Encapsulating jQuery within a JavaScript application

Every time I write some new code for a new site (especially in WordPress) I have to deal with the main problem of conflicting scripts. jQuery makes no exception. The solution is to encapsulate jQuery in the core of your JavaScript application. This doesn't imply that you have to extend your main object with the jQuery's `fn` object. Simply put, you have to create a wrapper around the main `jQuery` object. This is much simpler to achieve and also accessible to non-veterans jQuery developers. Let's see the details.

Here's a simple implementation:

```
(function($) {

        var APP = function() {

                this.core = jQuery; // public. Also
jQuery.noConflict()

                var _core = this.core; // private

                // implementation here
        };

})(jQuery);
```

jQuery becomes a member of our application, both for internal and external use. The self-executing function around our application prevents jQuery from conflicting with other libraries and adds an extra layer of isolation to our application.

Here's an example:

```
var app = new APP();

app.core(function() {

        alert(app.core.fn.jquery); // 1.7.2

});
```

As you can see, all the functionalities of jQuery remain intact. What's more, they can be easily decoupled from our application. This would not be possible with the other approach, that is, extending our object with jQuery.