

# PHP, Javascript & HTML

All about PHP, JS and HTML including its frameworks

- [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](#)

# Useful PHP Codes

## Increment a Date by Month, Day or Year

### Increment by month

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

### Increment by day

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

### Increment by year

```
1. $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, -106.346770999999, 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.*"
```

# 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.

# 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

# 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 dispatch
  // 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`