

# Logging in Cordova Applications

fork and edit tutorial (<https://github.ibm.com/MFPSamples/DevCenter/tree/master/tutorials/en/foundation/8.0/analytics/remote-controlled-client-side-log-collection/cordova.md>) | report issue (<https://github.ibm.com/MFPSamples/DevCenter/issues/new>)

## Overview

This tutorial provides the required code snippets in order to add logging capabilities in Cordova applications.

## Logging example

Outputs to a browser JavaScript console, LogCat, or Xcode console.

```
var MathUtils = function(){  
  var logger = WL.Logger.create({pkg: 'MathUtils'});  
  var sum = function(a, b){  
    var sum = a + b;  
    logger.debug('sum called with args ' + a + ' and ' + b + '. Returning ' + sum);  
    return sum;  
  };  
}();
```

## Additional API Methods For Specific Tasks

Log capture is enabled by default. To turn log capture on or off:

```
WL.Logger.config({capture: false})
```

The default capture level is DEBUG in development and FATAL in production. To control the capture level (verbosity):

```
WL.Logger.config({level: 'FATAL'})
```

Log sending is enabled by default. To turn automatic log sending on or off:

```
WL.Logger.config({autoSendLogs: false})
```

For more information about the `Logger` API, see the API reference in the user documentation.