

# Logging in JavaScript Adapters

## Overview

This tutorial provides the required code snippets in order to add logging capabilities in a JavaScript adapter.

## Logging example

The message below outputs to the `trace.log` file of the application server. If the server administrator is forwarding logs from the MobileFirst Server to the MobileFirst Analytics server the `logger` message will also appear in the Server Log Search view in the MobileFirst Analytics Console.

```
MFP.Logger.debug("This is a debug message from a JavaScript adapter");
```

Additional logging levels, from least to most verbose: ERROR, WARN, INFO, LOG and DEBUG.

## Server Setup

To set-up logging in JavaScript adapters:

1. In MobileFirst Operations Console select the **Settings** option from the sidebar navigation.
2. Click the **Edit** button in the **Runtime Properties** tab.
3. In the **Analytics → Additional packages** section, specify **MFP.Logger** to forward JavaScript Adapter logs to the MobileFirst Server.

The screenshot displays the MobileFirst Operations Console interface. The top navigation bar includes the title 'MobileFirst Operations Console', a link to 'Analytics Console', a user profile 'Hello, admin', and a help icon. The left sidebar lists various navigation options: Dashboard, mfp runtime, Applications (1), PinCode, Adapters (2), Runtime Settings (highlighted), Error Log, Devices, and Download Center. The main content area shows the 'Runtime Settings' page for the 'mfp' application. It features three tabs: 'Runtime Properties' (selected), 'Keystore', and 'Confidential Clients'. The 'Runtime Properties' tab contains a form with the following sections: 'Adapters' (with a 'GZIP compression threshold for adapter responses' field set to 20480), 'Analytics' (with an 'Additional packages' field containing 'MFP.Logger' and a green checkmark), and 'Data collection enabled' (with a dropdown set to 'true'). The page includes 'Save', 'Cancel', and 'Restore Default Values' buttons at the bottom.