Development, Programming & DevOps

All about software development, including development operations, methodologies, programming, continuous integration etc.

- MySQL / MariaDB & MongoDB
  - How to do Everything in MySQL/MariaDB
  - Setup for Remote Access
  - Basic MongoDB Operations
- PHP, Javascript & HTML
  - Useful PHP Codes
  - Standard PHP/JS/HTML Procedures
  - How to do Everything in CakePHP 2.x
  - How to do Everything in AngularJS
  - How to do Everything in Javascript (Pure JS)
  - How to do Everything in Metro UI CSS
  - ReactJS REDUX Summary
  - Standard ReactJS Procedures
- Git and SVN
  - Patching with Git
  - Solving Cryptic SVN Errors

- CSS
  - CSS Utilities
- Everything About APIs
  - Facebook APIs
  - Common Parse API Operations
- CMS/E-Commerce
  - How to do Everything in Joomla!
  - Magento Development
  - Clearing Magento 2.x Orders
  - Refreshing Magento 2.x Static Contents

MySQL / MariaDB & MongoDB

Everything about MySQL / MariaDB and MongoDB

How to do Everything in MySQL/MariaDB

Useful Tricks

Convert a Column to Uppercase

UPDATE table_name SET `column_name` = UPPER(`column_name`)

Show and Change View Definer

```
SHOW FULL TABLES IN database_name WHERE TABLE_TYPE LIKE 'VIEW';
SHOW CREATE VIEW [view_name];
ALTER DEFINER = '[username]'@'[host]'  VIEW [view_name]  AS [select statement];
```

Alter Views

```
 ALTER VIEW <view name> AS [view statements]
```

MySQL Datatype for Different Password Hashes

It depends on the hashing algorithm you use. Hashing always produces a result of the same length, regardless of the input. It is typical to represent the binary hash result in text, as a series of hexadecimal digits. Or you can use the [UNHEX()](http://dev.mysql.com/doc/refman/5.5/en/string-functions.html#function_unhex) function to reduce a string of hex digits by half.

- MD5 generates a 128-bit hash value. You can use CHAR(32) or BINARY(16)
- SHA-1 generates a 160-bit hash value. You can use CHAR(40) or BINARY(20)
- SHA-224 generates a 224-bit hash value. You can use CHAR(56) or BINARY(28)
- SHA-256 generates a 256-bit hash value. You can use CHAR(64) or BINARY(32)
- SHA-384 generates a 384-bit hash value. You can use CHAR(96) or BINARY(48)
- SHA-512 generates a 512-bit hash value. You can use CHAR(128) or BINARY(64)
- BCrypt generates an implementation-dependent 448-bit hash value. You might need CHAR(56), CHAR(60), CHAR(76), BINARY(56) or BINARY(60)

NIST recommends using SHA-256 or higher for passwords. Lesser hashing algorithms have their uses, but they are known to be crackable.

You should salt your passwords before applying the hashing function. Salting a password does not affect the length of the hash result.

Chinese Support in MySQL

Convert entire database to UTF-8: `ALTER DATABASE databasename CHARACTER SET utf8 COLLATE utf8_unicode_ci;`

Convert entire table to UTF-8: `ALTER TABLE tablename CONVERT TO CHARACTER SET utf8 COLLATE utf8_unicode_ci;`

Convert field to UTF-8: `ALTER TABLE tablename MODIFY columnname columndef CHARACTER SET utf8 COLLATE utf8_unicode_ci;`

MySQL Grant Permission

```
Grant all on {dbname}.* to 'Id'@'localhost'  identified by 'password'
```

Convert All Table Columns' Charset and Collation

```
ALTER TABLE <table> CONVERT TO CHARACTER SET <charset> COLLATE <collation>;
```

Convert Database to MyISAM

```
#! /bin/sh

DB=' <DB Name>'
TABLES=$( /opt/lampp/bin/mysql -uroot --skip-column-names -B -D $DB -e 'show tables' )

for T in $TABLES
do
 /opt/lampp/bin/mysql -uroot -D $DB -e "ALTER TABLE $T ENGINE=MYISAM"
done
```

Solution to Common Problems

MariaDB Function Error From mysqldump

Use DELIMITER keyword to change end of function delimiter, e.g.:

```
DELIMITER //
CREATE FUNCTION counter () RETURNS INT
 BEGIN
 UPDATE counter SET c = c + 1;
 RETURN (SELECT c FROM counter LIMIT 1);
 END;
//
CREATE FUNCTION counter2 () RETURNS INT
 BEGIN
 UPDATE counter SET c = c + 2;
 RETURN (SELECT c FROM counter LIMIT 1);
 END;
//
DELIMITER ;
```

Error while sending QUERY packet

Change your maxallowedpacket by using one of the following methods:

- In mysql prompt, enter `SET GLOBAL max_allowed_packet=524288000;`
- Set `max_allowed_packet` in `my.ini`

Setup for Remote Access

1. Grant Privileges
    1. ```
       GRANT ALL ON <database>.* TO <user>@'%' IDENTIFIED BY '<password>';
       ```
2. Testing Remote Access
    1. ```
       mysql -uroot -p -h <host/ip> <database>
       ```

Basic MongoDB Operations

Queries and Indexes

Display query stats

```
db.<collection>.find({<query conditions>}).explain('executionStats')
```

Basic Document Operations

Find all documents in a collection

```
db.<collection>.find({})
```

Sorting documents

```
db.<collection>.find({}).sort()
```

Find one document in a collection

```
db.<collection>.findOne({})
```

Count documents in a collection

```
db.<collection>.count()
```

Insert a document

```
db.<collection>.insert({fieldA: 'a', fieldB: 'b'})
```

Updating a document

```
db.<collection>.update({_id: '<document id>'}, {'$set': {'fieldA': 'value'}})
```

Updating multiple documents

```
db.<collection>.update({<field condition>: '<condition>'}, {'$set': {'fieldA': 'value'}}, {multi
```

Delete a document

```
db.<collection>.remove({_id: '<document id>'})
```

Collection Operations

Remove a collection

```
db.<collection>.drop()
```

List all collections

```
show collections
```

Database Operations

List all databases

```
show databases
```

Switch to a database

```
use <database>
```

PHP, Javascript & HTML

All about PHP, JS and HTML including its frameworks

PHP, Javascript & HTML

Useful PHP Codes

Increment a Date by Month, Day or Year

Increment by month

```
$time = strtotime("2014-12-11");
$d = date("Y-m-d", strtotime("+1 month", $time));
```

Increment by day

```
$time = strtotime("2014-12-11");
$d = date("Y-m-d", strtotime("+1 day", $time));
```

Increment by year

```
$time = strtotime("2014-12-11");
$d = date("Y-m-d", strtotime("+1 year", $time));
```

Verify Email Accounts

```
<?php
function verifyEmail($toemail, $fromemail, $getdetails = false)
{
    $email_arr = explode("@", $toemail);
    $domain    = array_slice($email_arr, -1);
    $domain    = $domain[0];
    // Trim [ and ] from beginning and end of domain string, respectively
    $domain    = ltrim($domain, "[");
    $domain    = rtrim($domain, "]");
    if ("IPv6:" == substr($domain, 0, strlen("IPv6:"))) {
        $domain = substr($domain, strlen("IPv6") + 1);
    }
    $mxhosts = array();
    if (filter_var($domain, FILTER_VALIDATE_IP))
```

```
    $mx_ip = $domain;
else
    getmxrr($domain, $mxhosts, $mxweight);
$details = '';
if (!empty($mxhosts))
    $mx_ip = $mxhosts[array_search(min($mxweight), $mxhosts)];
else {
    if (filter_var($domain, FILTER_VALIDATE_IP, FILTER_FLAG_IPV4)) {
        $record_a = dns_get_record($domain, DNS_A);
    } elseif (filter_var($domain, FILTER_VALIDATE_IP, FILTER_FLAG_IPV6)) {
        $record_a = dns_get_record($domain, DNS_AAAA);
    }
    if (!empty($record_a))
        $mx_ip = $record_a[0]['ip'];
    else {
        $result = "invalid";
        $details .= "No suitable MX records found.";
        return ((true == $getdetails) ? array(
            $result,
            $details
        ) : $result);
    }
}

$connect = @fsockopen($mx_ip, 25);
if ($connect) {
    if (preg_match("/^220/i", $out = fgets($connect, 1024))) {
        fputs($connect, "HELO $mx_ip\r\n");
        $out = fgets($connect, 1024);
        $details .= $out . "\n";

        fputs($connect, "MAIL FROM: <$fromemail>\r\n");
        $from = fgets($connect, 1024);
        $details .= $from . "\n";
        fputs($connect, "RCPT TO: <$toemail>\r\n");
        $to = fgets($connect, 1024);
        $details .= $to . "\n";
        fputs($connect, "QUIT");
        fclose($connect);
        if (!preg_match("/^250/i", $from) || !preg_match("/^250/i", $to)) {
            $result = "invalid";
        } else {
            $result = "valid";
        }
    }
} else {
    $result = "invalid";
    $details .= "Could not connect to server";
}
if ($getdetails) {
    return array(
        $result,
        $details
```

```
        );
    } else {
        return $result;
    }
}

//Sample usage:
print_r(verifyEmail('<to email>', '<from email>'));
```

Calculate Coordinate Distances

```
function getDistance( $latitude1, $longitude1, $latitude2, $longitude2 ) {
    $earth_radius = 6371;

    $dLat = deg2rad( $latitude2 - $latitude1 );
    $dLon = deg2rad( $longitude2 - $longitude1 );

    $a = sin($dLat/2) * sin($dLat/2) + cos(deg2rad($latitude1)) * cos(deg2rad($latitude2)) * sin
    $c = 2 * asin(sqrt($a));
    $d = $earth_radius * $c;

    return $d;
    }

    $distance = getDistance( 56.130366, -106.34677099999, 57.223366, -106.34675644699 );
    if( $distance < 100 ) {
        echo "Within 100 kilometer radius";
    } else {
        echo "Outside 100 kilometer radius";
    }
}
```

Standard PHP/JS/HTML Procedures

Updating PHPUnit to Version 3.7

```
sudo mv /opt/lampp/lib/php/PHPUnit /opt/lampp/lib/php/PHPUnit.bak
curl -sS https://getcomposer.org/installer | php
sudo mv composer.phar /opt/lampp/bin/composer
sudo rm -rf /opt/lampp/share/openssl/certs/
sudo ln -s /etc/ssl/cert.pem /opt/lampp/share/openssl/cert.pem
sudo ln -s /etc/ssl/certs /opt/lampp/share/openssl/certs
composer global require "phpunit/phpunit=4.7.*"
```

PHP, Javascript & HTML

How to do Everything in CakePHP 2.x

Writing Test Case for AuthComponent's login() Function with CakePHP's Mocking

```
$this->controller = $this->generate('Users', array(
    'components' => array('Auth' => array('login')) //Mock all Auth methods
));

//This will make sure that Auth->login() function returns true
$this->controller->Auth->expects($this->once())
    ->method('login') //The method login()
    ->will($this->returnValue(true)); //And will return something for me
```

Loading other models from AppModel

Use `ClassRegistry::init('anothermodel');`

Irregular Naming Convention

```
Edit app/Config/bootstrap.php and add the following line:
Inflector::rules('plural', array('irregular' => array('staff' => 'staves')));
```

Expecting exception in CakePHP 2.x Test Case

```
$this->expectException();
$this->testAction('/controller/action');
```

Cakephp - Saving to Multiple Models from a Single Form

Controller: `$this->Model->saveAll($this->request->data);`

View:

```
echo $this->Form->input('<CURRENT MODEL>.<CURRENT MODEL FIELD>');
echo $this->Form->input('<MODEL A>.0.<MODEL A FIELD>');
echo $this->Form->input('<MODEL B>.0.<MODEL B FIELD>');
```

Problem Cakephp blank for certain controller actions only

Cause:

- Extra characters after the php tag "`?>`"
- Retrieved text is not in `UTF-8`

Solutions:

- Do not close php tag for php only files
- Remove the extra characters
- use `utf8_encode([text])` function to convert it before returning the data.

PHP, Javascript & HTML

How to do Everything in AngularJS

Prevent Route Change

1. Add `target="_self"` to all elements
2. Create new directive to prevent the defaults:
3. ```
   app.directive('a', function() {
     return {
       restrict: 'E',
       link: function(scope, elem, attrs) {
         if (attrs.ngClick || attrs.href === '' || attrs.href === '#') {
           elem.on('click', function(e) {
             e.preventDefault();
           });
         }
       }
     };
   });
   ```

Updating Model Within Directives

```
App.directive('myDirective', function ($parse) {
  return {
    require: 'ngModel',
    link: function (scope, elm, attrs, ctrl) {
      ctrl.$setViewValue(newValue);
      ctrl.$render();
      e.preventDefault();
      scope.$apply();
    });
  };
});
```
```

Making AngularJS Work with Bootstrap Vertical Button Group

```
<div class="btn-group-vertical" ng-class="{'active':selected}">
    <input type="checkbox" autocomplete="off" ng-checked="selected" />
</div>
```

<Enter> Key Event Directive

JS:

```
app.directive('ngEnter', function () {
  return function (scope, element, attrs) {
    element.bind("keydown keypress", function (event) {
      if(event.which === 13) {
        scope.$apply(function (){
          scope.$eval(attrs.ngEnter);
        });
        event.preventDefault();
      }
    });
  };
});
```

HTML Usage:

```
<div ng-app="" ng-controller="MainCtrl">
    <input type="text" ng-enter="doSomething()">
</div>
```

Passing Functions to Directives

HTML:

```
<test color1="color1" update-fn="updateFn(msg)"></test>
```

JS:

```
var app = angular.module('dr', []);

app.controller("testCtrl", function($scope) {
  $scope.color1 = "color";
    $scope.updateFn = function(msg) {
```

```
    alert(msg);
  }
});

app.directive('test', function() {
  return {
    restrict: 'E',
    scope: {
      color1: '=',
      updateFn: '&'
    },
    // object is passed while making the call
    template: "<button ng-click='updateFn({msg : \"Hello World!\"})'>Click</button>",
    replace: true,
    link: function(scope, elm, attrs) {
    }
  }
});
```

Angular JS Best Practices

1. Do not put the main controller into the main module, instead the main controller should be declared in a new module, this will make the application more modular e.g.

```
angular.module('app', ['Controller'])
angular.module('Controller', []).controller('Controller', function($scope) {
  $scope.something = 100
})
```

2. Do not call functions for ng-show and ng-hide directives, as it may degrade performance

3. Using too much $scope.$watch is going to degrade performance, try only watch what you really need to and remove the watchers when it's not needed anymore

How to do Everything in Javascript (Pure JS)

Mobile user-agent detection

```
var isMobile = {
  Android: function() {
    return navigator.userAgent.match(/Android/i);
  },
  BlackBerry: function() {
    return navigator.userAgent.match(/BlackBerry/i);
  },
  iOS: function() {
    return navigator.userAgent.match(/iPhone|iPad|iPod/i);
  },
  Opera: function() {
    return navigator.userAgent.match(/Opera Mini/i);
  },
  Windows: function() {
    return navigator.userAgent.match(/IEMobile/i);
  },
  any: function() {
  return (isMobile.Android()
    || isMobile.BlackBerry()
    || isMobile.iOS()
    || isMobile.Opera()
    || isMobile.Windows());
  }
};

if (isMobile.Android()) {
  document.location.href = "y";
} else if(isMobile.iOS()) {
  document.location.href = "x";
}
```

---

jQuery UI Datepicker Reset Date

```
$(document).ready(
  function () {
    $(.datepicker).datepicker({
    showOn: 'focus', showButtonPanel: true,
    closeText: 'Clear', // Text to show for close button
    onClose: function () {
      var event = arguments.callee.caller.caller.arguments[0]; // If Clear gets clicked, then re
      if ($(event.delegateTarget).hasClass('ui-datepicker-close')) {
        $(this).val('');
      }
    }
  });
});
```

Convert object to string

Use JSON.stringify() function:

```
var obj = new Date();
console.log(JSON.stringify(obj));
```

DataTables Editor jQuery UI Datepicker Issue

- When 2 datepickers are used, it will jump to first datepicker when the second datepicker's date is selected. Make sure that datepickers aren't used as the first field.

Problem: DataTable - Cannot read property 'style' of undefined jquery.dataTables.js

Make sure that HTML number table columns matches the number of 'mData' definition during data table initialization

PHP, Javascript & HTML

How to do Everything in Metro UI CSS

Metro UI CSS with isLoading jQuery plugin

IsLoading jQuery plugin:

```
/* jQuery Plugin */
$.isLoading({
  class: 'spin',
  text: 'Loading',
  position: 'overlay',
  tpl: '<span class="bg-darker isloading-wrapper %wrapper%">%text%<i class="%class% icon-spin"><
});

/* CSS */
/* Chrome, Safari, Opera */
@-webkit-keyframes rotate {
  from {-webkit-transform: rotate(0deg);}
  to {-webkit-transform: rotate(360deg);}
}

/* Mozilla */
@-moz-keyframes rotate {
  from {-moz-transform: rotate(0deg);}
  to {-moz-transform: rotate(360deg);}
}

/* Opera */
@-moz-keyframes rotate {
  from {-o-transform: rotate(0deg);}
  to {-o-transform: rotate(360deg);}
}

/* Spin animation */
i.spin, span.spin {
  -webkit-animation: rotate 1s ease-in-out infinite;
  animation: rotate 1s ease-in-out infinite;
  -moz-animation: rotate 1s ease-in-out infinite;
}

/* jQuery isLoading styles */
```

```
span.isloading-wrapper.isloading-overlay {
  position: absolute;
  top: 50%;
  left: 50%;
  padding: 10px;
}
```

Metro UI CSS Datepicker Change Event

```
$("#datepicker").datepicker({
  selected: function(dateString, dateObject) {
    alert('date-selected');
  }
});
```

ReactJS REDUX Summary

Using stores and reducers

A simple example

```
import {createStore} from 'redux'

var reducer_1 = (state, action) => {
  console.log('reducer_0 was called with state', state, 'and action', action);
};
var store_1 = createStore(reducer_1);
```

Real world example

```
import {createStore} from 'redux'

var reducer_1 = (state = {}, action) => {
  switch(action.type) {
  case '':
  return {
    ...state,
    message: action.value
  };
  default:
    return state;
  }
};
var store_1 = createStore(reducer_1);
```

Combining reducers

```
import {combineReducers, createStore} from 'react';

var userReducer = (state = {}, action) => {
  switch(action) {
    case 'ADD_USER':
    return {
      //Return modified state
  };
  default:
    return state;
```

```
  }
};

var itemReducer = (state = [], action) => {
  switch(action) {
    case 'ADD_ITEM':
      return {
        //Return modified state
      };
    default:
      return state;
  }
};

var reducers = combineReducers(user: userReducer, item: itemReducer);
var store = createStore(reducers);
console.log(store.getState());
// {
// user: {}, // {} is the slice returned by our userReducer
// items: [] // [] is the slice returned by our itemsReducer
// }
```

Dispatching an action

Flow of application: ActionCreator -> Action -> dispatcher -> reducer

Without action creator

```
import {combineReducers, createStore} from 'react';

var userReducer = (state = {}, action) => {
  switch(action) {
  case 'ADD_USER':
    return {
      //Return modified state
    };
    default:
      return state;
  }
};

var itemReducer = (state = [], action) => {
  switch(action) {
    case 'ADD_ITEM':
      return {
        //Return modified state
      };
      default:
        return state;
    }
```

```
};

var reducers = combineReducers(user: userReducer, item: itemReducer);
var store = createStore(reducers);
store.dispatch({type: 'ACTION'});
```

With Action Creator (Adopted from Flux)

```
import {combineReducers, createStore} from 'react';

var userReducer = (state = {}, action) => {
  switch(action) {
    case 'ADD_USER':
      return {
        //Return modified state
      };
    default:
      return state;
  }
};

var itemReducer = (state = [], action) => {
  switch(action) {
    case 'ADD_ITEM':
      return {
        //Return modified state
      };
    default:
      return state;
  }
};

var reducers = combineReducers(user: userReducer, item: itemReducer);
var store = createStore(reducers);
var addItemActionCreator = (name) => {
  return {
    item: name,
    type: 'ADD_ITEM'
  };
};
store.dispatch(addItemActionCreator);
```

Async Action with Middleware

```
import {combineReducers, createStore, applyMiddleware} from 'react';

var userReducer = (state = {}, action) => {
  switch(action) {
    case 'ADD_USER':
      return {
        //Return modified state
```

```
      };
    default:
      return state;
  }
};

var itemReducer = (state = [], action) => {
  switch(action) {
    case 'ADD_ITEM':
      return {
        //Return modified state
      };
    default:
      return state;
    }
};

//Set the state after 2s
var addItemActionCreator = (name) => {
  return (dispatch) => {
    setTimeout(() => {
      dispatch({
        item: name,
        type: 'ADD_ITEM'
      });
    }, 2000);
  };
};

// 1) The first level provide the dispatch function and a getState function (if your
// middleware or your action creator needs to read data from state) to the 2 other levels
// 2) The second level provide the next function that will allow you to explicitly hand over
// your transformed input to the next middleware or to Redux (so that Redux can finally call all
// 3) the third level provides the action received from the previous middleware or from your dis
// and can either trigger the next middleware (to let the action continue to flow) or process
// the action in any appropriate way.
var thunkMiddleware = ({dispatch, getState}) => {
  return (next) => {
    return (action) => {
      return typeof action === 'function' ? action(dispatch, getState) : next(action);
    }
  };
};

var reducers = combineReducers(user: userReducer, item: itemReducer);
var finalCreateStore = applyMiddleware(thunkMiddleware)(createStore);
var store = finalCreateStore(reducers);
store.dispatch(addItemActionCreator);
```

Subscribing to a Store

```
import {createStore} from 'redux'

var reducer_1 = (state = {}, action) => {
  switch(action.type) {
    case '':
      return {
        ...state,
        message: action.value
      };
    default:
      return state;
  }
};
var store_1 = createStore(reducer_1);
store_1.subscribe(() => {
  //Update react views
});
```

Standard ReactJS Procedures

Initialize ReactJS Project

1. Enter npm init
2. Enter npm install --save-dev babel-core babel-loader babel-preset-react babel-preset-es2015 webpack webpack-dev-server
3. Enter npm install --save react react-dom jquery
4. Append "scripts": {"start": "node_modules/.bin/webpack-dev-server --progress"},
5. Create and edit webpack.config.js:
    1. ```
       module.exports = {
         entry: [
           './src/app.js'
         ],
         output: {
           path: __dirname,
           filename: "bundle.js"
         },
         module: {
           loaders: [{
             test: /\.jsx?$/,
             loader: 'babel',
             exclude: 'node_modules',
             query: {
               presets:['react', 'es2015']
             }
           }]
         }
       };
       ```
6. Enter `npm start`
       ```

Git and SVN

All about Git and SVN

Patching with Git

Creating a Patch

Patch for Working Copy (Not Committed)

`git diff > patch.diff`

Patch from one commit to another

`git diff <from-commit-hash> <to-commit-hash> > patch.diff`

Patching a Project Generated from `git diff`

Applying a Patch: `git apply patch.diff`

Solution for Whitespace Errors

`git apply --ignore-space-change --ignore-whitespace patch.diff`

Make sure that you removed whitespaces from both patch file and from the files you are going to patch.

References:

1. Whitespace errors
2. Applying git patch

Solving Cryptic SVN Errors

SVN File already exists error

```
svn update path/ --accept=mine-full
```

CSS

Everything about CSS

CSS

CSS Utilities

Media Queries min-width and max-width

@media only screen and (min-width: 330px) {...}:

- If [device width] is greater than or equal to 330px, then do {...}

@media only screen and (max-width: 330px) {...}:

- If [device width] is less than or equal to 330px, then do {...}

Infinite rotate animation CSS3

```
/* Chrome, Safari, Opera */
@-webkit-keyframes rotate {
  from {-webkit-transform: rotate(0deg);}
  to {-webkit-transform: rotate(360deg);}
}

/* Standard syntax */
@keyframes rotate {
  from {transform: rotate(0deg);}
  to {transform: rotate(360deg);}
}

/* Mozilla */
@-moz-keyframes rotate {
  from {-moz-transform: rotate(0deg);}
  to {-moz-transform: rotate(360deg);}
}

/* Elements to rotate */
.rotate {
  -webkit-animation: rotate 1s ease-in-out infinite;
  -moz-animation: rotate 1s ease-in-out infinite;
  animation: rotate 1s ease-in-out infinite;
}
```

Everything About APIs

Solutions and tricks for all types of APIs including Facebook and Google

Facebook APIs

Generating Graph Access Token that Never Expire

1. Make sure you are the admin of the FB page you wish to pull info from
2. Create a FB App using the Page admin's account
3. Head over to the Facebook Graph API Explorer
4. Select the FB App you created from the "Application" drop down list
5. Click "Get User Access Token"
6. Make sure that "manage_pages" permission is checked
7. Make a GET request to: https://graph.facebook.com/oauth/access_token?client_id=<App ID>&client_secret=<App secret>&grant_type=fb_exchange_token&fb_exchange_token=<short-lived access token>
8. Copy the new long-lived token from the response
9. Make another get request to: https://graph.facebook.com/me/accounts?access_token=<your long-lived access token>
10. Copy the new token and paste it to Access Token Debug Tool, make sure that it never expires

References

- This stackoverflow post

Common Parse API Operations

Get schema

```
curl -X GET \
  -H "X-Parse-Application-Id: ${APPLICATION_ID}" \
  -H "X-Parse-Master-Key: ${MASTER_KEY}" \
  -H "Content-Type: application/json" \
  https://wiki.twcloud.tech:1337/parse/schemas
```

Drop schema

```
curl -X DELETE\
  -H "X-Parse-Application-Id: ${APPLICATION_ID}" \
  -H "X-Parse-Master-Key: ${MASTER_KEY}" \
  -H "Content-Type: application/json" \
  https://api.parse.com/1/schemas/${SCHEMA}
```

CMS/E-Commerce

All content management system and E-Commerce platform related stuff such as Joomla!, Drupal and Wordpress

CMS/E-Commerce

How to do Everything in Joomla!

Changing Read More Text

1. In your Joomla admin go to `Extensions > Language Manager > Overrides > New`.
2. In the Language Constant inputbox put: `READ_MORE`
3. Then place your desired text in the Text box. Then select Save and Close.

Reference: http://www.rockettheme.com/forum/joomla-extension-roksprocket/207826-solved-changing-load-more-in-mosaic?start=0#1019038

Joomla "Fatal error Call to a member function isEnabled() on a non-object"

After moving to a new site, login to admin panel and clear the cache

Magento Development

Magento Controllers

Code Pools

Magento code pools are stored in app/code/ directory, it consists of:

- core: All the core Magento modules, DO NOT edit core code pools directly as it may break Magento installation due to incompatibilities etc.
- community: Modules by third-party codes, e.g. extensions
- local: Custom made modules, copy core modules here (preserving directory structure) if to modify the core modules instead of modifying core modules directly

Code pool execution priority (lowest to highest, find in next if module not found):

1. local
2. community
3. core
4. `/app/lib`

Namespaces (A.K.A Packages)

Fresh Magento core modules are stored app/code/core directory, "Mage" and "Zend" are 2 namespaces created by Magento. Creating namespaces for your custom modules are just to create a folder in app/code/local directory, it can be any name, e.g. Practice

Naming Conventions

- DO NOT include any "_" (underscore) for folder and file names, as it will be replaced by directory separator (DS) in Magento's autoloader
- Initial caps and camelcase for naming folders and classes
- Use "_" (underscore) in class names to specify path to class files e.g:
  - Magento expect "class Mage_Catalog_Block_Product_Widget_New" in class declaration to find the class file in Mage/Catalog/Block/Product/Widget/New.php

Magento's Autoloaders Class Initialization Steps:

1. "_" (underscores) are replaced by spaces
2. Convert all words to initial caps
3. Spaces are replaced by DS (directory separator)
4. Append .php

e.g. $instance = new Practive_ControllerTest_Model_MyClass() will be converted to Practice/ControllerTest/Model/MyClass.php where the class file is expected

Module Folder Structure

Directory structure for module "ControllerTest" (app/code/local/Practice/ControllerTest/) should contain the following directories:

- Block/
- controllers/
  - Optional as not all modules contain controllers
- etc/
  - Store module configuration and system files (expects .xml extension)
- Helper/
- Model/
- sql/

Configuration Files (Module Configuration)

- File can be any names as long as it end with xml
- Recommended naming conventions: Namespace_ModuleName.xml
- System wide configuration stored in app/etc/modules/ directory

Configuration Steps:

1. Create and edit app/etc/modules/Practice_ControllerTest.xml where Magento expects to find the module (ControllerTest) main config file in `app/local/Practice/ControllerTest/etc/config.xml`:

```
<?xml version="1.0"?>
<config>
```

```
    <modules>
        <Practice_ControllerTest>
           <active>true</active>
           <codepool>local</codepool>
        </Practice_ControllerTest>
    </modules>
</config>
```

2. Create and edit app/code/local/Practice/ControllerTest/etc/config.xml:

```
<?xml version="1.0"?>
<config>
    <modules>
            <Practice_ControllerTest>
                 <version>0.0.1</version>
        </Practice_ControllerTest>
    </modules>
</config>
```

3. To define a controller, add the following code between "" tags after "" tags in `app/code/local/Practice/ControllerTest/etc/config.xml`:

```
<frontend>
    <routers>
        <test_controller>
            <module>Practice_ControllerTest</module>
            <frontName>requestflowtest</frontName>
        </test_controller>
    </routers>
</frontend>
```

- frontend => Specify an "area", possible values: frontend, backend or global
- routers => A role
- test_controller => Unique controller config

Creating a Controller

Create and edit `app/code/local/Practice/ControllerTest/controllers/IndexController.php`

```
class Practice_ControllerTest_IndexController extends
Mage_Core_Controller_Front_Action {
```

```
    public function indexAction() { echo "Hello World!"; }
}
```

Testing

Navigate to "//requestflowtest"

Routing

- To specify as admin router, use "admin" between the "" tags in config.xml, causing it to route for admin interface only
- To specify as frontend router, use "frontend" between "" tags in config.xml, causing it to route for frontend interface only

Order:

1. Admin
2. Standard
3. Cms
4. Default

Sample URL: http:////catalog/product/view/id/1

- catalog: frontName tag in config.xml
- product: ProductController
- catalog/product results: Mage/Catalog/controllers/ProductController.php
- view
  - viewAction() method in the controller
  - If viewAction() method not found, assume indexAction() in IndexController

---

Magento View

Layouts

Magento uses xml files to specify the layout, available handles for

.xml includes: 1. label *Specify the title or label to be displayed to user 2. reference* Add an existing block to the page *Attribute "name" specifies the existing block name* action *block 3. remove* Remove an existing block from the layout *Attribute "name" specifies the block to remove*

e.g. 4. update *Load specified layout handle into the current layout handle, provides a kind of inheritance to the current layout handles* Can be use to paste an entire layout handle into the current ones *Attribute "handle" must be included which specifies the layout handle* Syntax: 5. block *type/class* Mandatory, sets an identifier of a PHP class name *name* Specifies the name of the block, can be reused for the reference directives *template* Specifies which .phtml file to render the page *as* Used to assign an alias for the block *If specified, this block can be used in the $this->getChildHtml('alias') method* output *Specifies the method name the system will use to gather an output instead of the default toHtml() method* Only required for the root block hierarchy because child blocks can be obtained through the getChildHtml() method *parent* Override the default parent block *before* Used to specify that the current block is before the specified block *after* *Used to specify that the current block is after the specified block 6. action* method *Specifies the method name that should be called* block *ifconfig* Allows checking of the configuration parameter before the action processor *json* Specifies that the data has to be decoded before passing into the methods *translate* Specifies which identifier in the child node must be passed for processing for the translation procedure *module* Specifies action module *helper* Specifies the helper functions, allow processing of data
Magento stores all .xml files in app/design/[areaname]/[packagename]/[themename]/layout folder ### Block Blocks are just a set of PHP classes used in rendering. Block types:
Mage_Core_Block_Template (core/template) Render template block defined by its template attribute *Mage_Core_Block_Text_List (core/text_list)* All child blocks are rendered automatically without the need to call GetChildHtml() method *Mage_Core_Block_Messages (core/messages)* Renders success or notice messages *Mage_Core_Block_Template_Links (page/template_links)* Used to create a list of links *Mage_Core_Block_Switch (page/switch)* Render a store switcher ---
Retrieve Categories from Other Stores `**Using App Emulation**`

```
    $appEmulation = Mage::getSingleton('core/app_emulation');
    $store_id = 1; // The ID if your store
    $initialEnvironmentInfo = $appEmulation->startEnvironmentEmulation($store_id);

    $children = Mage::getModel('catalog/category')->load(306)->getChildrenCategories();
    foreach($children as $child){
       echo $child->getName() . " " . $child->getUrl() . "<br/>";
    }

    $appEmulation->stopEnvironmentEmulation($initialEnvironmentInfo);
```

--- ## Magento New Product Widget by Store 1. Create directory structure: `mkdir -p app/code/local/Mage/Catalog/Block/Product/Widget` 2. Copy `app/code/core/Mage/Catalog/Block/Product/Widget/New.php` to `app/code/local/Mage/Catalog/Block/Product/Widget`: `cp app/code/core/Mage/Catalog/Block/Product/Widget/New.php to app/code/local/Mage/Catalog/Block/Product/Widget` 3. Modify `New.php` `_getRecentlyAddedProductsCollection()` function and add the following lines: `$_rootcatID = Mage::app()->getStore()->getRootCategoryId();`

```
    $collection = $this->_addProductAttributesAndPrices($collection)
       ->joinField('category_id', 'catalog/category_product', 'category_id',
'product_id=entity_id', null, 'left')
       ->addAttributeToFilter('category_id', array('in' => $_rootcatID))
       ->addAttributeToSelect('*')
```

--- ## Add new field in magento(1.9) customer registration You need to create a new extension to make it clean.

Let's call the extension `StackExchange_Customer`.
You will need the following files: `app/etc/modules/StackExchange_Customer.xml` - the declaration file <?xml version="1.0"?> truelocal`app/code/local/StackExchange/Customer/etc/config.xml` - the configuration file: <?xml version="1.0"?> 1.0.0StackExchange_Customer_Helper StackExchange_CustomerMage_Customer_Model_Resource_Setupstackexchange_customer.xml StackExchange_Customer.csv `app/code/local/StackExchange/Customer/sql/stackexchange_customer_setup/install-1.0.0.php` - the install file. Will add the new attribute. <?php $this->addAttribute('customer', 'license_number', array('type' => 'varchar', 'label' => 'License Number', 'input' => 'text', 'position' => 120, 'required' => false,//or true 'is_system' => 0,)); $attribute = Mage::getSingleton('eav/config')->getAttribute('customer', 'license_number'); $attribute->setData('used_in_forms', array('adminhtml_customer', 'checkout_register', 'customer_account_create', 'customer_account_edit',)); $attribute->setData('is_user_defined', 0); $attribute->save(); `app/code/local/StackExchange/Customer/Helper/Data.php` - the module main helper <?php class StackExchange_Customer_Helper_Data extends Mage_Core_Helper_Abstract { } This will add your attribute for the customer.
It should work nicely on the backend.
Unfortunately you have to edit the frontend templates manually now because Magento does not have any event or empty block where you can put your fields.
For this you need the following. `app/design/frontend/base/default/layout/stackexchange_customer.xml` <?xml version="1.0"?> stackexchange_customer/form/edit.phtml stackexchange_customer/register.phtml

And now the templates.

`app/design/frontend/base/default/template/stackexchange_customer/register.phtml` - the registration template.
For this one make a clone of the `/app/design/frontend/{package}/{theme}/template/persistent/customer/form/register.phtml` and just insert this somewhere inside the form. I don't need to post the full file here. Arrange it as you please

```
<li>
    <label for="license_number"><?php echo $this->__('License Number') ?></label>
    <div class="input-box">
        <input type="text" name="license_number" id="license_number" value="<?php echo $this->escapeHtml($this->getFormData()->getLicenseNumber()) ?>" title="<?php echo $this->__('License Number') ?>" class="input-text" />
    </div>
</li>
```

`/app/design/frontend/base/default/template/stackexchange_customer/form/edit.phtml` For this one clone `/app/design/frontend/{package}/{theme}/template/customer/form/edit.phtml` and insert somewhere inside the form this:

```
<li>
    <label for="license_number"><?php echo $this->__('License Number') ?></label>
    <div class="input-box">
        <input type="text" name="license_number" id="license_number" value="<?php echo $this->htmlEscape($this->getCustomer()->getLicenseNumber()) ?>" title="<?php echo $this->__('License Number') ?>" class="input-text" />
    </div>
</li>
```

You can also create the translation file. Is not mandatory but it's nice to have `app/locale/en_US/StackExchange_Customer.csv` `"License Number","License Number"`

Clear the cache and you should be set.

How to delete an eav product attribute programmatically

It is my view that this sort of thing should be done via set-up script or if that is not possible then via the admin section. I cannot think of a situation where you need to write a script for this, but maybe there is one.

For a set-up script it is incredibly simple:

```
$installer = $this;
$installer->startSetup();
// Remove Product Attribute
$installer->removeAttribute('catalog_product', 'product_attribute_code');
// Remove Customer Attribute
$installer->removeAttribute('customer', 'customer_attribute_code');
$installer->endSetup();
```

If you need something more complex like removing groups or attribute sets that can also be done via a script too.

Fix Magento Error "dbModel read resource does not..."

We came across the following magento error this morning whilst conducting a database rollback for a client of ours. "dbModel read resource does not implement Zend_Db_Adapter_Abstract"

The solution to solving this is actually quite simple, clear the folders/files within the magento_root /var/cache/ folder.

Error 503 Admin and Frontend

Login to webroot and remove the file maintenance.flag

Display Stock Level on Product Details Page

Edit `app/design/frontend/<theme>/<template>/template/catalog/product/view/type/availability/default.phtml`:

```
<span class="value">
<?php
$quantity = Mage::getModel('cataloginventory/stock_item')->loadByProduct($_pro
duct)->getQty();
if ($quantity > 0)
     echo intVal($quantity);
else
     echo $this->helper('catalog')->__('In stock');
?>
</span>
```

##Adding Module Dependencies

In `app/etc/modules/Practice_OneStepCheckout.xml` add a depends node under `Practice_OneStepCheckout` tags:

```
<Practice_OneStepCheckout>
    <depends>
        <Mage_Sales />
        <Mage_CatalogInventory />
        <Mage_Checkout />
    </depends>
</Practice_OneStepCheckout>
```

Adding Custom Scripts to Magento

You can set different scripts per website or even store views:

Use the field Miscellaneous Scripts from System->Configuration->Design->Head and put your scripts in there. They will be added before the tag and you can set different scripts per website or even store views.

Adding Static Block to .phtml

`echo $this->getLayout()->createBlock('cms/block')->setBlockId('static_block_identifier')->toHtml();`

Where `static_block_identifier` is the identifier for the CMS static block "identifier"

CMS/E-Commerce

Clearing Magento 2.x Orders Script

```
SET FOREIGN_KEY_CHECKS=0;

# Clean order history
TRUNCATE TABLE `sales_bestsellers_aggregated_daily`;
TRUNCATE TABLE `sales_bestsellers_aggregated_monthly`;
TRUNCATE TABLE `sales_bestsellers_aggregated_yearly`;

# Clean order infos
TRUNCATE TABLE `sales_creditmemo`;
TRUNCATE TABLE `sales_creditmemo_comment`;
TRUNCATE TABLE `sales_creditmemo_grid`;
TRUNCATE TABLE `sales_creditmemo_item`;
TRUNCATE TABLE `sales_invoice`;
TRUNCATE TABLE `sales_invoiced_aggregated`;
TRUNCATE TABLE `sales_invoiced_aggregated_order`;
TRUNCATE TABLE `sales_invoice_comment`;
TRUNCATE TABLE `sales_invoice_grid`;
TRUNCATE TABLE `sales_invoice_item`;
TRUNCATE TABLE `sales_order`;
TRUNCATE TABLE `sales_order_address`;
TRUNCATE TABLE `sales_order_aggregated_created`;
TRUNCATE TABLE `sales_order_aggregated_updated`;
TRUNCATE TABLE `sales_order_grid`;
TRUNCATE TABLE `sales_order_item`;
TRUNCATE TABLE `sales_order_payment`;
TRUNCATE TABLE `sales_order_status_history`;
TRUNCATE TABLE `sales_order_tax`;
TRUNCATE TABLE `sales_order_tax_item`;
TRUNCATE TABLE `sales_payment_transaction`;
TRUNCATE TABLE `sales_refunded_aggregated`;
```

```
TRUNCATE TABLE `sales_refunded_aggregated_order`;
TRUNCATE TABLE `sales_shipment`;
TRUNCATE TABLE `sales_shipment_comment`;
TRUNCATE TABLE `sales_shipment_grid`;
TRUNCATE TABLE `sales_shipment_item`;
TRUNCATE TABLE `sales_shipment_track`;
TRUNCATE TABLE `sales_shipping_aggregated`;
TRUNCATE TABLE `sales_shipping_aggregated_order`;

# Clean cart infos
TRUNCATE TABLE `quote`;
TRUNCATE TABLE `quote_address`;
TRUNCATE TABLE `quote_address_item`;
TRUNCATE TABLE `quote_id_mask`;
TRUNCATE TABLE `quote_item`;
TRUNCATE TABLE `quote_item_option`;
TRUNCATE TABLE `quote_payment`;
TRUNCATE TABLE `quote_shipping_rate`;

# Reset indexes (if you want your orders number start back to 1
TRUNCATE TABLE sequence_invoice_1;
TRUNCATE TABLE sequence_order_1;
TRUNCATE TABLE sequence_shipment_1;
TRUNCATE TABLE sequence_creditmemo_1;



SET FOREIGN_KEY_CHECKS=1;
```

CMS/E-Commerce

Refreshing Magento 2.x Static Contents

1. Remove `static` and `var` contents `rm -rf pub/static/* var/cache/* var/composer_home/* var/generation/* var/page_cache/* var/view_preprocessed/*`
2. Regenerate static files `php bin/magento setup:static-content:deploy`