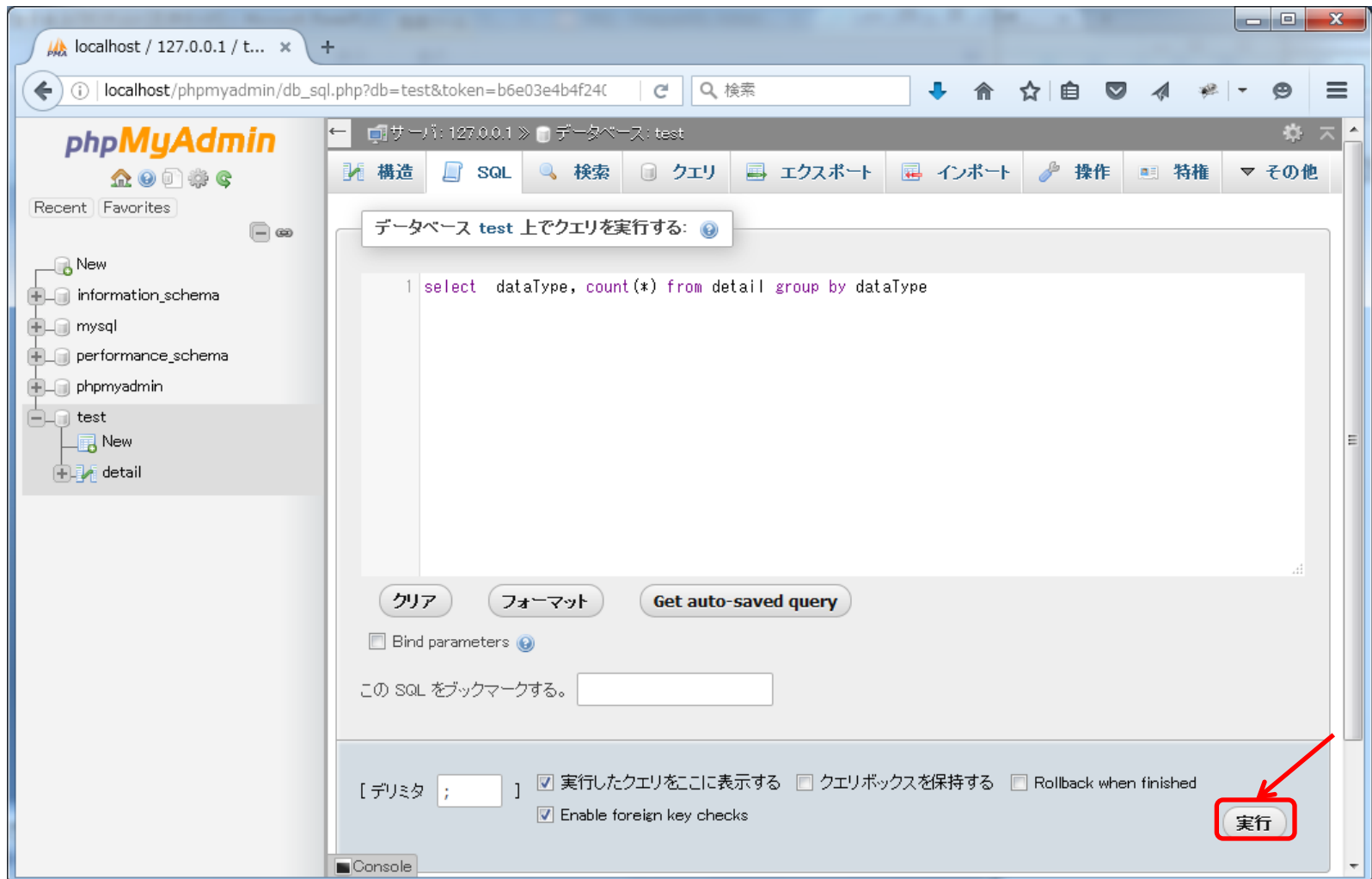
名前: カラム数: 4

実行

SQLを実行してみる - 2

- 集計してみる

- `select dataType, count(*) from detail group by dataType`



The screenshot shows the phpMyAdmin web interface in a browser window. The address bar shows the URL `localhost/phpmyadmin/db_sql.php?db=test&token=b6e03e4b4f24c`. The interface is in Japanese. On the left, a tree view shows the database structure, with the 'test' database selected and the 'detail' table visible. The main area is titled 'データベース test 上でクエリを実行する:' and contains a text area with the SQL query: `1 select dataType, count(*) from detail group by dataType`. Below the text area are buttons for 'クリア', 'フォーマット', and 'Get auto-saved query'. There is a checkbox for 'Bind parameters' which is unchecked. Below that is a text input field for bookmarking the SQL. At the bottom, there are several checkboxes: '実行したクエリをここに表示する' (checked), 'クエリボックスを保持する' (unchecked), 'Rollback when finished' (unchecked), and 'Enable foreign key checks' (checked). A red arrow points to the '実行' (Execute) button, which is highlighted with a red box.

もし、壊してしまったら

- テーブル「detail」を削除し、再登録

The screenshot shows the phpMyAdmin interface for a local server at 127.0.0.1. The database 'test' is selected. The table 'detail' is highlighted in the table list. The table's details are shown below:

テーブル	操作	行	データ型	照合順序	サイズ	オーバーヘッド
detail	★ 表示 構造 検索 挿入 空にする 削除	444,202	MyISAM	ujis_japanese_ci	36.4 MiB	-
1 テーブル 合計			444,202	InnoDB latin1_swedish_ci	36.4 MiB	0 バイト

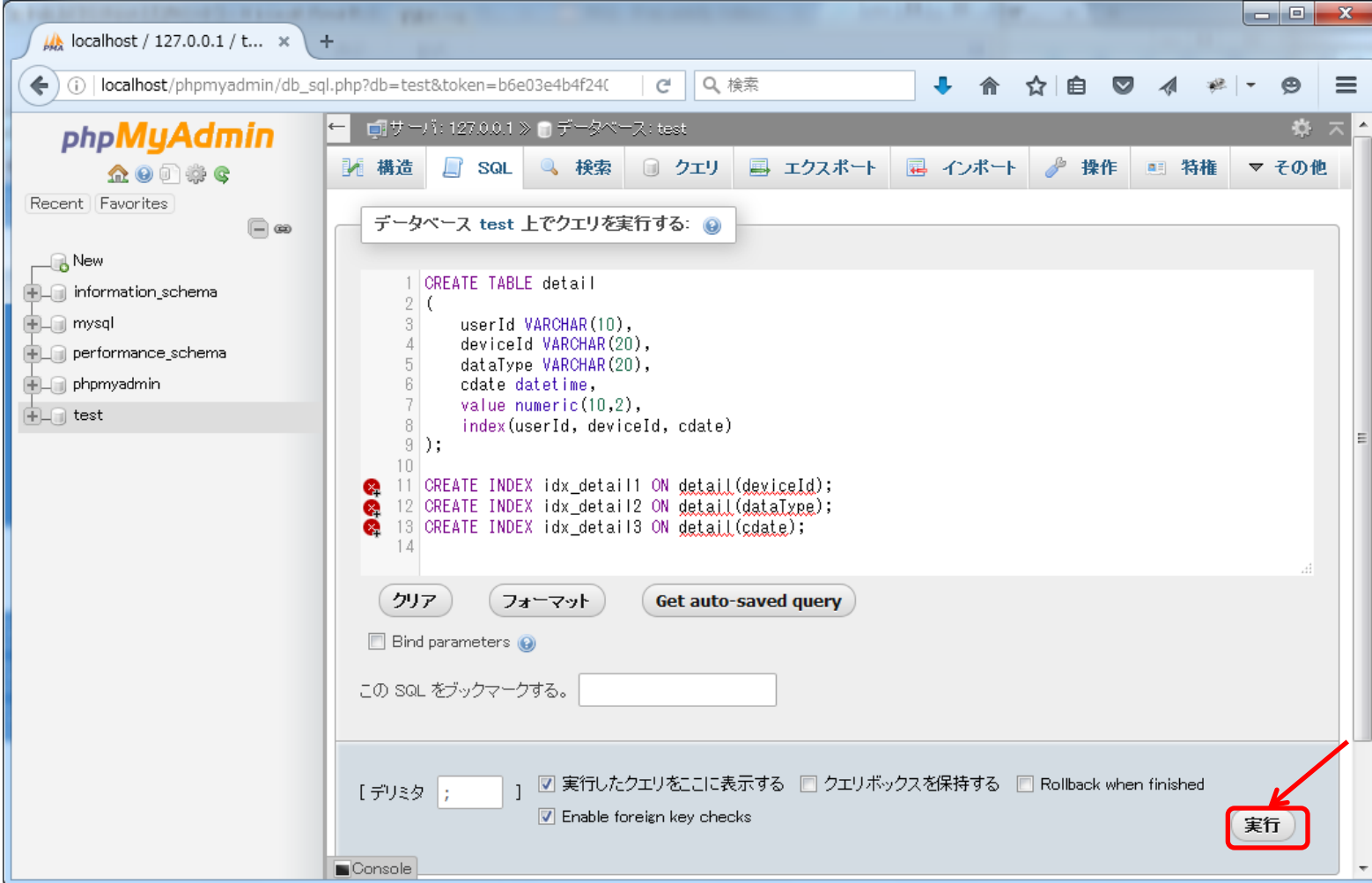
Below the table list, there is a section for creating a new table:

名前: カラム数: 4

実行

1 からデータを登録したければ、


- テーブル「detail」を削除し、「DDL.txt」を実行
 - 空のテーブルが作れます



The screenshot shows the phpMyAdmin interface for a database named 'test'. The main area displays SQL code for creating a table and indexes. The code is as follows:

```
1 CREATE TABLE detail
2 (
3     userId VARCHAR(10),
4     deviceId VARCHAR(20),
5     dataType VARCHAR(20),
6     cdate datetime,
7     value numeric(10,2),
8     index(userId, deviceId, cdate)
9 );
10
11 CREATE INDEX idx_detail1 ON detail(deviceId);
12 CREATE INDEX idx_detail2 ON detail(dataType);
13 CREATE INDEX idx_detail3 ON detail(cdate);
14
```

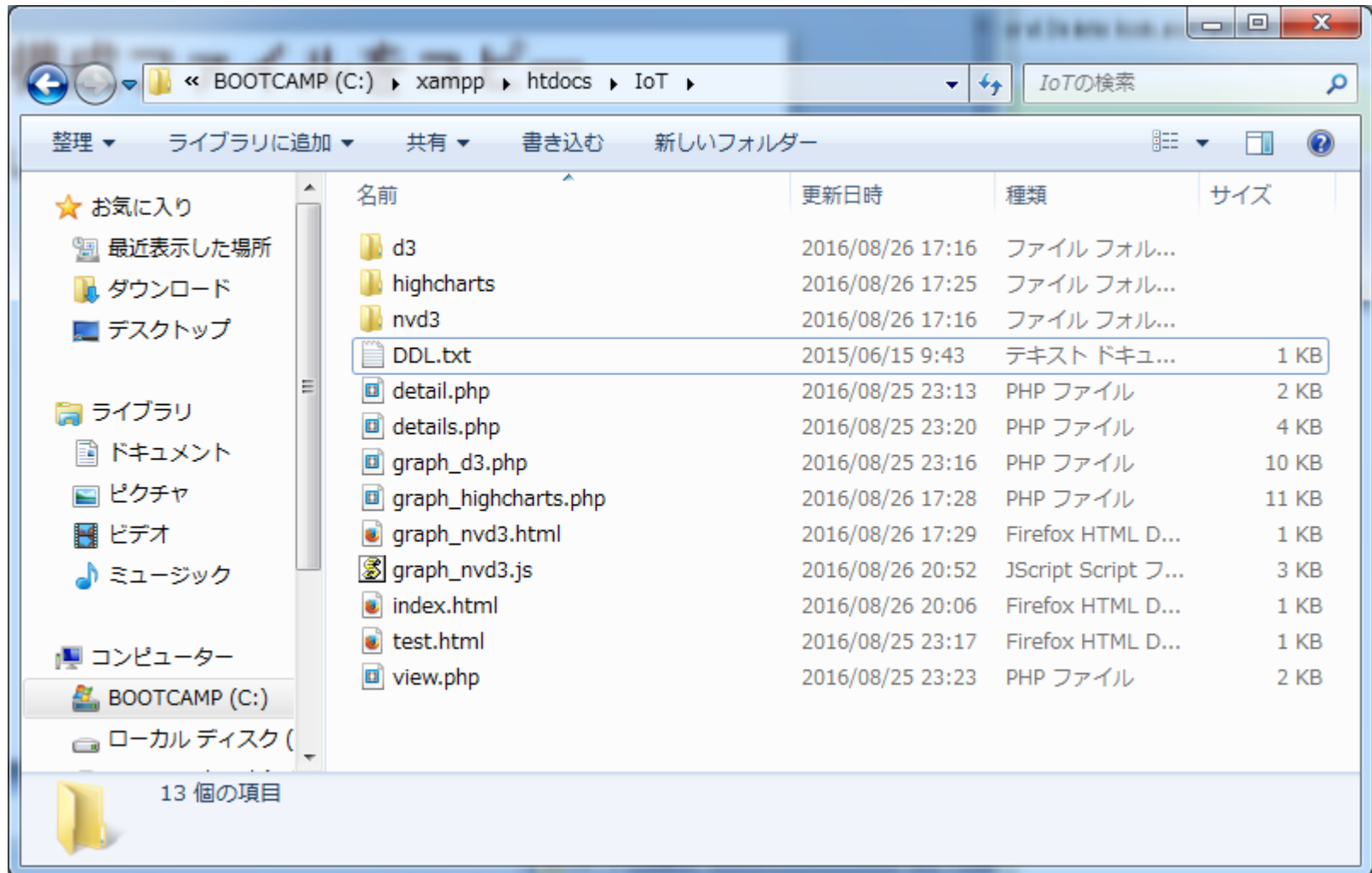
Below the code, there are buttons for 'クリア' (Clear), 'フォーマット' (Format), and 'Get auto-saved query'. There is also a checkbox for 'Bind parameters' and a text input for bookmarking the SQL. At the bottom right, there is a red arrow pointing to the '実行' (Execute) button.



アプリのインストール

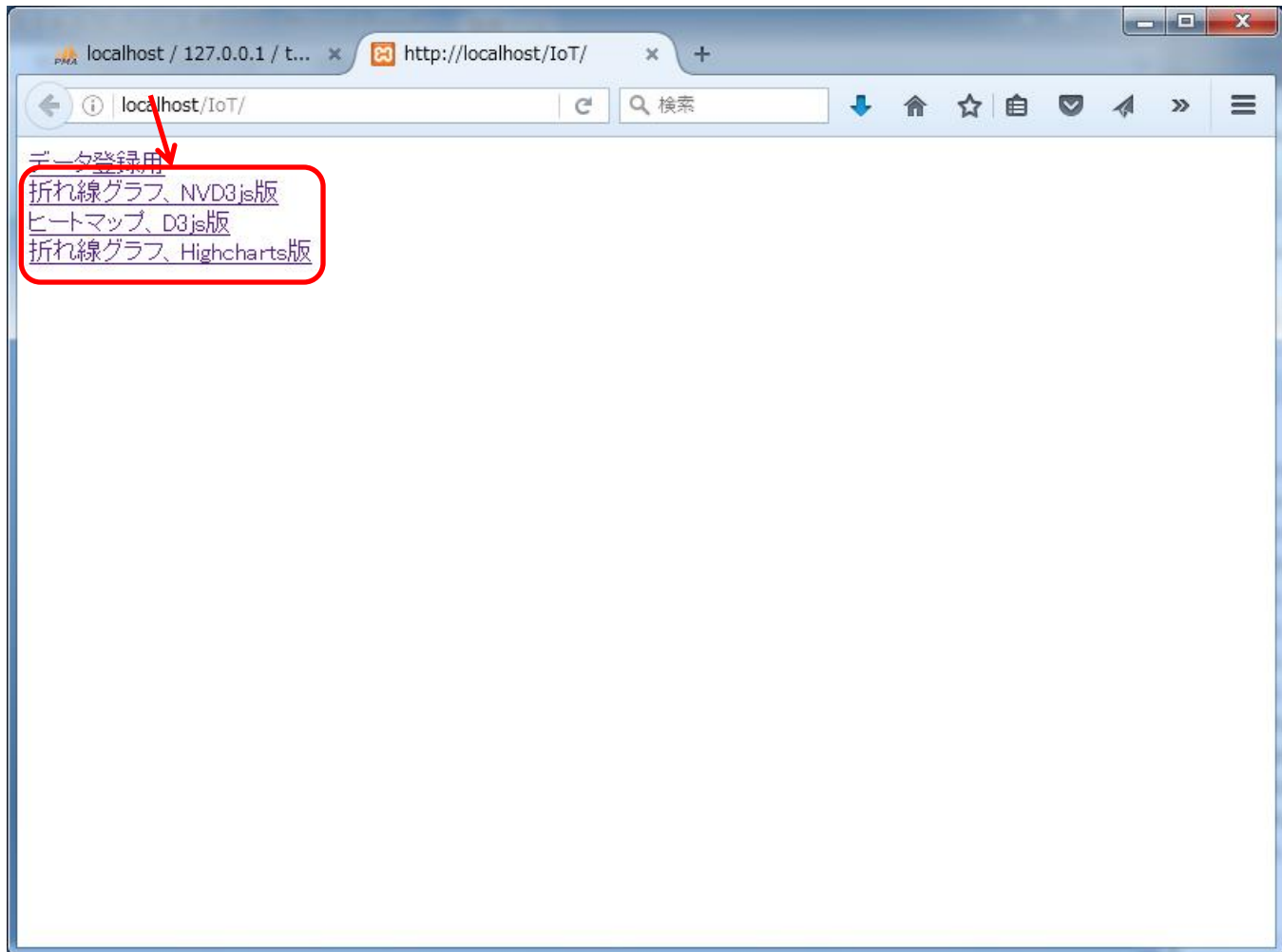
構成ファイルをコピー

- 作成した「IoT」ディレクトリ以下にコピー



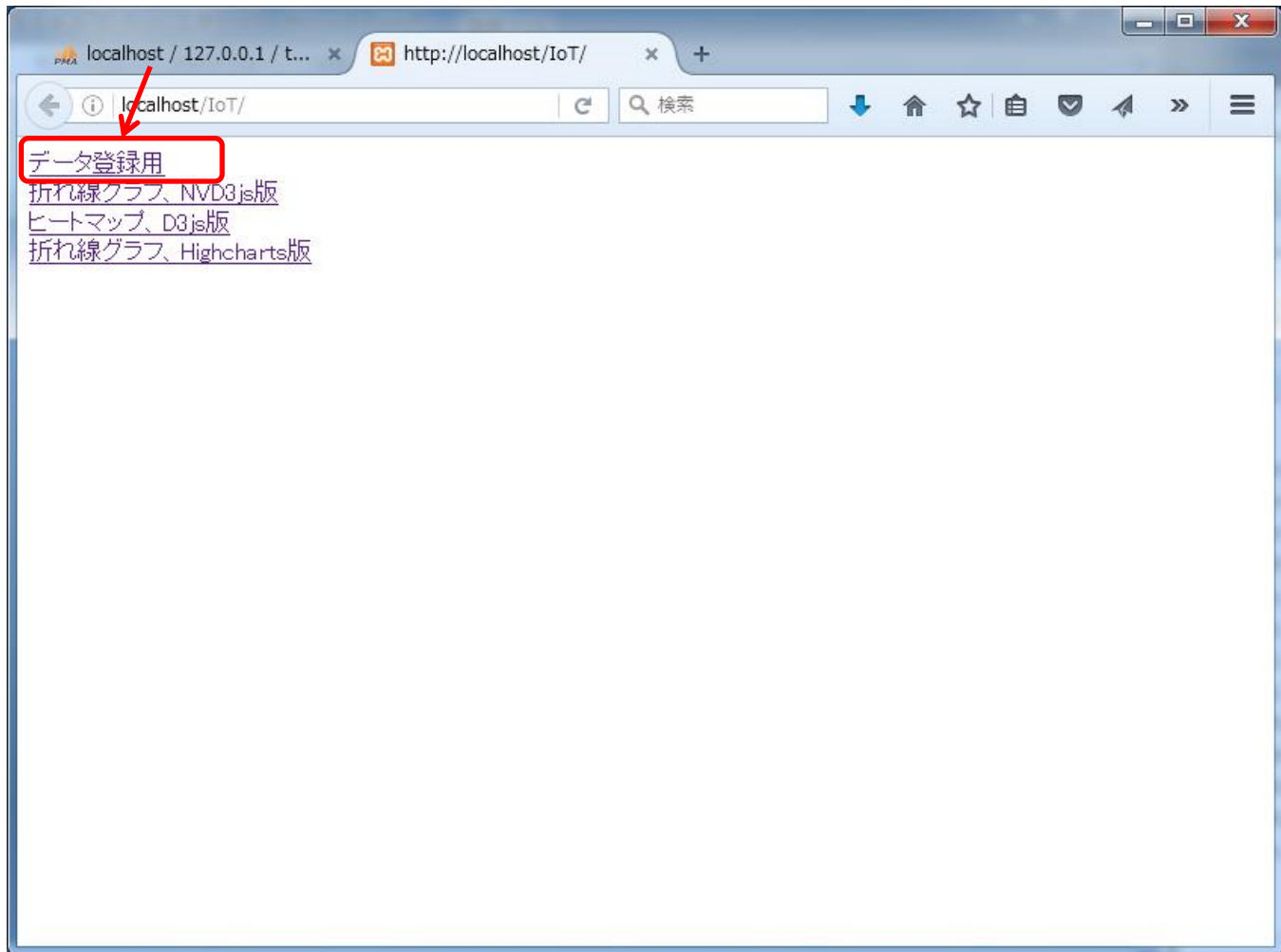
アクセスしてみる

- <http://localhost/IoT/>
 - データを使った画面が表示されるか確認



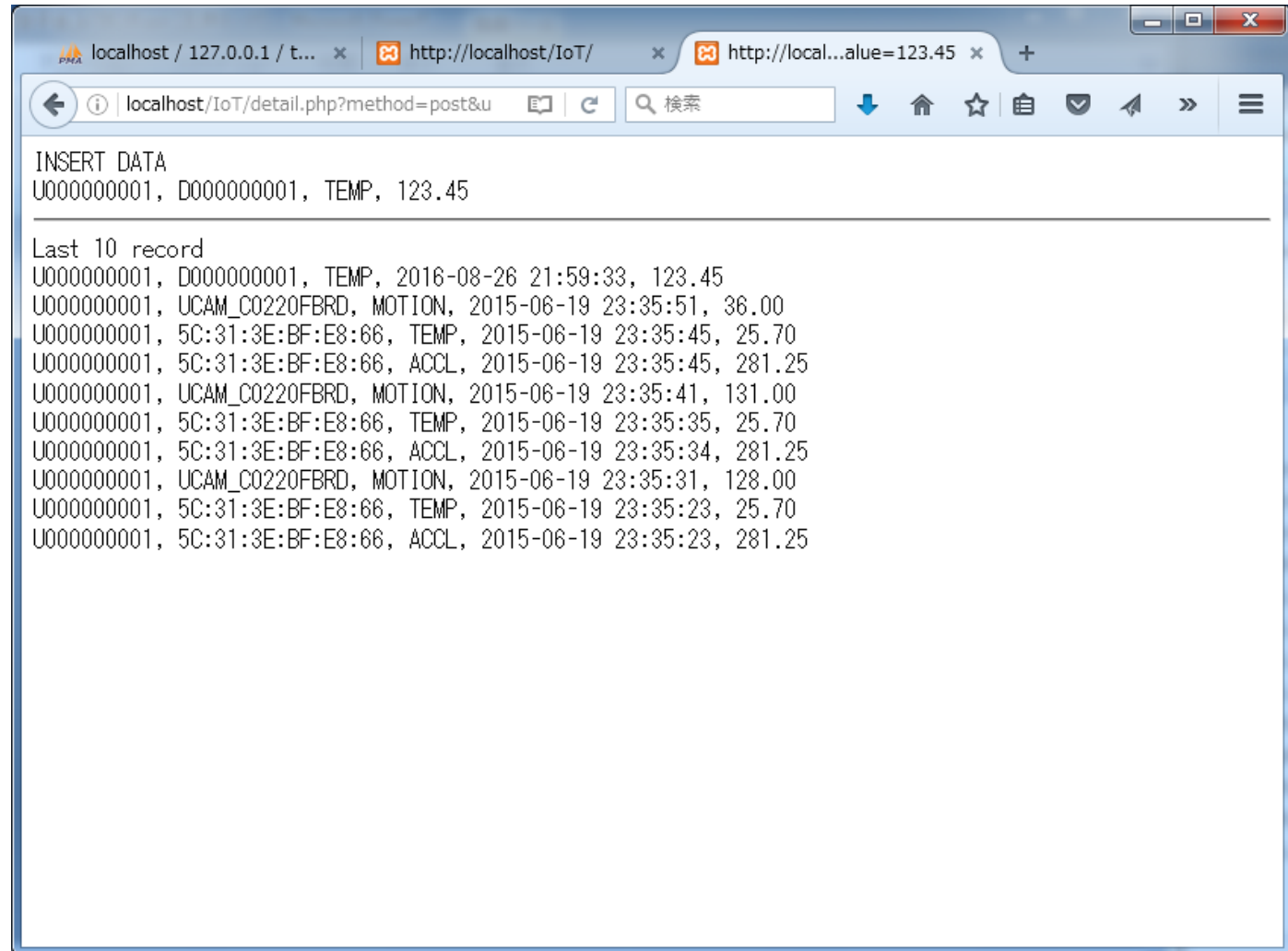
データの登録方法－1

- <http://localhost/IoT/>
 - データ登録のURLを生成する画面を開く



データの登録方法ー 2

- [http://localhost/IoT/detail.php?method=post&userId=U000000000001&deviceId=D0000000001&dataType=TEMP&value=123.45](http://localhost/IoT/detail.php?method=post&userId=U00000000001&deviceId=D0000000001&dataType=TEMP&value=123.45)



URLを叩くために必要なもの

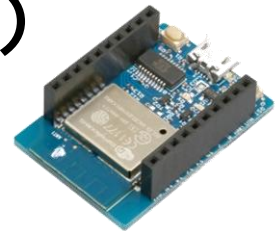
- Arduinoの場合

- イーサネットシールド (¥3,240-)

- <https://www.switch-science.com/catalog/2270/>

- 無線LAN付きのArduino互換品 (¥2,580-)

- <https://www.switch-science.com/catalog/2652/>



- Raspberry PIの場合

- PI2までは、有線LANが付いている

- PI3からは、無線LANも付いている

- 必要になる知識

- ネットワーク

- プログラミング



Arduino用サンプル（前半）

- コピペ用

- <http://aramoto.sakura.ne.jp/20151121/>

```
#include <SPI.h>
#include <Ethernet.h>
```

複数置く場合は、
赤字の部分を変える

```
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED }; // 自分のMACを決める
IPAddress server(219,94,129,54); // 接続先のIPアドレス：aramoto.sakura.ne.jp
int port = 80; // 接続先のポート番号
```

```
void setup() {
  Serial.begin(9600);

  Serial.println("DHCP Request");
  for(;Ethernet.begin(mac) == 0;) { // DHCPサーバからIPアドレスを取得
    Serial.println("Failed to configure Ethernet using DHCP");
    delay(10000);
  }
  Serial.println("START!");
}
```

Arduino用サンプル（後半）

```
void loop(){
  int val = analogRead(0);          // アナログ入力
  Serial.println ("val :" + String(val));

  EthernetClient client;
  if (client.connect(server,port)) {
    Serial.println("HTTP connected -----");
    // HTTPリクエストの文字列を生成
    String req = "GET /data/?id=123&val=" + String(val) + "&pass=hogehoge HTTP/1.0\r\n";
    req = req + "Host: aramoto.sakura.ne.jp\r\n"; // 仮想ホストなので、必要
    req = req + "\r\n";
    client.print(req); // リクエスト実行

    while (client.connected()){
      if (client.available()) { // read可能なデータがあるか？
        char c = client.read(); // レスポンスを読む
        Serial.print(c);      //そのままシリアルに表示
      } else {
        delay(10); // バッファが空なので、ちょっと待つ
      }
    }
    Serial.println("\r\nHTTP END -----");
    client.stop();
  } else {
    Serial.println("connection failed");
  }
  delay(1000);
}
```

URLは要変更

Raspberry PI のサンプル

- Pythonで記述する場合

```
try:
    userId = "U000000001"
    deviceId = "D000000001"
    type = "TEMP"
    value = 25
    conn = httplib.HTTPConnection("aramoto.sakura.ne.jp", timeout=10)
    get = "/IoT/detail.php?method=post&userId="+userId+"&deviceId="+deviceId+"&dataType="+type+"&value="+value
    print get
    conn.request("GET", get)
    r1 = conn.getresponse()
    print r1.status, r1.reason
except:
    pass
```

全体の流れが 体感できたでしょうか？

クラウドを利用すると、
楽チンです。

次の課題は、
セキュリティです。

