

# Debugging Adapters

fork and edit tutorial (<https://github.ibm.com/MFPSamples/DevCenter/tree/master/tutorials/en/foundation/7.0/server-side-development/debugging-adapters.html>) | report issue (<https://github.ibm.com/MFPSamples/DevCenter/issues/new>)

## Overview

The available debugging options for adapters are:

- Testing Adapter Procedures
- Debugging Java Adapters

## Testing the adapter procedures

### CLI

In the terminal, within your project, use `$ mfp adapter call` and follow the interactive menu to test an adapter. Make sure the adapter is built and deployed before you try.

### Studio

It is possible to test adapter procedures by using MobileFirst Studio.

Testing a procedure is done by right-clicking an adapter folder and selecting **Run As > Call MobileFirst Adapter**.



After selecting to call the adapter, you can select the procedure you wish to run. you can also enter comma-separated parameters if the selected procedure requires that.

Call MobileFirst Procedure

Adapter Name : MyAdapter

Procedure name :
getStories (interest)

REST Call Type :
GET

Procedure Arguments

Headers

| Key      | Value |
|----------|-------|
| interest |       |
|          |       |
|          |       |
|          |       |
|          |       |
|          |       |
|          |       |
|          |       |
|          |       |
|          |       |

Load

Save

Remember to surround strings with quotes.

Run

Cancel

Adapter invocation result:

Invocation Result of procedure: 'getStories' from the MobileFirst Server:

```
{
  "errors": [
  ],
  "info": [
  ],
  "isSuccessful": true,
  "responseHeaders": {
    "Alternate-Protocol": "80:quic,p=0.01,80:quic,p=0.01",
    "Cache-Control": "private, max-age=0",
    "Content-Type": "text/xml; charset=UTF-8",
    "Date": "Tue, 28 Oct 2014 12:44:22 GMT",
    "ETag": "X8aekjl3CvT45xpcepn6EK2pDJw",
    "Expires": "Tue, 28 Oct 2014 12:44:22 GMT",
    "Last-Modified": "Tue, 28 Oct 2014 12:44:19 GMT",
    "Server": "GSE",
    "Transfer-Encoding": "chunked",
    "X-Content-Type-Options": "nosniff",
    "X-XSS-Protection": "1; mode=block"
  },
  "responseTime": 299,
  "rss": {
    "channel": {
      "copyright": "Copyright 2014 Cable News Network LP, LLLP.",
      "description": "CNN.com delivers up-to-the-minute news and information on the latest top stories, weather, entertainment, politics and more.",
      "image": {
        "description": "CNN.com delivers up-to-the-minute news and information on the latest top stories, weather, entertainment, politics and more.",
        "height": "33",
        "link": "http://vedition.cnn.com/V/index.html?eref=edition",
        "title": "CNN.com - Top Stories",
        "url": "http://vi.cdn.turner.com/cnn/v.eVimgV1.0VlogoVcnn.logo.rss.gif",
        "width": "144"
      },
      "info": {
        "feedburner": "http://vrssnamespace.org/V/feedburnerV/extV1.0",
        "uri": "rssVedition"
      },
      "item": [
        {
          "description": "The South African state is to appeal both judgment and sentence after athlete Oscar Pistorius was jailed for five years for shooting his girlfriend.",
          "guid": "http://vedition.cnn.com/V2014V10V27VjusticeVsouth-africa-oscar-pistorius-appealV/index.html",

```

# Debugging Java Adapters

To debug Java adapter code, you must use Eclipse in debug mode. This required the MobileFirst Studio (you can import you CLI project if needed).

## Starting debug mode

1. Right-click a Java adapter, then select **Debug As -> Debug MobileFirst Java Adapters**.
2. Optional: Open the debug perspective in MobileFirst Studio.

After you entered debug mode, you can debug the Java code normally, as you would do a standard Java application. You might need to issue a request to the adapter to make its code run and hit the breakpoints.

## Stopping debug mode

1. Open the debug perspective in MobileFirst Studio.
2. On the **Debug** tab, right-click the item **remote debug [Remote Java Application]** -> **Terminate**.

