

THE JQUERY DEFERRED OBJECT

\$.Deferred

deferred.promise()

deferred.resolve()

deferred.reject()

deferred.done()

deferred.fail()

deferred.then()

deferred.always()

\$.when()

`$.Deferred` creates a new Deferred object

`deferred.promise()` creates promises to tell callbacks when action is done

`deferred.resolve()` notifies callbacks the action was successful

`deferred.reject()` notifies callbacks the action was unsuccessful

`deferred.done()` attaches callbacks for when the action is successful

`deferred.fail()` attaches callbacks for when the action is unsuccessful

`deferred.then()` attaches one successful and one unsuccessful callback

`deferred.always()` attaches callbacks that execute after success or failure

`$.when()` to wait on more than one promise to finish

```
deferred.reject({  
  errorCode : 401,  
  errorMessage : "File not found"  
});
```

```
deferred.resolve({  
  result : "some text",  
  code : 200  
});
```

```
doneCallback = function ( deferredResult ) {  
  alert( deferredResult.result ); // alerts "some text"  
};
```

```
failCallback = function ( deferredResult ) {  
  alert( deferredResult.errorMessage ); // alerts "File not found"  
};
```

```
failCallback2 = function ( deferredResult ) {  
  alert( deferredResult.errorCode ); // alerts "401"  
};
```


Asynchronous actions before

```
var performAsyncStuff;
```

```
performAsyncStuff = function ( ) {  
    $.ajax({  
        url : 'http://whateverurl.com/resource',  
        dataType : 'json',  
        error : function ( jqXHR, textStatus, errorThrown ) {  
            alert('An error ocurred');  
        },  
        success : function ( json ) {  
            $('#element1').html(json.element1);  
            $('#element2').html(json.element2);  
        }  
    });  
};
```

But what if you have something that depends on two or more asynchronous actions?


```
var performAsyncStuff2, price1, price2, totalPrice;
```

```
performAsyncStuff2 = function ( ) {  
    $.ajax({  
        url : 'http://whateverurl.com/resource',  
        dataType : 'json',  
        error : function ( jqXHR, textStatus, errorThrown ) { alert('An error occurred'); },  
        success : function ( json ) {  
            price1 = json.price;  
            $.ajax({  
                url : 'http://whateverurl.com/secondresource',  
                dataType : 'json',  
                error : function ( jqXHR, textStatus, errorThrown ) { alert('An error occurred'); },  
                success : function ( json2 ) {  
                    price2 = json2.price;  
                    totalPrice = price1 + price2;  
                }  
            });  
        });  
    }  
});  
};
```


Simplifying code flow

```
var firstAjax, secondAjax, totalPrice;
```

```
firstAjax = function ( ) {  
    var dfd = new $.Deferred();  
    $.ajax({  
        url : 'http://whateverurl.com/resource',  
        dataType : 'json',  
        error : function ( jqXHR, textStatus, errorThrown ) { dfd.reject( {jqXHR : jqXHR, textStatus : textStatus, errorThrown : errorThrown} ); },  
        success : function ( json ) { dfd.resolve(json); }  
    });  
    return dfd.promise(); // MUST return the promise  
};
```

```
secondAjax = function ( ) {  
    var dfd = new $.Deferred()  
    $.ajax({  
        url : 'http://whateverurl.com/secondresource',  
        dataType : 'json',  
        error : function ( jqXHR, textStatus, errorThrown ) { dfd.reject(); },  
        success : function ( json2 ) {  
            dfd.resolve(json2);  
        }  
    });  
    return dfd.promise()  
};
```

```
$.when( firstAjax(), secondAjax() )  
    .done( function ( firstResult, secondResult ) {  
        totalPrice = firstResult.price + secondResult.price;  
    })  
    .fail( function () {  
        alert('it failed');  
    })  
    .always( function () {  
        alert('this gets executed after fail() or after done() get executed');  
    });
```


Don't need to do it all at once


```
var ajaxCall, savePromise;
```

```
ajaxCall = function ( ) {  
    var dfd = new $.Deferred();  
    $.ajax({  
        url : 'http://whateverurl.com/resource',  
        dataType : 'json',  
        error : function ( jqXHR, textStatus, errorThrown ) {  
            dfd.reject( {jqXHR : jqXHR, textStatus : textStatus, errorThrown : errorThrown} );  
        },  
        success : function ( json ) {  
            dfd.resolve(json);  
        }  
    });  
    return dfd.promise();  
};
```

```
savePromise = ajaxCall(); // savePromise = dfd.promise();  
savePromise.then( doneFirst, failFirst );  
dosomething();  
dosomethingelse();  
savePromise.done( doneSecond ); // only gets executed once ajaxCall returns (in the order added)  
savePromise.done( doneThird ); // if ajaxCall has already returned they get executed immediately
```

`$.ajax()`

also `$.get()`, `$.post()`, `$.getJSON()`...

```
var ajaxPromise, doneCallback1, doneCallback2, failCallBack;
```

```
doneCallback1 = function ( json ) {  
    // do something with the returned json here  
};
```

```
doneCallback2 = function ( json ) {  
    // do something else with the returned json  
};
```

```
failCallback = function ( jqXHR, textStatus, errorThrown ) {  
    // figure out why it failed  
};
```

```
ajaxPromise = $.ajax({  
    url : 'http://whateverurl.com/resource',  
    dataType : 'json'  
});
```

```
ajaxPromise  
    .done( doneCallback1, doneCallback2 )  
    .fail( failCallback );
```


More Reading

<http://api.jquery.com/category/deferred-object/>

<http://msdn.microsoft.com/en-us/scriptjunkie/gg723713>

<http://www.erichynds.com/jquery/using-deferreds-in-jquery/>

<http://intridea.com/2011/2/8/fun-with-jquery-deferred?blog=company>