* Jump To … +  
    
  [<<< back to documentation](http://docs.google.com/index.html)  [base.js](http://docs.google.com/base.html)   [constraint.js](http://docs.google.com/constraint.html)   [defaults.js](http://docs.google.com/defaults.html)   [factory.js](http://docs.google.com/factory.html)   [field.js](http://docs.google.com/field.html)   [form.js](http://docs.google.com/form.html)   [main.js](http://docs.google.com/main.html)   [multiple.js](http://docs.google.com/multiple.html)   [pubsub.js](http://docs.google.com/pubsub.html)   [remote.js](http://docs.google.com/remote.html)   [ui.js](http://docs.google.com/ui.html)   [utils.js](http://docs.google.com/utils.html)   [validator.js](http://docs.google.com/validator.html)   [validator\_registry.js](http://docs.google.com/validator_registry.html)

# pubsub.js

* [¶](#gjdgxs)  
  import $ from 'jquery';  
  import Field from './field';  
  import Form from './form';  
  import Utils from './utils';  
    
  var o = $({});  
  var deprecated = function () {  
   Utils.warnOnce("Parsley's pubsub module is deprecated; use the 'on' and 'off' methods on parsley instances or window.Parsley");  
  };
* [¶](#30j0zll)  
  Returns an event handler that calls fn with the arguments it expects  
  function adapt(fn, context) {
* [¶](#1fob9te)  
  Store to allow unbinding  
   if (!fn.parsleyAdaptedCallback) {  
   fn.parsleyAdaptedCallback = function () {  
   var args = Array.prototype.slice.call(arguments, 0);  
   args.unshift(this);  
   fn.apply(context || o, args);  
   };  
   }  
   return fn.parsleyAdaptedCallback;  
  }  
    
  var eventPrefix = 'parsley:';
* [¶](#3znysh7)  
  Converts ‘parsley:form:validate’ into ‘form:validate’  
  function eventName(name) {  
   if (name.lastIndexOf(eventPrefix, 0) === 0)  
   return name.substr(eventPrefix.length);  
   return name;  
  }
* [¶](#2et92p0)  
  $.listen is deprecated. Use Parsley.on instead.  
  $.listen = function (name, callback) {  
   var context;  
   deprecated();  
   if ('object' === typeof arguments[1] && 'function' === typeof arguments[2]) {  
   context = arguments[1];  
   callback = arguments[2];  
   }  
    
   if ('function' !== typeof callback)  
   throw new Error('Wrong parameters');  
    
   window.Parsley.on(eventName(name), adapt(callback, context));  
  };  
    
  $.listenTo = function (instance, name, fn) {  
   deprecated();  
   if (!(instance instanceof Field) && !(instance instanceof Form))  
   throw new Error('Must give Parsley instance');  
    
   if ('string' !== typeof name || 'function' !== typeof fn)  
   throw new Error('Wrong parameters');  
    
   instance.on(eventName(name), adapt(fn));  
  };  
    
  $.unsubscribe = function (name, fn) {  
   deprecated();  
   if ('string' !== typeof name || 'function' !== typeof fn)  
   throw new Error('Wrong arguments');  
   window.Parsley.off(eventName(name), fn.parsleyAdaptedCallback);  
  };  
    
  $.unsubscribeTo = function (instance, name) {  
   deprecated();  
   if (!(instance instanceof Field) && !(instance instanceof Form))  
   throw new Error('Must give Parsley instance');  
   instance.off(eventName(name));  
  };  
    
  $.unsubscribeAll = function (name) {  
   deprecated();  
   window.Parsley.off(eventName(name));  
   $('form,input,textarea,select').each(function () {  
   var instance = $(this).data('Parsley');  
   if (instance) {  
   instance.off(eventName(name));  
   }  
   });  
  };
* [¶](#tyjcwt)  
  $.emit is deprecated. Use jQuery events instead.  
  $.emit = function (name, instance) {  
   deprecated();  
   var instanceGiven = (instance instanceof Field) || (instance instanceof Form);  
   var args = Array.prototype.slice.call(arguments, instanceGiven ? 2 : 1);  
   args.unshift(eventName(name));  
   if (!instanceGiven) {  
   instance = window.Parsley;  
   }  
   instance.trigger(...args);  
  };  
    
  export default {};