

## Scenario 1:

**Create a scheduler (Product Sales) to get monthly sales every first Monday of the month :**

```
email = user.get("email");
workspaceName = "<enter_workspace_name>";
viewName = "<enter_table/report_name>";
paramsMap = Map();
paramsMap.put("ZOHO_ACTION","EXPORT");
paramsMap.put("ZOHO_OUTPUT_FORMAT","JSON");
paramsMap.put("ZOHO_ERROR_FORMAT","JSON");
paramsMap.put("ZOHO_API_VERSION","1.0");
url = "https://analyticsapi.zoho.com/api/" + email + "/" + workspaceName + "/" +
viewName;
url = url.replaceAll(" ","%20");
getSalesReports = invokeurl
[
    url :url
    type :GET
    parameters:paramsMap
    detailed:true
    connection:"<enter_connection_name>"
];
info getSalesReports;
allReports =
getSalesReports.get("responseText").get("response").get("result").get("rows");
headersList =
getSalesReports.get("responseText").get("response").get("result").get("column_order");
rowsList = list();
for each report in allReports
{
rowsList.add({headersList.get(0):report.get(0),headersList.get(1):report.get(1),headersL
ist.get(2):report.get(2),headersList.get(3):report.get(3),headersList.get(4):report.get(4),h
eadersList.get(5):report.get(5)});
}
postMessage = {"text":"### Number of products sold in November
2022","card":{"theme":"10","title":"Sales
2022"},"slides":{"type":"table","data":{"headers":headersList,"rows":rowsList}}},"bot":{"nam
```

```
e:"Sales Bot","image":"","});  
info zoho.cliq.postToChannel("<channel_unique_name>",postMessage);
```

### **Scenario 2:**

**Create a command in Cliq (/sales) to post message in a channel to get sales made in 2022 :**

```
email = user.get("email");  
workspaceName = "<enter_workspace_name>";  
viewName = "<enter_table_name>";  
paramsMap = Map();  
paramsMap.put("ZOHO_ACTION","EXPORT");  
paramsMap.put("ZOHO_OUTPUT_FORMAT","JSON");  
paramsMap.put("ZOHO_ERROR_FORMAT","JSON");  
paramsMap.put("ZOHO_API_VERSION","1.0");  
url = "https://analyticsapi.zoho.com/api/" + email + "/" + workspaceName + "/" +  
viewName;  
url = url.replaceAll(" ","%20");  
getSalesReports = invokeurl  
[  
    url :url  
    type :GET  
    parameters:paramsMap  
    detailed:true  
    connection:"<enter_connection_name>"  
];  
info getSalesReports;  
allReports =  
getSalesReports.get("responseText").get("response").get("result").get("rows");  
headersList =  
getSalesReports.get("responseText").get("response").get("result").get("column_order");  
rowsList = list();  
for each report in allReports  
{  
  
rowsList.add({headersList.get(0):report.get(0),headersList.get(1):report.get(1),headersL  
ist.get(2):report.get(2)});  
}
```

```

postMessage = {"text":"### Number of sales made in
2022","card":{"theme":"10","title":"Sales
2022"},"slides":{"type":"table","data":{"headers":headersList,"rows":rowsList}}}};
return postMessage;

```

### Scenario 3

**Create a command (/orderrequest) to display a message card :**

```

postMessage = {"text":"### Order request raised
successfully!","slides":{"type":"text","title":"Product Details","buttons":{"label":"Pick
Up","action":{"type":"invoke.function","data":{"name":"orderPickup"}}},"data":"*Product:*
Laptop\n*Raised On:* " + zoho.currentdate.toString("dd/MMM/yyyy")}}};
info zoho.cliq.postToChat(chat.get("id"),postMessage);
return Map();

```

**Create a function(/orderPickup) to execute when a user clicks Pickup button :**

```

info message;
contentDetails = message.get("content").get("slides").get(0);
dataMessage = contentDetails.get("data") + "\n*Picked by:* " + user.get("first_name");
email = user.get("email");
workspaceName = "<enter_workspace_name>";
viewName = "<enter_report_name>";
// fetch existing count in analytics
targetId = 4;
paramsMap = Map();
paramsMap.put("ZOHO_ACTION","EXPORT");
paramsMap.put("ZOHO_OUTPUT_FORMAT","JSON");
paramsMap.put("ZOHO_ERROR_FORMAT","JSON");
paramsMap.put("ZOHO_API_VERSION","1.0");
query = "SELECT * FROM \"<enter_report_name>\" WHERE \"S.No\"=\"\" + targetId + \"\"";
paramsMap.put("ZOHO_SQLQUERY",query);
url = "https://analyticsapi.zoho.com/api/" + email + "/" + workspaceName + "/" +
viewName;
url = url.replaceAll(" ","%20");
info url;
getRecord = invokeurl
[
    url:url

```

```

        type :GET
        parameters:paramsMap
        detailed:true
        connection:"<enter_connection_name>"
];
info getRecord;
deliveryCount =
getRecord.get("responseText").get("response").get("result").get("rows").get(0).get(2);
deliveryCount = abs(deliveryCount.toNumber() + 1);
// update record in analytics
targetId = 4;
paramsMap = Map();
paramsMap.put("ZOHO_ACTION","UPDATE");
paramsMap.put("ZOHO_OUTPUT_FORMAT","JSON");
paramsMap.put("ZOHO_ERROR_FORMAT","JSON");
paramsMap.put("ZOHO_API_VERSION","1.0");
paramsMap.put("ZOHO_CRITERIA","\S.No\"=" + targetId + "");
paramsMap.put("Deliveries made",deliveryCount);
url = "https://analyticsapi.zoho.com/api/" + email + "/" + workspaceName + "/" +
viewName;
url = url.replaceAll(" ","%20");
updateRecord = invokeurl
[
    url :url
    type :POST
    parameters:paramsMap
    detailed:true
    connection:"<enter_connection_name>"
];
info updateRecord;
postMessage =
{"text":message.get("text"),"slides":{{"type":"text","title":contentDetails.get("title"),"data":dataMessage}}};
info zoho.cliq.editMessage(chat.get("id"),message.get("id"),postMessage);
return {"type":"banner","text":"You have picked successfully.,"status":"success"};

```

#### **Scenario 4:**

**Create a widget using iframe url :**

```
tabsList = list();
tabsList.add({"label":"Dashboard","id":"dashboard"});
if(target.containsKey("id"))
{
    activeTab = target.get("id");
}
else
{
    activeTab = "dashboard";
}
return
{"type":"applet","data_type":"web_view","tabs":tabsList,"web_view":{"url":"<enter_iframe_url
>"},"active_tab":activeTab};
```