

Tips & Troubleshooting

- [Tips](#)
- [Troubleshooting](#)

Tips

[top](#)

1. The enclosing element should be a block-level element, such as a div
2. The enclosing element should have no padding or border. If you want a border or padding around your report or dashboard you should enclose the element in another element that has the padding or border. For example:

```
<div style="padding: 10px; border: 1px solid black;">
  <div id="myReport">
    </div>
  </div>
<script type="text/javascript">
  yellowfin.loadReport({reportUUID: 'e5e5aaf3-c3b8-4f9b-8280-e21e4d848e63', elementId: 'myReport' });
</script>
```

If you are using the basic method, and do not pass an elementId to the API, the enclosing element is created automatically. You can enclose the script tag in another element with padding or border styles applied. For example:

```
<div style="padding: 10px; border: 1px solid black;">
  <script type="text/javascript" src="http://localhost/JsAPI?reportUUID=e5e5aaf3-c3b8-4f9b-8280-e21e4d848e63&width=500&height=500"></script>
</div>
```

3. For reports, the enclosing element should have a fixed height/width.
For chart only reports, the chart will be rendered to fit within the enclosing element. In case a report does not fit within the element, scrollbars will be included, so the rest of your page layout is not affected.
If you do not include a fixed height and width on your element, the browser will choose how to layout the report. This may mean that the report will increase or decrease in size when different options are clicked by the user. This may be preferable in some cases.
4. Make sure the Javascript API URL is accessible for any end-users. Avoid using localhost or private IP addresses.
5. Your HTML should be well-structured where possible. For example, you may have problems with the API if you do not have a <body> tag.

Troubleshooting

[top](#)

As mentioned previously, you can check the `window.yellowfin` variable to determine if the Javascript API has been loaded successfully. For example, when using the basic method:

```
<script src="http://localhost/JsAPI?reportUUID=e5e5aaf3-c3b8-4f9b-8280-e21e4d848e63&width=500&height=500" type="text/javascript"></script>
<script type="text/javascript">
  if (!window.yellowfin) {
    alert('Error loading API');
  }
</script>
```

When using the advanced method, you can use this to determine whether or not to attempt to load a report or dashboard:

```
<script src="http://localhost/JsAPI" type="text/javascript"></script>
<script type="text/javascript">
  if (window.yellowfin) {
    yellowfin.loadReport({reportUUID: 'e5e5aaf3-c3b8-4f9b-8280-e21e4d848e63', elementId: 'myReport' });
  } else {
    alert('Error loading API');
  }
</script>
```

Errors encountered when using the API are generally presented through Javascript alerts. These include a description of the error. Common errors include:

Error	Description
API version requested is not supported	The server does not support the API version requested.
reportUUID not specified	The <code>yellowfin.loadReport</code> function is called without a report identifier (<code>reportUUID</code> , <code>reportId</code> or <code>wsName</code>).
Report not found	The report specified by the <code>reportUUID</code> , <code>reportId</code> or <code>wsName</code> could not be found.
Report not loaded	An operation has been attempted on a report that has not yet been loaded.
Report is in draft mode	The report specified is in draft mode. Only active reports can be accessed through the Javascript API.
User does not have access to this report	The user logged in does not have access to the requested report.
Invalid command specified Invalid request	An invalid request has been made to the API.
Error loading report Error running report Error running report command Error loading dashboard	A server-side error occurred while running an API command. Check the Yellowfin server logs for more information.
dashUUID not specified	The <code>yellowfin.loadDash</code> function is called without a dashboard identifier (<code>dashUUID</code>).
Dashboard not found	The dashboard specified by the <code>dashUUID</code> could not be found.
User does not have access to this dashboard	The user logged in does not have access to the requested dashboard.
Dashboard is in draft mode	The dashboard specified is in draft mode. Only active dashboards can be accessed through the Javascript API.

[top](#)