arRESTed Development

Using REST APIs for Monthly Reporting

precursor.ca/slides

Precursor Systems

EST'd 1994

precursor.ca/slides

“Alex Narvey is the original Mac MSP”

- Allen Hancock

precursor.ca/slides

Mac *mini* Dev Ops

precursor.ca/slides

Basics

Interacting with REST APIs

Examples

Monthly Reporting with REST

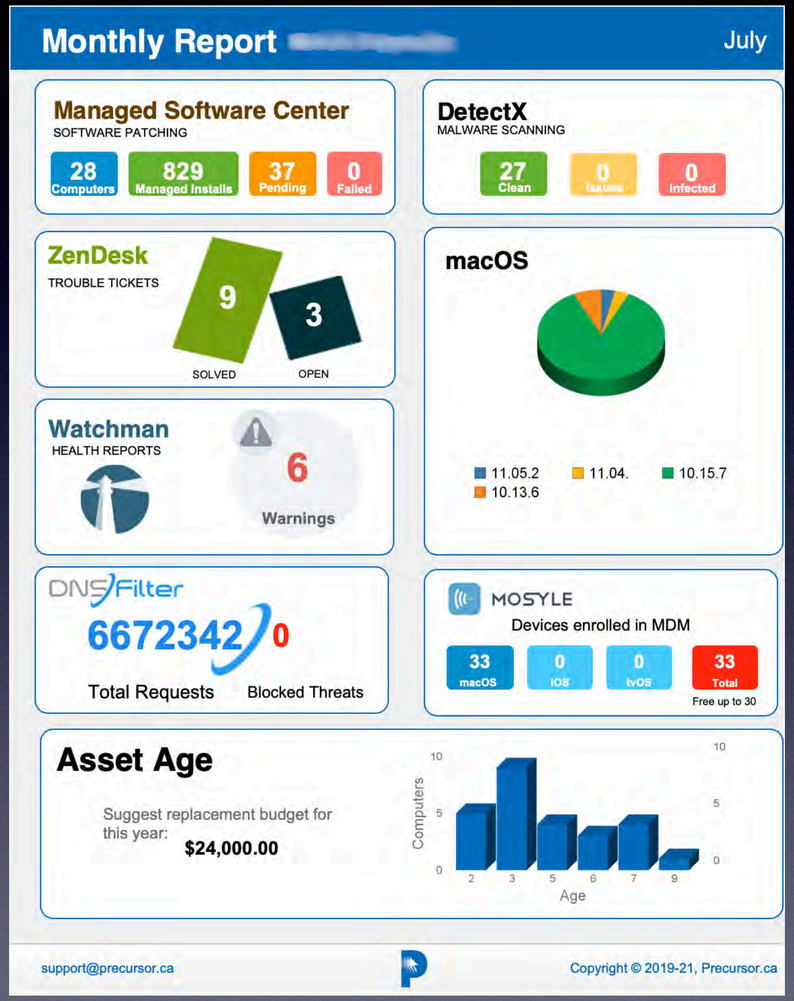
precursor.ca/slides

Software Patches —

Support Tickets —

Warnings —

DNS Filtering —



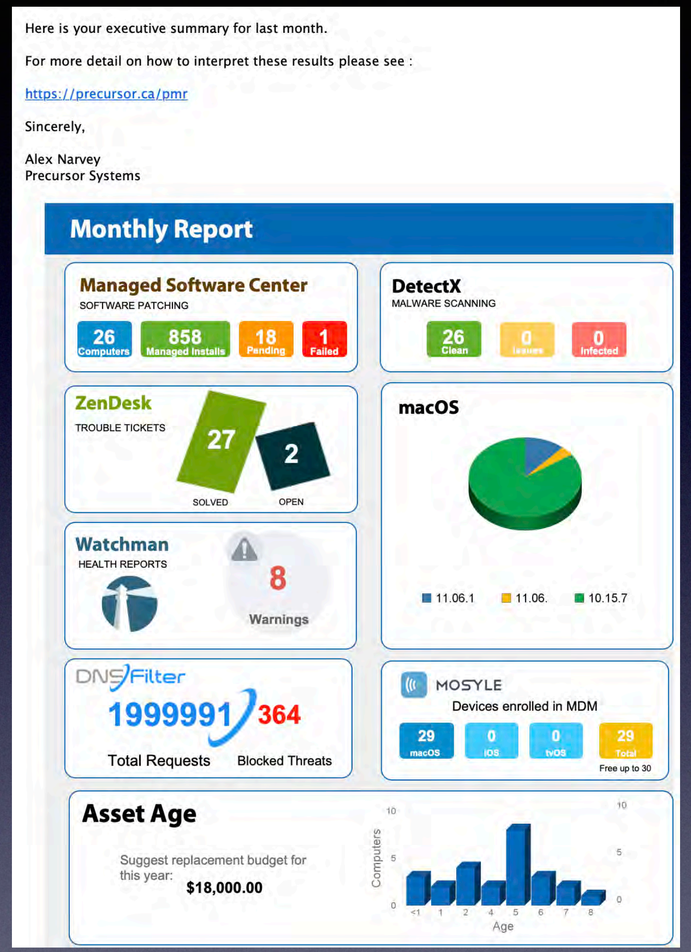
— Malware

— macOS Version

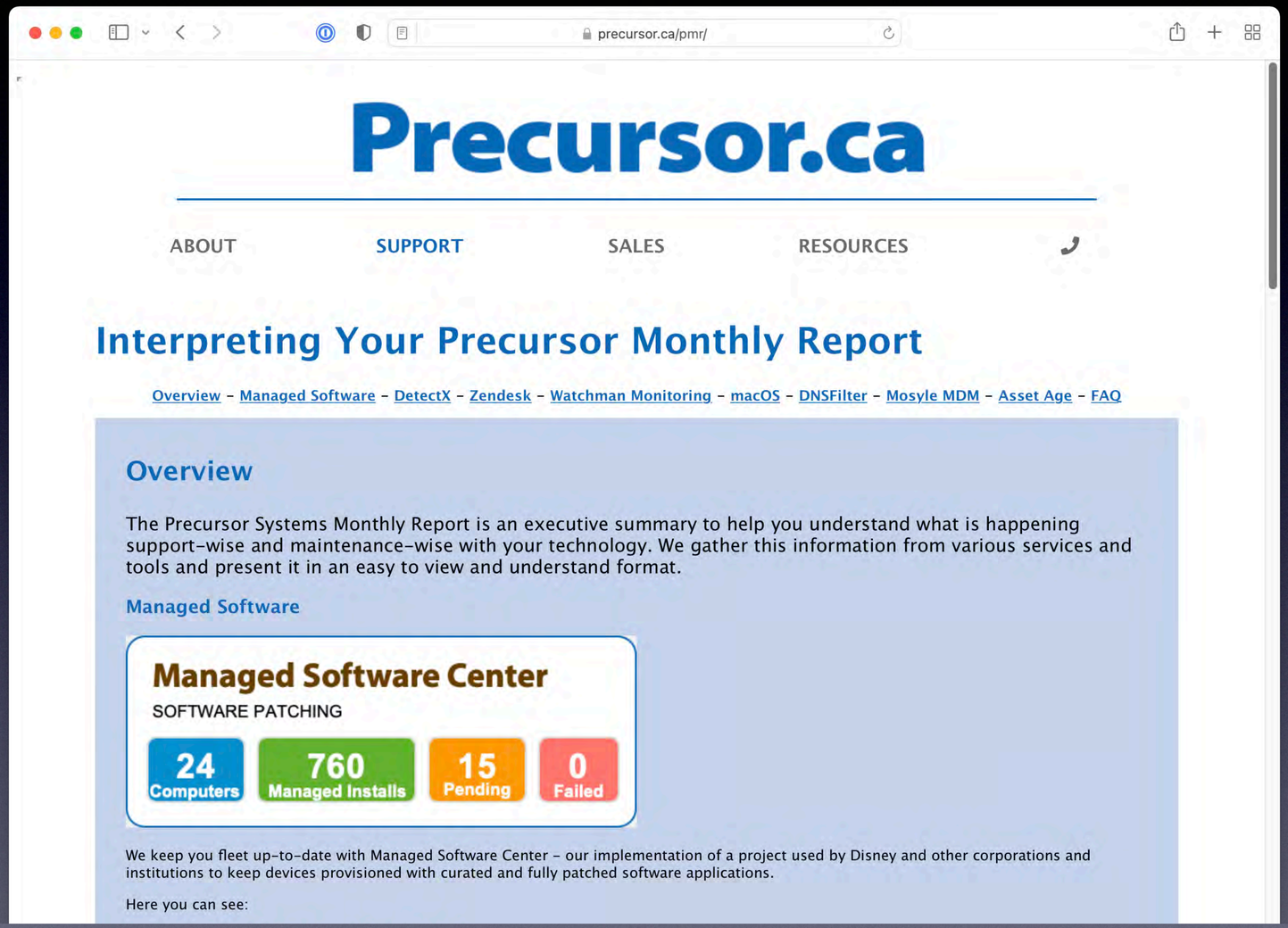
— MDM Devices

— Hardware Age

precursor.ca/slides



precursor.ca/slides



precursor.ca/slides

REST APIs

precursor.ca/slides

“The world is just a collection of REST endpoints.”

Charles Edge

precursor.ca/slides

{ REST }

Representational State Transfer

precursor.ca/slides

cURL

precursor.ca/slides

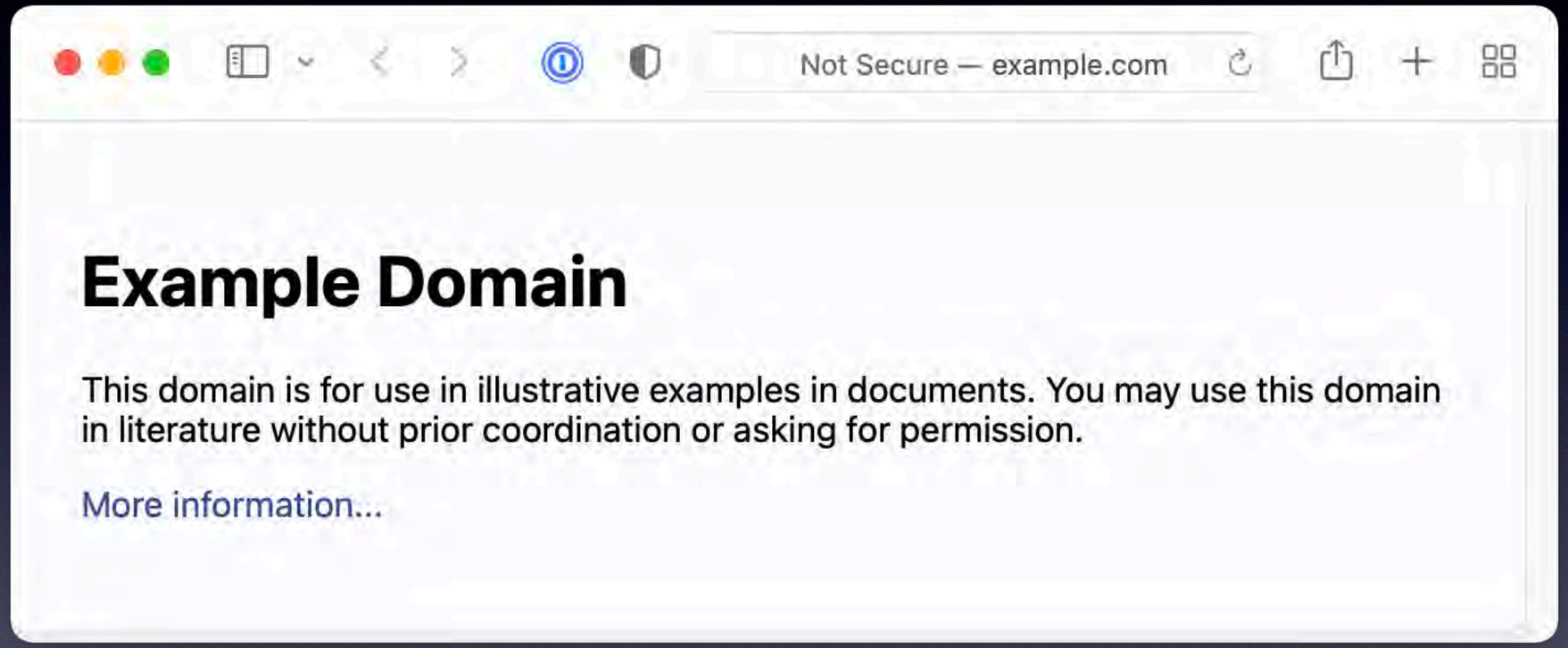
cURL, which stands for client URL, is a command line tool that developers use to transfer data to and from a server.

precursor.ca/slides

The most basic command in curl is:

```
curl http://example.com
```

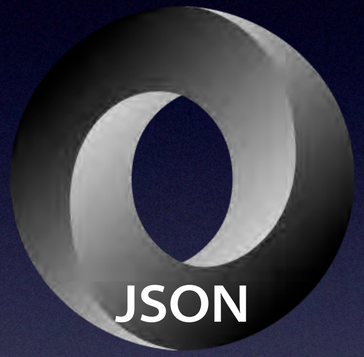
precursor.ca/slides



precursor.ca/slides

```
<!doctype html>
<html>
<head>
    <title>Example Domain</title>

    <meta charset="utf-8" />
    <meta http-equiv="Content-type" content="text/html; charset=utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <style type="text/css">
    body {
        background-color: #f0f0f2;
        margin: 0;
        padding: 0;
        font-family: -apple-system, system-ui, BlinkMacSystemFont, "Segoe UI", "Open Sans",
"Helvetica Neue", Helvetica, Arial, sans-serif;

    }
    div {
        width: 600px;
        margin: 5em auto;
        padding: 2em;
        background-color: #fdfdff;
        border-radius: 0.5em;
        box-shadow: 2px 3px 7px 2px rgba(0,0,0,0.02);
    }
    a:link, a:visited {
```

precursor.ca/slides



JavaScript Object Notation

precursor.ca/slides

Buildinfo.json

```
{
    "distribution_style": true,
    "identifier": "com.eriknq.installapplications",
    "install_location": "/",
    "name": "InstallApplications-${version}.pkg",
    "ownership": "recommended",
    "postinstall_action": "none",
    "signing_info": {
        "identity": "Mac Installer: Erik Gomez (XXXXXXXXXX)",
        "timestamp": true
    },
    "suppress_bundle_relocation": true,
    "version": "2.1"
}
```

precursor.ca/slides

```
cURL https://businessapi.mosyle.com/v1/users
```

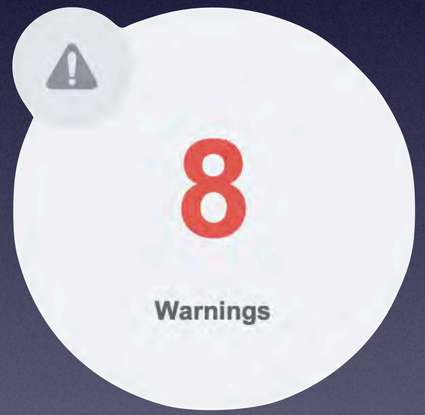
precursor.ca/slides


```
{
  "status": "OK",
  "response": [
    {
      "users": [
        {
          "iduser": "69",
          "code": “2QTBH",
          "name": "Alex Narvey",
          "type": "ADMIN",
          "identifier": "alex",
          "email": "anarvey@precursor.ca",
          "is_removed": false
        }
      ],
      "page": 1
    }
  ]
}
```


precursor.ca/slides

cURL

```
https://exampleco.monitoringclient.com/v2.5/computers/?
api_key="123456789"&group_id="24"&expand[]=plugin_results"
```



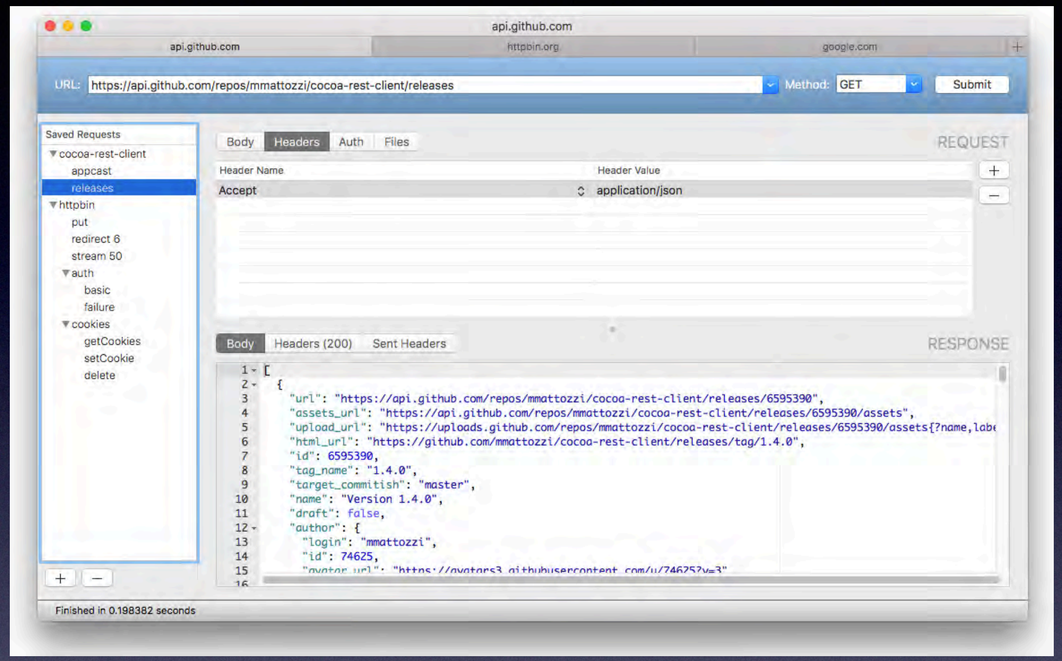
precursor.ca/slides

```
{
    "plugins" :
    [
        {
            "active_mac_address" : "A8:60:B6:03:E5:91",
            "agent_removed" : false,
            "agent_version" : "6.6.9.101",
            "apple_product_description" : "Mac mini (Late 2014)",
```

•••

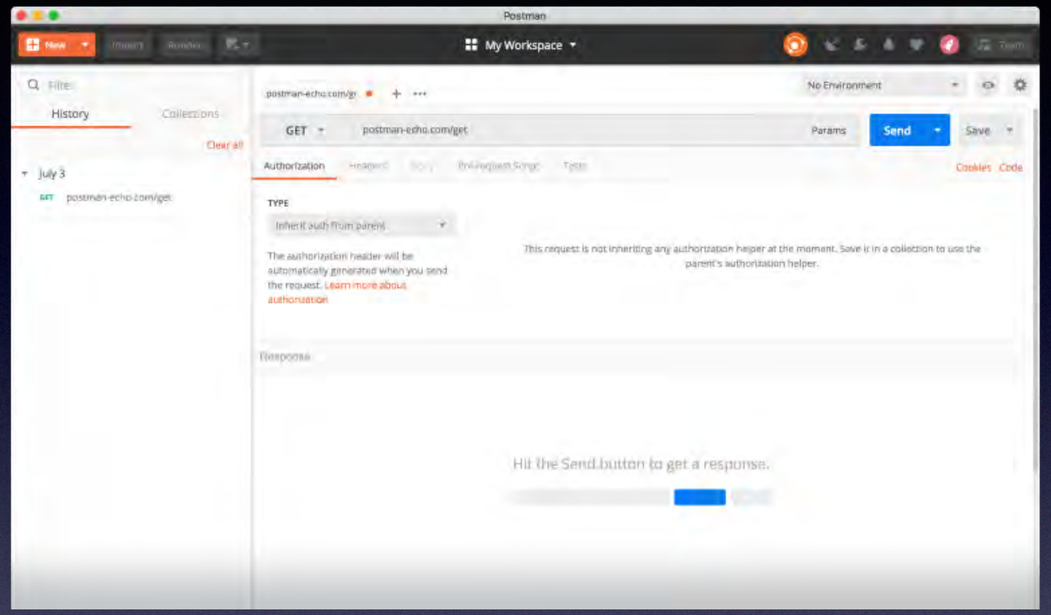
```
            "teamviewer_id" : null,
            "teamviewer_release" : null,
            "teamviewer_unattended" : null,
            "uid" : "c_5553c4e38a",
            "updates_enabled" : true,
            "watchman_id" : "20160106-BF59-UN4R8B"
        }
    ]
}
```

precursor.ca/slides



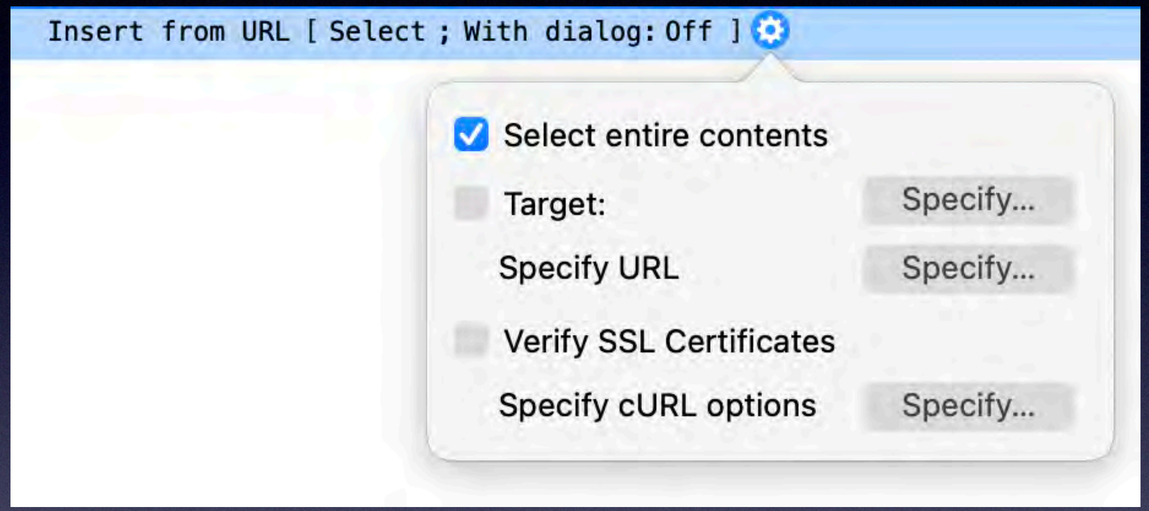
https://mmattozzi.github.io/cocoa-rest-client/

precursor.ca/slides



https://www.getpostman.com/downloads/

precursor.ca/slides



https://claris.com/filemaker/

precursor.ca/slides

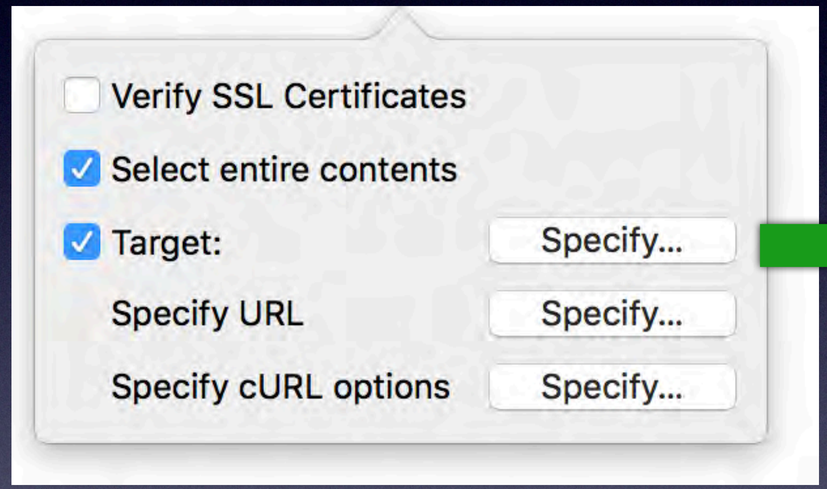
precursor.ca/slides

Insert From URL

Takes cURL commands

precursor.ca/slides

Target

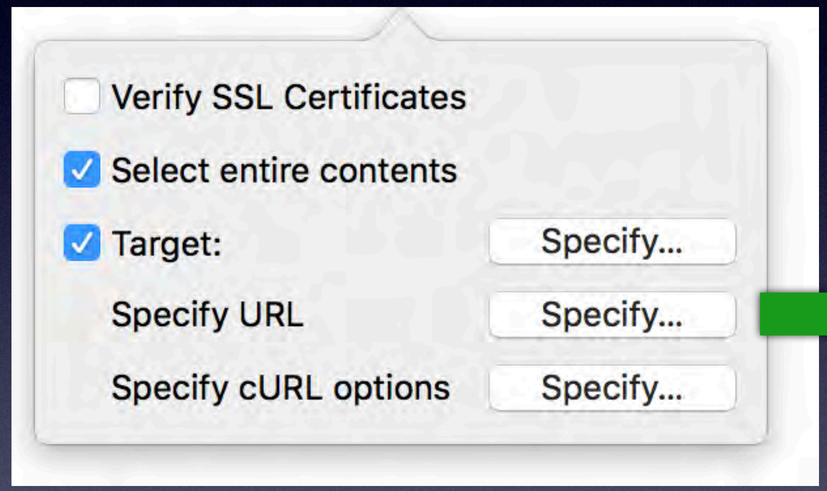


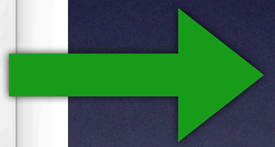
FileMaker Field

FileMaker Variable

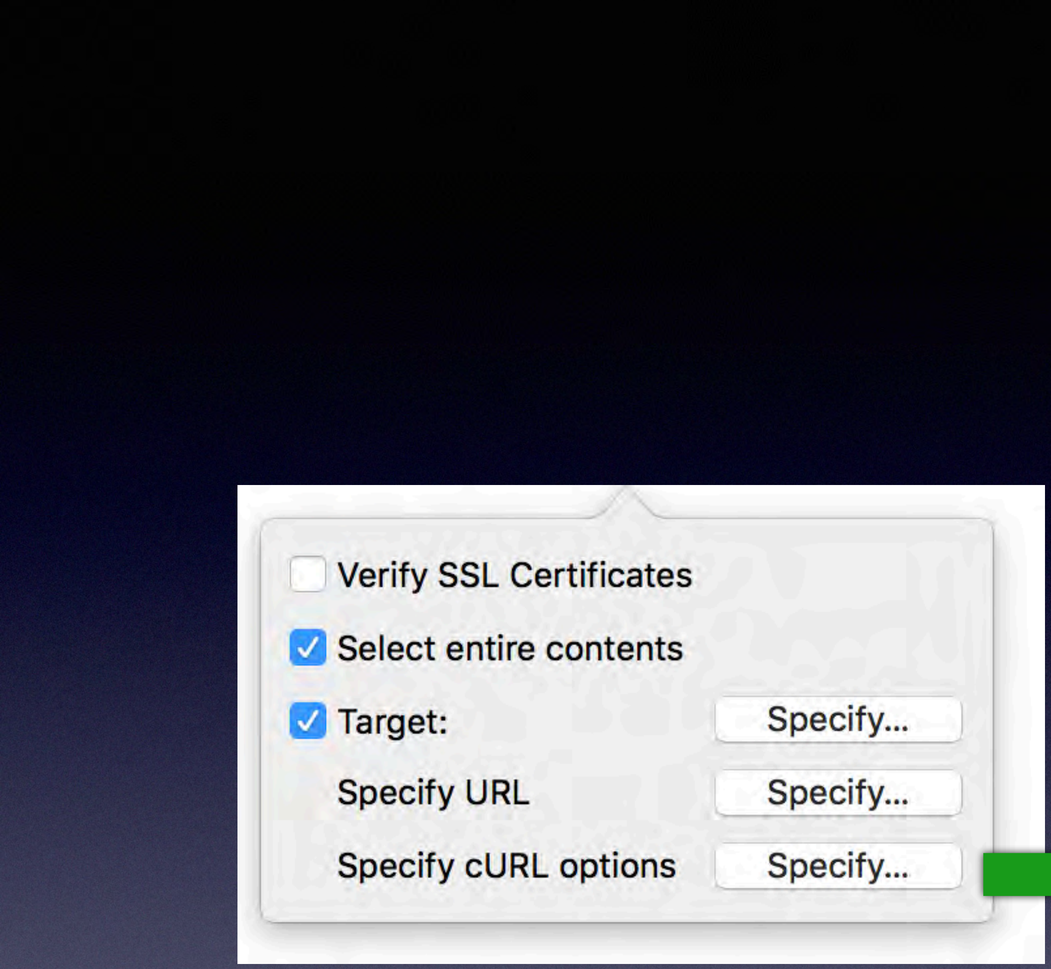
precursor.ca/slides

URL





```
https://api.example.com/v1/groups
```



cURL Options

Header
Data
Form data
Authentication
Cookies

precursor.ca/slides

“Getting started with the new cURL options is quite daunting and a steep learning curve.”

`https://support.claris.com/s/article/new-curl-options-in-insert-from-url`

precursor.ca/slides

Basic GET Command

Authorization in URL

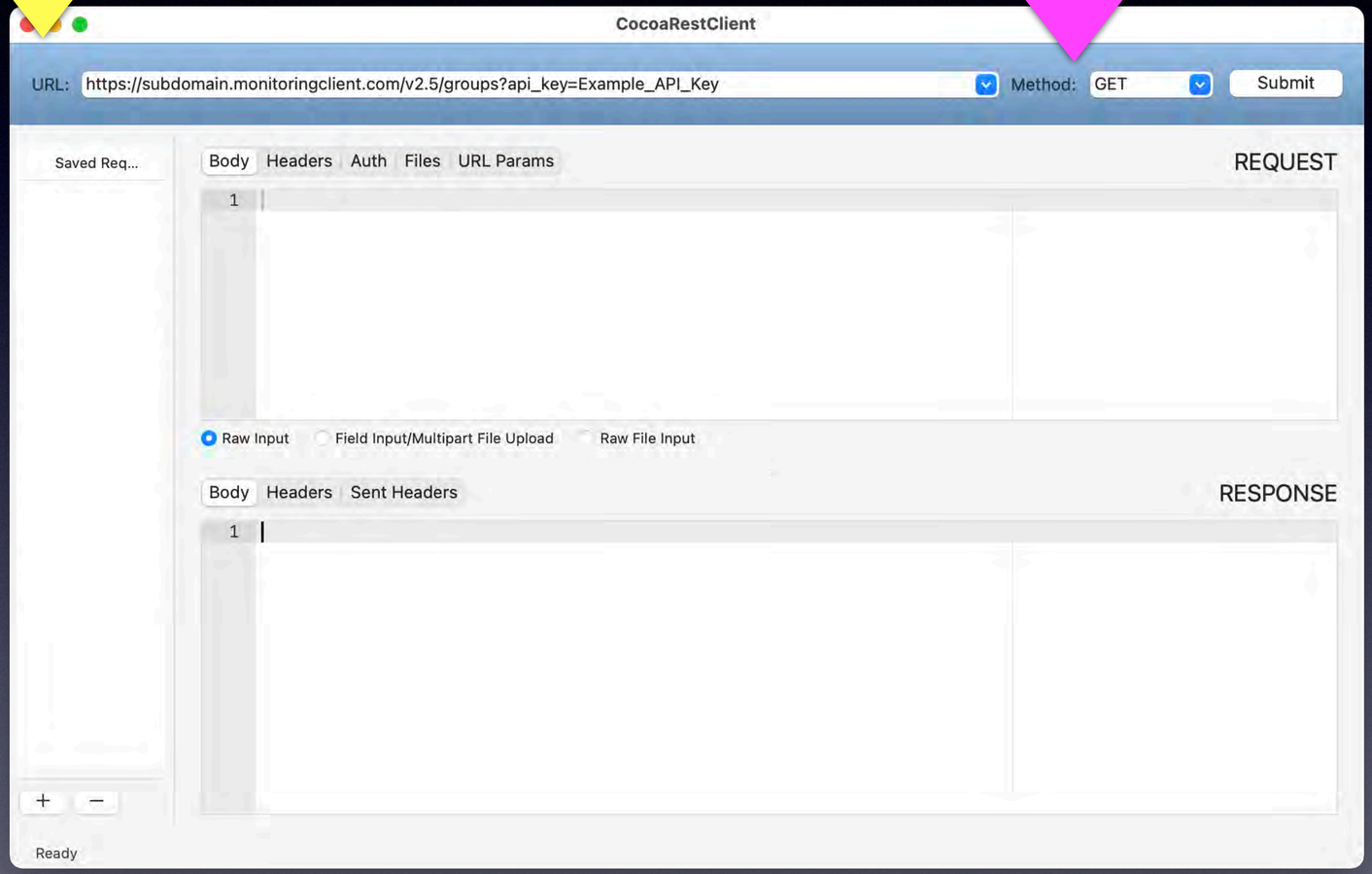
precursor.ca/slides

```
GET   curl https://subdomain.monitoringclient.com/v2.5/groups?api_key=Example_API_Key
```

precursor.ca/slides

```
curl https://subdomain.monitoringclient.com/v2.5/groups?api_key=Example_API_Key
```

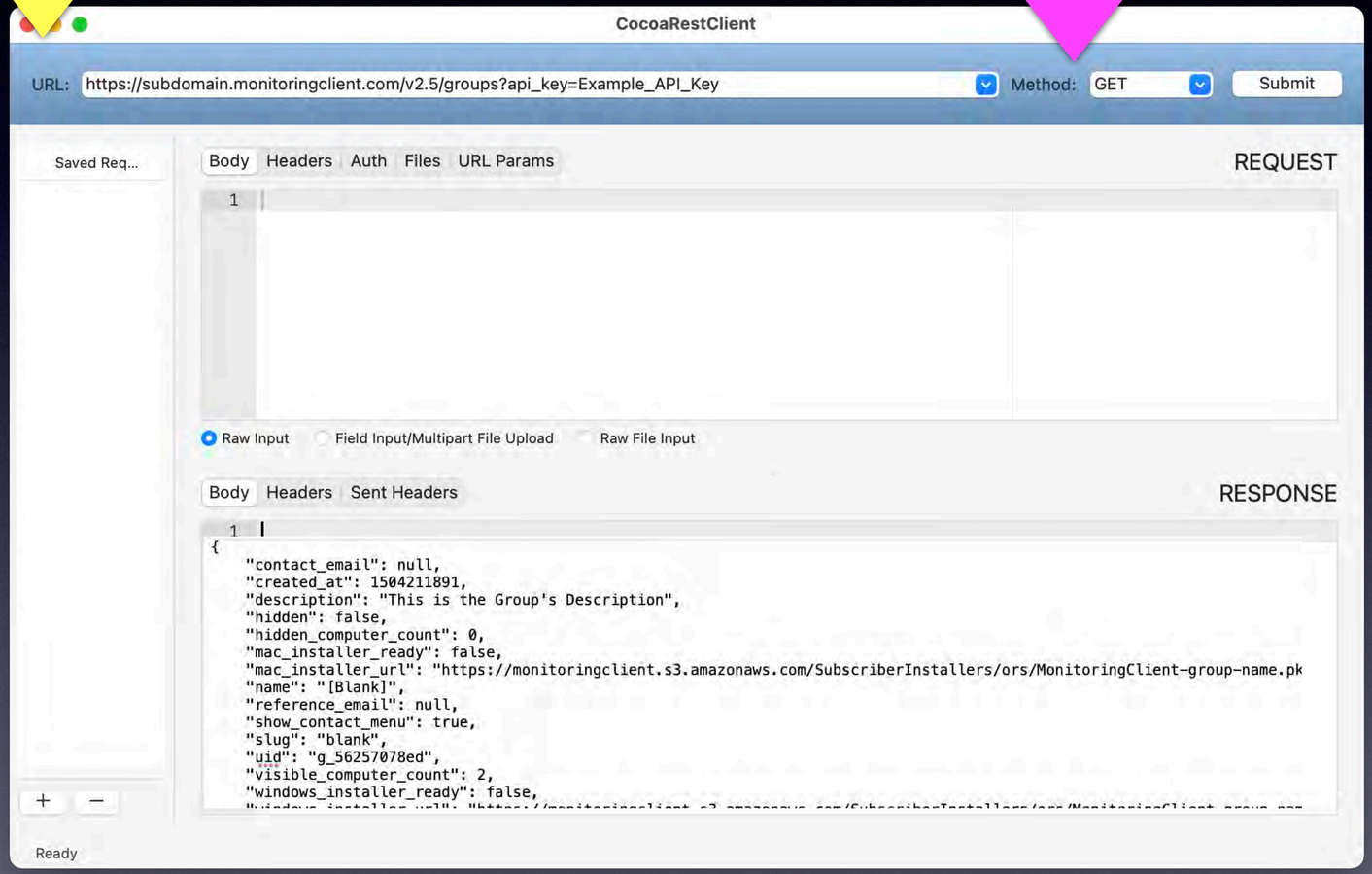
GET



precursor.ca/slides

```
curl https://subdomain.monitoringclient.com/v2.5/groups?api_key=Example_API_Key
```

GET



precursor.ca/slides

```
https://subdomain.monitoringclient.com/v2.5/groups?api_key=Example_API_Key
```

Insert From URL script step

precursor.ca/slides

https://subdomain.monitoringclient.com/v2.5/groups?api_key=Example_API_Key

Script Workspace (WatchmanQuery_1.0.2)

Scripts

Search

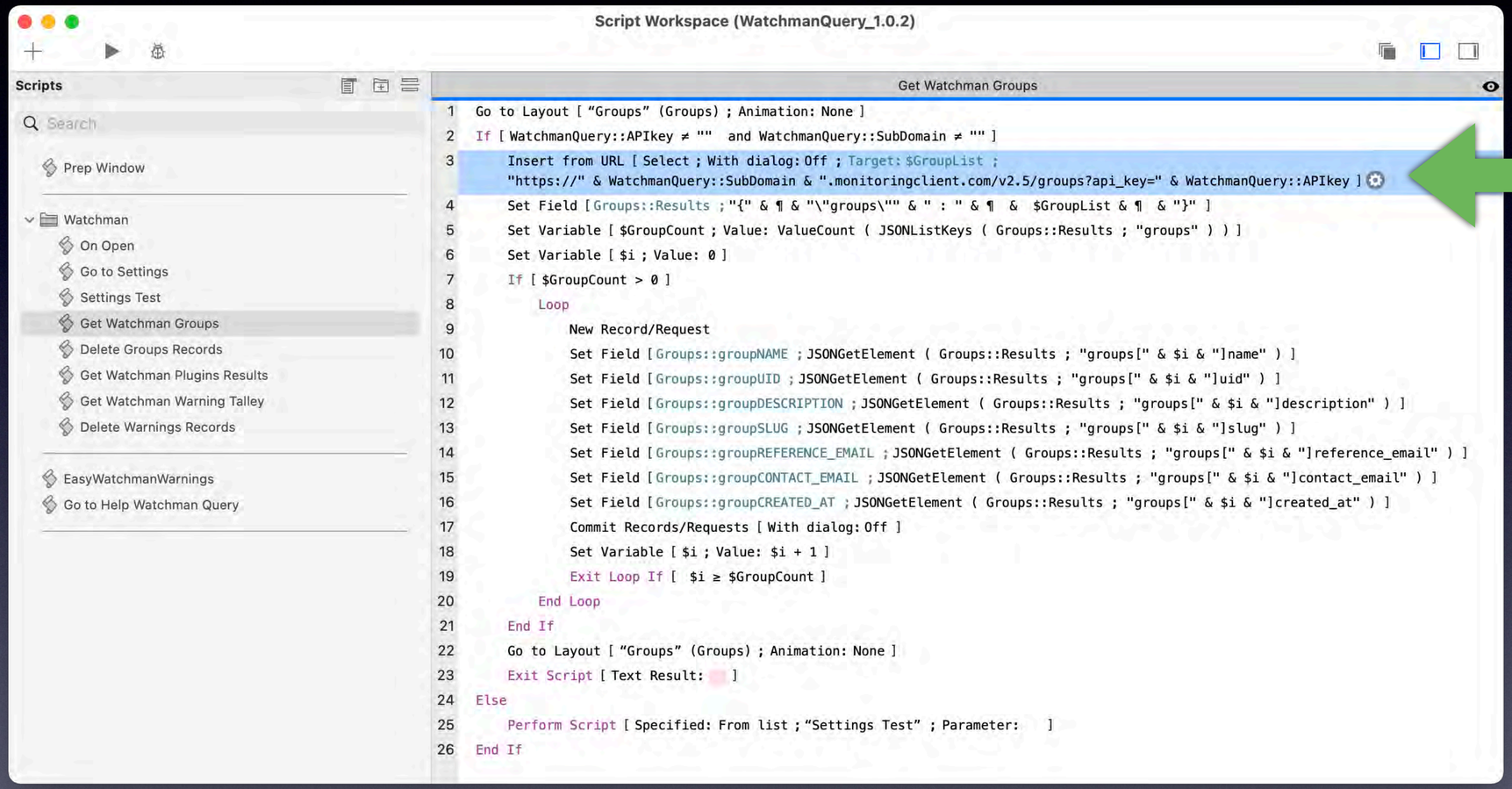
- Prep Window
- Watchman
  - On Open
  - Go to Settings
  - Settings Test
  - Get Watchman Groups
  - Delete Groups Records
  - Get Watchman Plugins Results
  - Get Watchman Warning Talley
  - Delete Warnings Records
- EasyWatchmanWarnings
- Go to Help Watchman Query

Get Watchman Groups

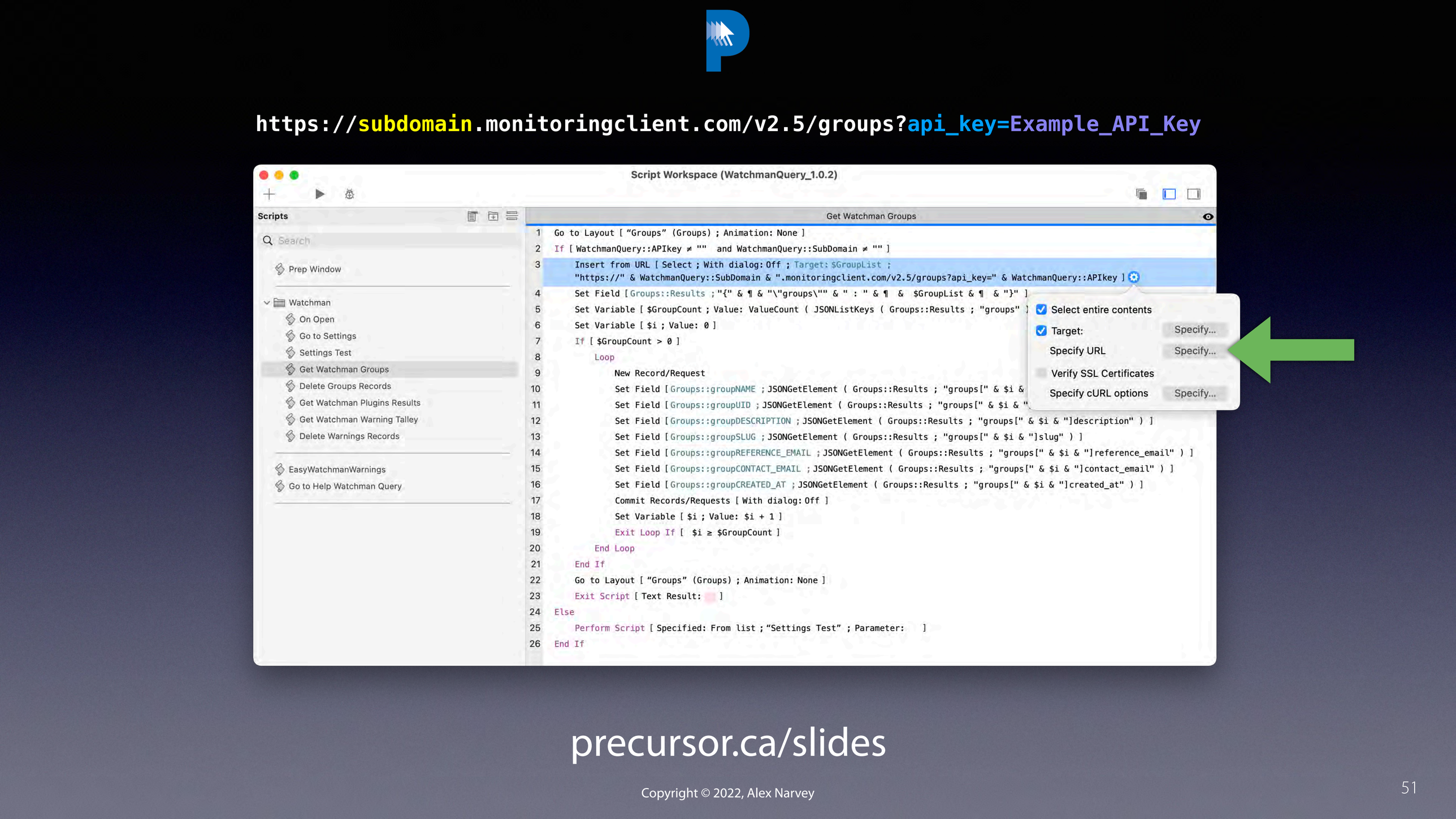
```
1  Go to Layout [ “Groups” (Groups) ; Animation: None ]
2  If [ WatchmanQuery::APIkey ≠ ""  and WatchmanQuery::SubDomain ≠ "" ]
3      Insert from URL [ Select ; With dialog: Off ; Target: $GroupList ;
       "https://" & WatchmanQuery::SubDomain & ".monitoringclient.com/v2.5/groups?api_key=" & WatchmanQuery::APIkey ]
4      Set Field [ Groups::Results ; "{" & ¶ & "\"groups\"" & " : " & ¶  &  $GroupList & ¶  & "}" ]
5      Set Variable [ $GroupCount ; Value: ValueCount ( JSONListKeys ( Groups::Results ; "groups" ) ) ]
6      Set Variable [ $i ; Value: 0 ]
7      If [ $GroupCount > 0 ]
8          Loop
9              New Record/Request
10             Set Field [ Groups::groupNAME ; JSONGetElement ( Groups::Results ; "groups[" & $i & "]name" ) ]
11             Set Field [ Groups::groupUID ; JSONGetElement ( Groups::Results ; "groups[" & $i & "]uid" ) ]
12             Set Field [ Groups::groupDESCRIPTION ; JSONGetElement ( Groups::Results ; "groups[" & $i & "]description" ) ]
13             Set Field [ Groups::groupSLUG ; JSONGetElement ( Groups::Results ; "groups[" & $i & "]slug" ) ]
14             Set Field [ Groups::groupREFERENCE_EMAIL ; JSONGetElement ( Groups::Results ; "groups[" & $i & "]reference_email" ) ]
15             Set Field [ Groups::groupCONTACT_EMAIL ; JSONGetElement ( Groups::Results ; "groups[" & $i & "]contact_email" ) ]
16             Set Field [ Groups::groupCREATED_AT ; JSONGetElement ( Groups::Results ; "groups[" & $i & "]created_at" ) ]
17             Commit Records/Requests [ With dialog: Off ]
18             Set Variable [ $i ; Value: $i + 1 ]
19             Exit Loop If [  $i ≥ $GroupCount ]
20         End Loop
21     End If
22     Go to Layout [ “Groups” (Groups) ; Animation: None ]
23     Exit Script [ Text Result:  ]
24 Else
25     Perform Script [ Specified: From list ; “Settings Test” ; Parameter:  ]
26 End If
```

precursor.ca/slides

https://subdomain.monitoringclient.com/v2.5/groups?api_key=Example_API_Key

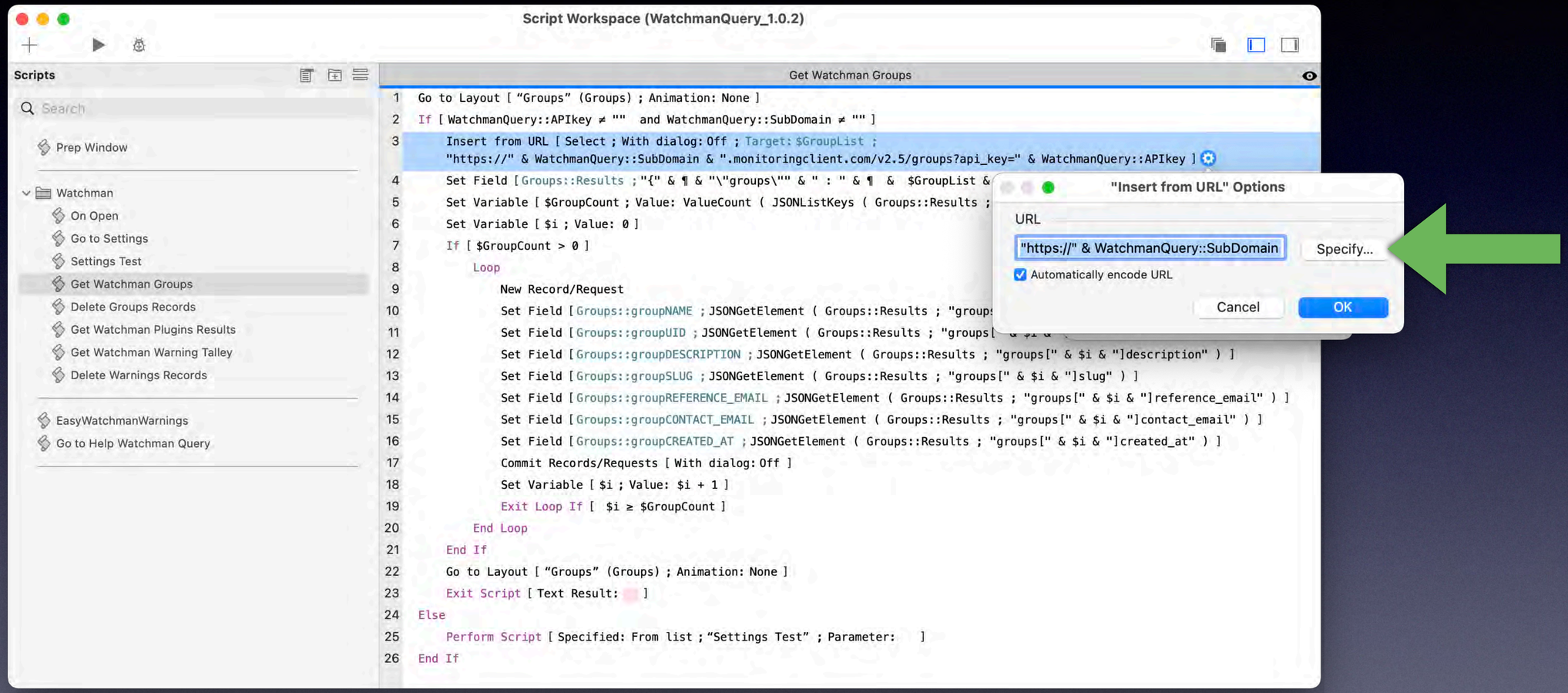


precursor.ca/slides



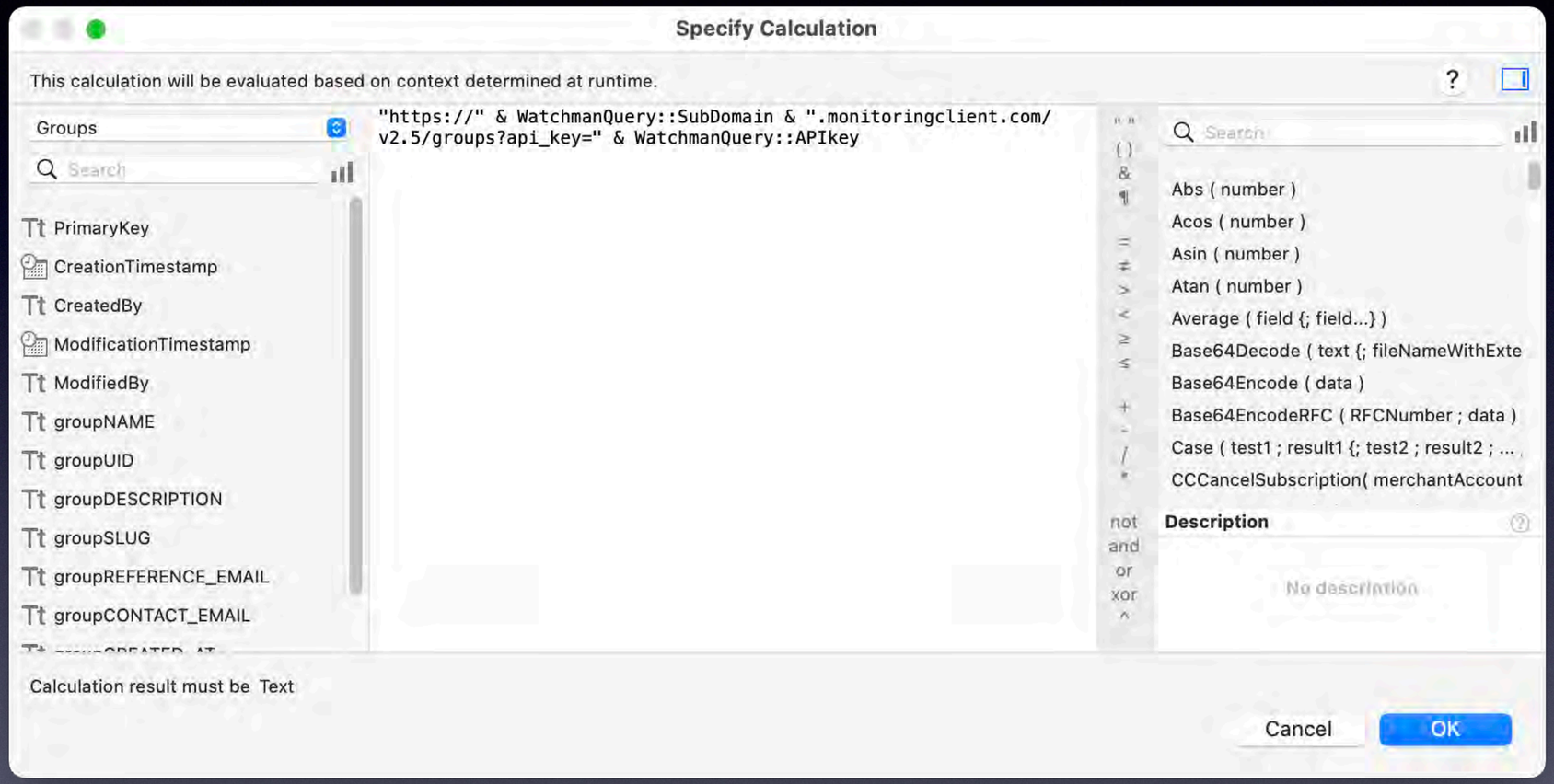
precursor.ca/slides

https://subdomain.monitoringclient.com/v2.5/groups?api_key=Example_API_Key

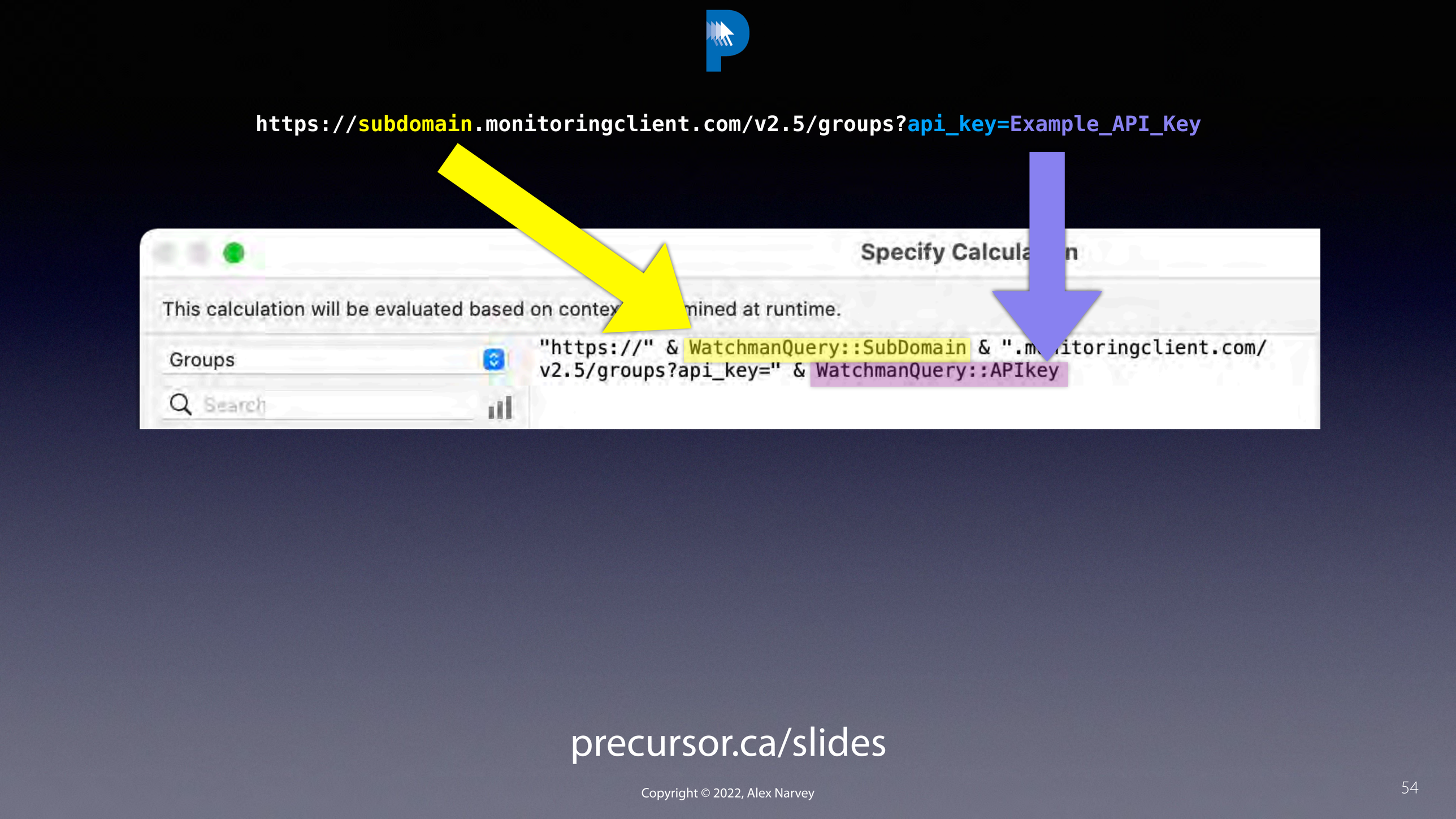


precursor.ca/slides

https://subdomain.monitoringclient.com/v2.5/groups?api_key=Example_API_Key



precursor.ca/slides



precursor.ca/slides

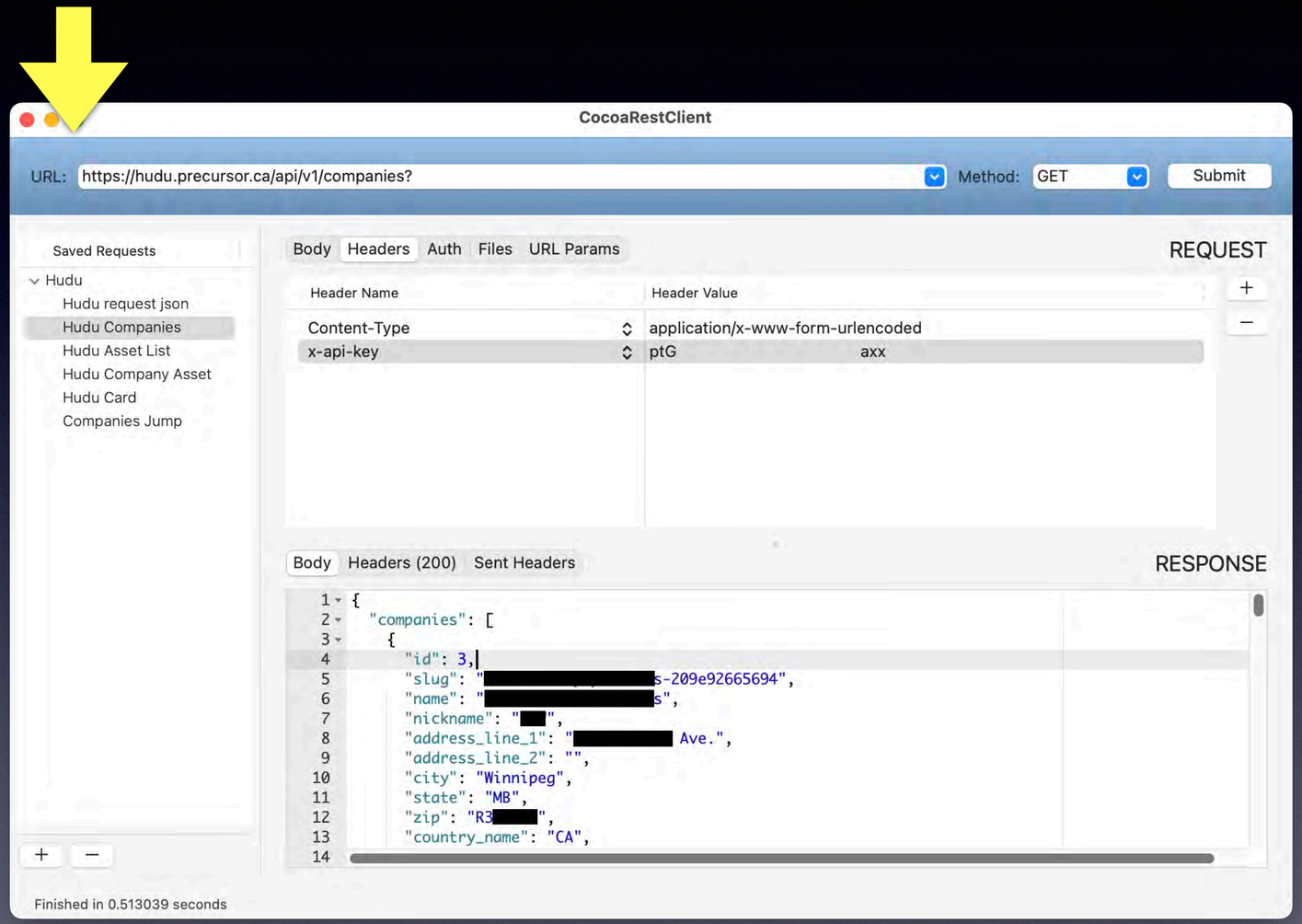
Authentication in Header

precursor.ca/slides

```
GET    curl https://[YOUR_DOMAIN]/api/v1/companies? -H "x-api-key: YOUR_API_KEY_HERE"
```

precursor.ca/slides

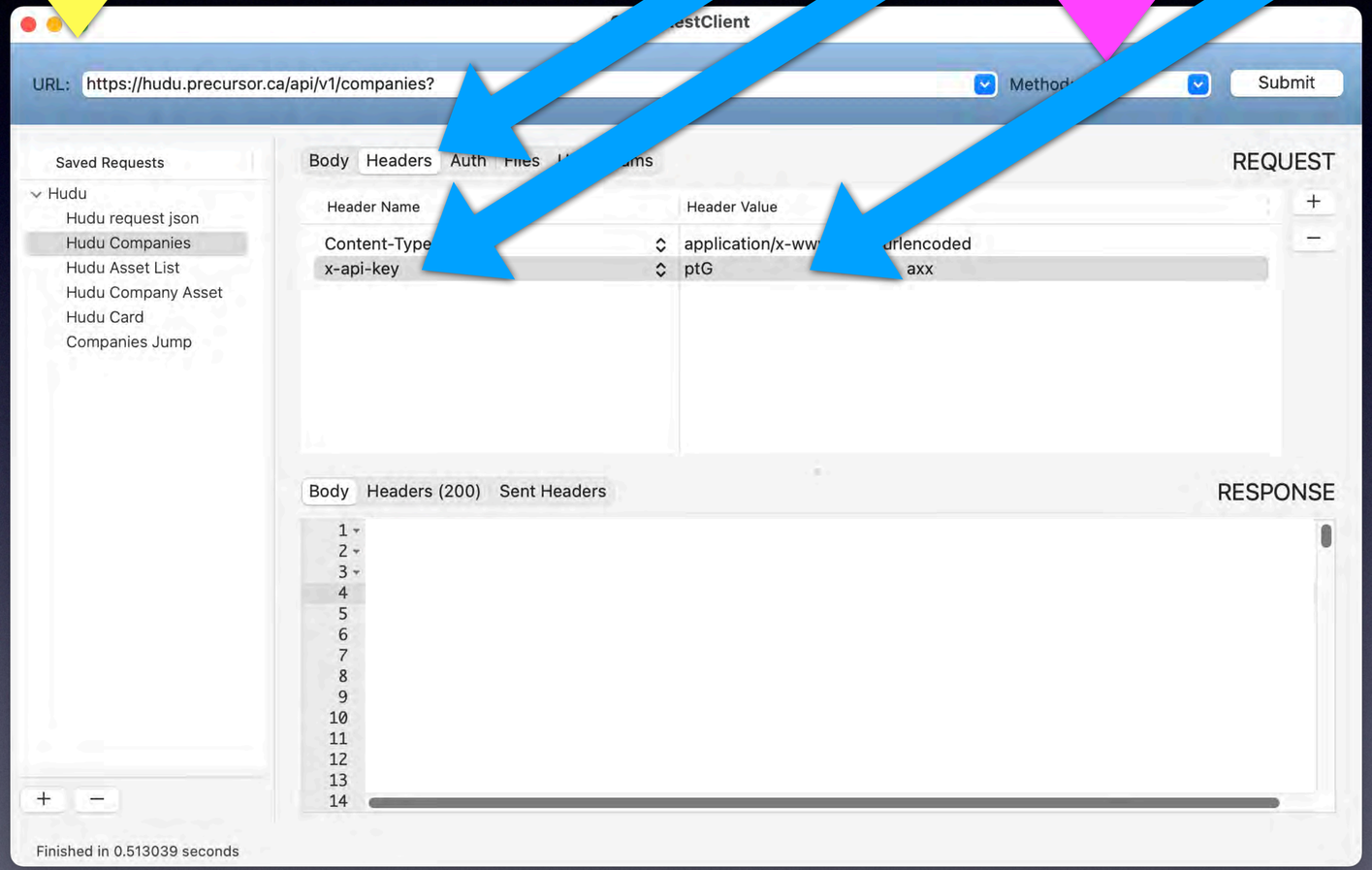
```
curl https://[YOUR_DOMAIN]/api/v1/companies? -H "x-api-key: YOUR_API_KEY_HERE"
```



precursor.ca/slides

```
curl https://[YOUR_DOMAIN]/api/v1/companies? -H "x-api-key: YOUR_API_KEY_HERE"
```

GET



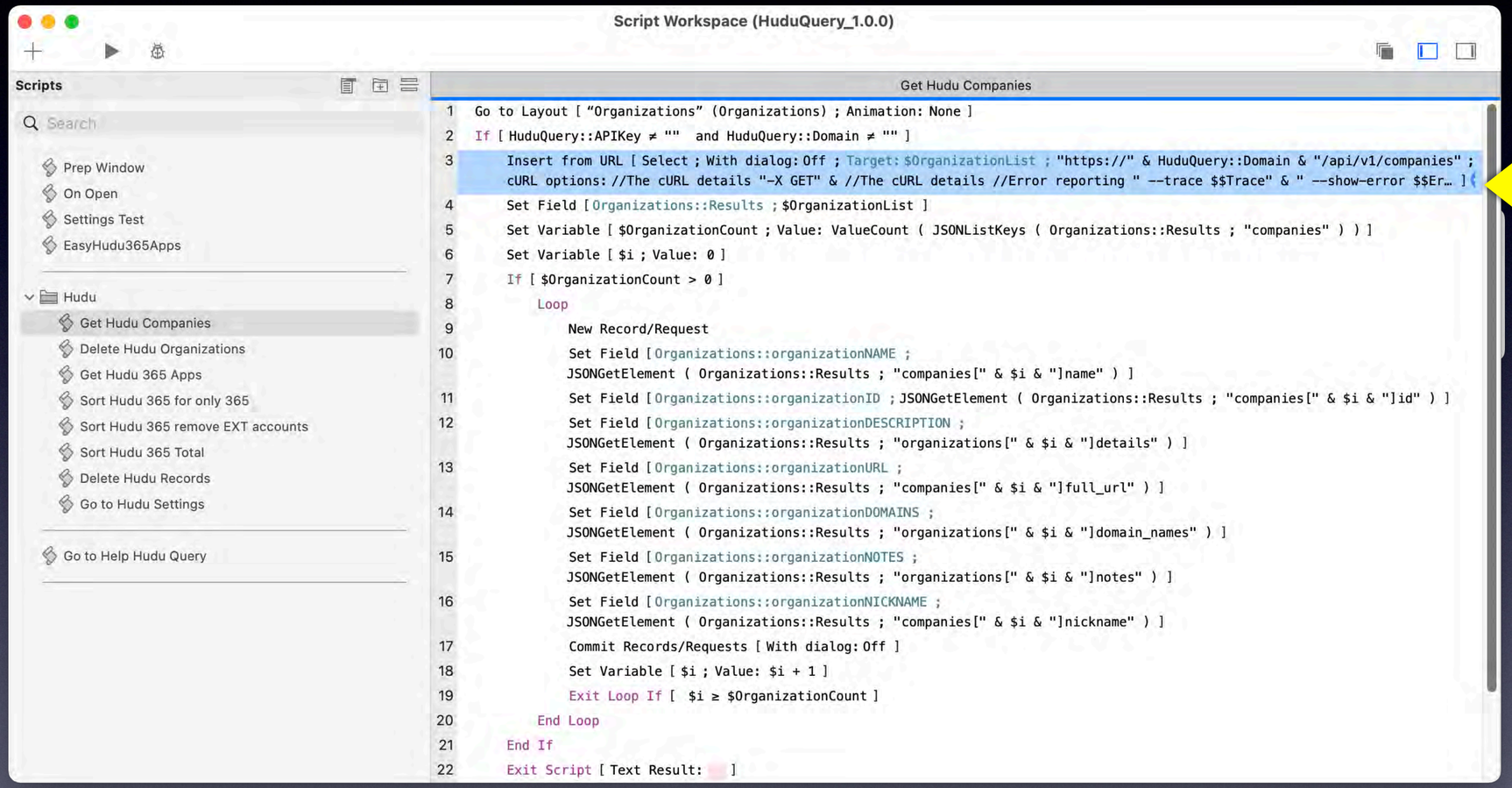
precursor.ca/slides

```
curl https://[YOUR_DOMAIN]/api/v1/companies? -H "x-api-key: YOUR_API_KEY_HERE"
```

Insert From URL script step

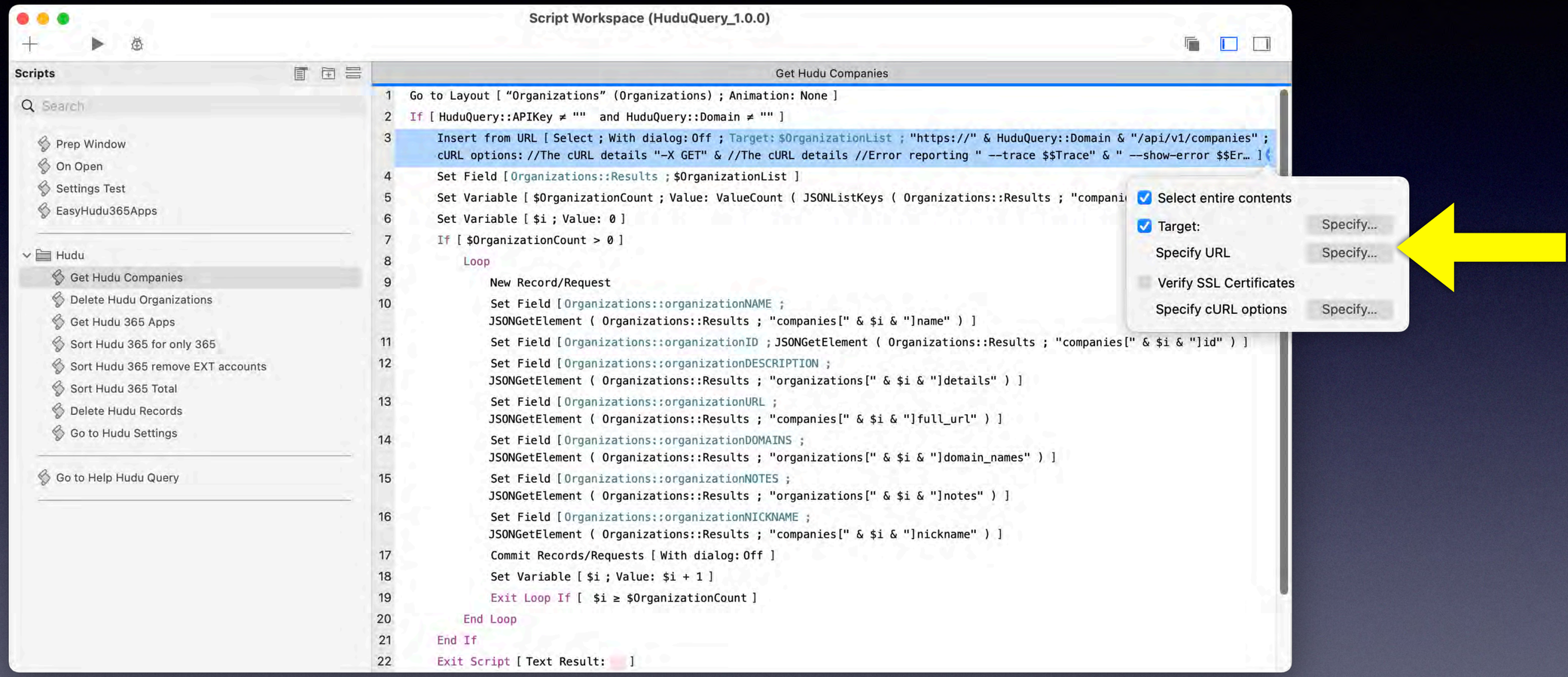
precursor.ca/slides

```
https://[YOUR_DOMAIN]/api/v1/companies? -H "x-api-key: YOUR_API_KEY_HERE"
```

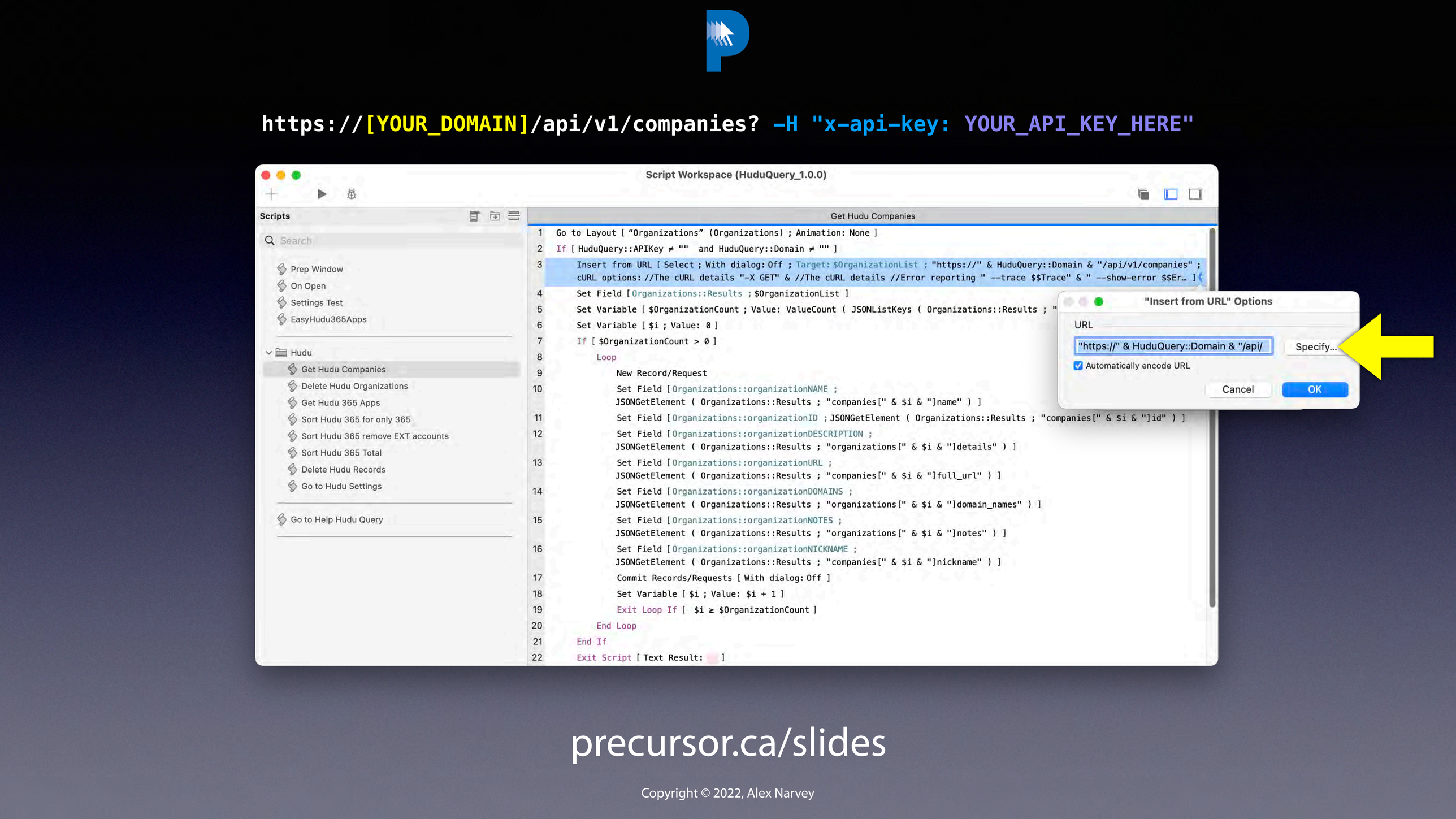


precursor.ca/slides

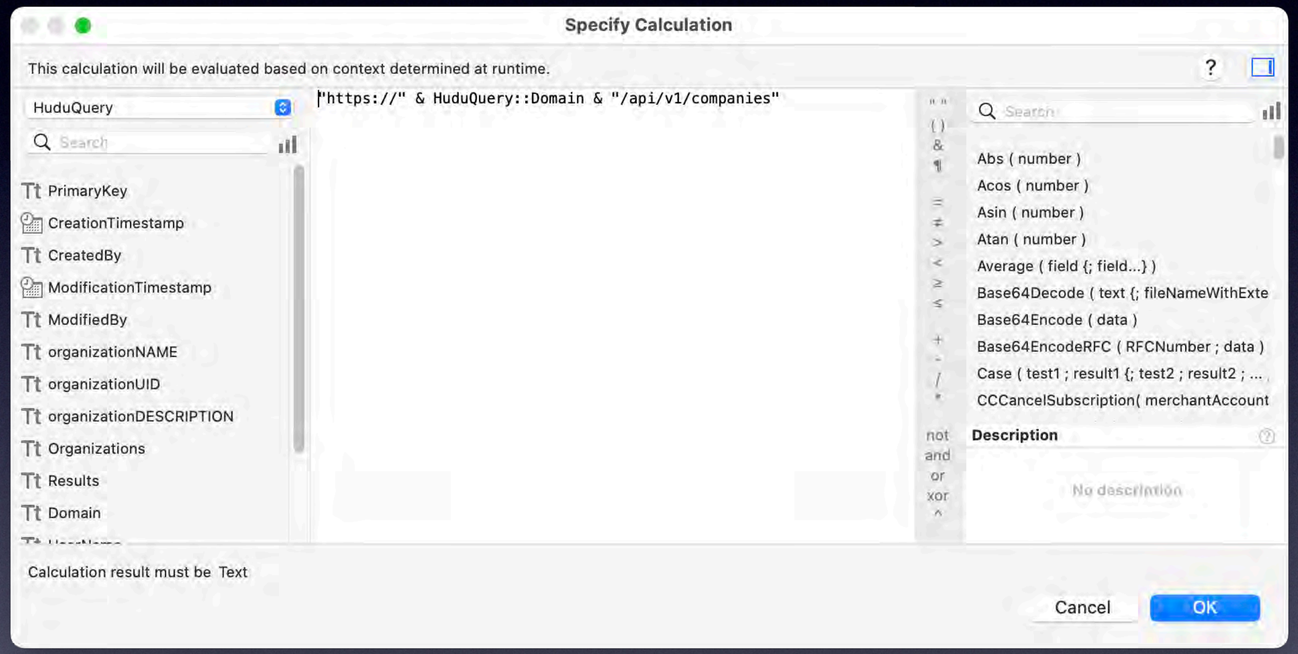
```
https://[YOUR_DOMAIN]/api/v1/companies? -H "x-api-key: YOUR_API_KEY_HERE"
```



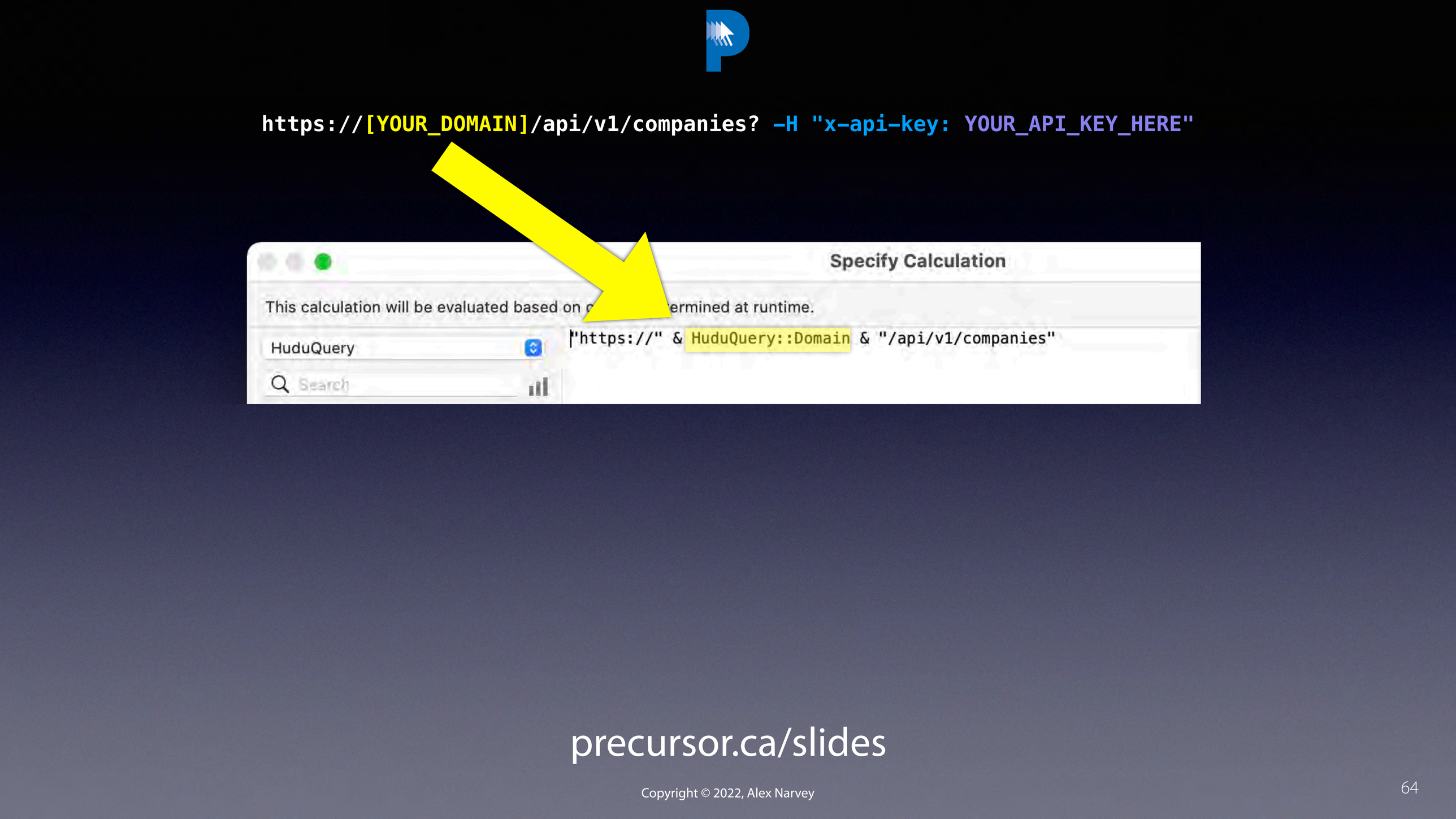
precursor.ca/slides



```
https://[YOUR_DOMAIN]/api/v1/companies? -H "x-api-key: YOUR_API_KEY_HERE"
```

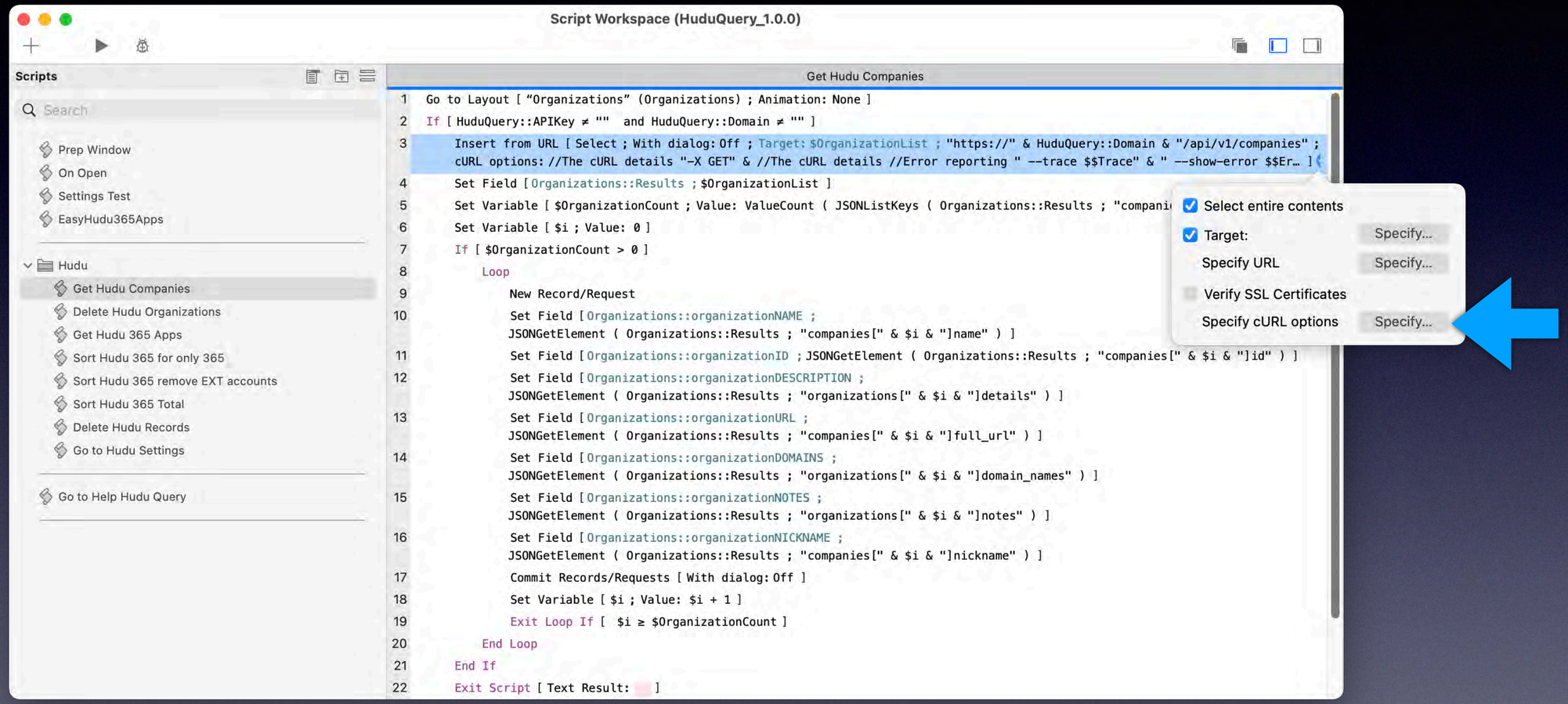


precursor.ca/slides



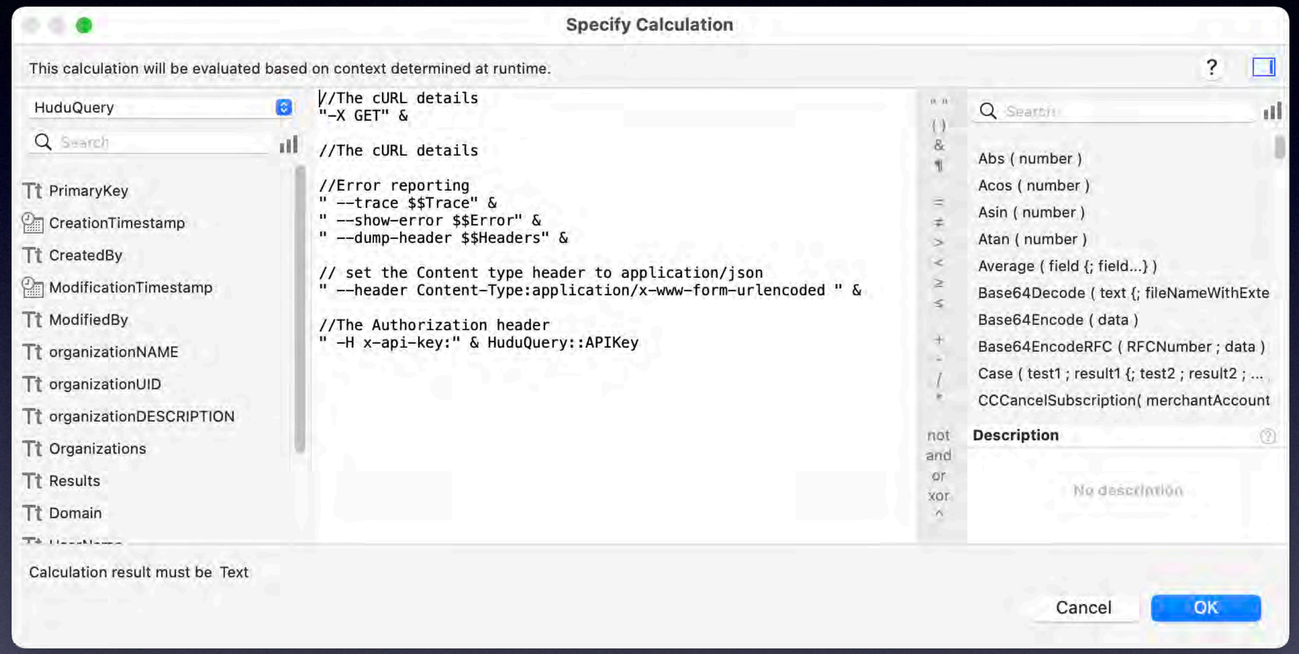
precursor.ca/slides

```
https://[YOUR_DOMAIN]/api/v1/companies? -H "x-api-key: YOUR_API_KEY_HERE"
```



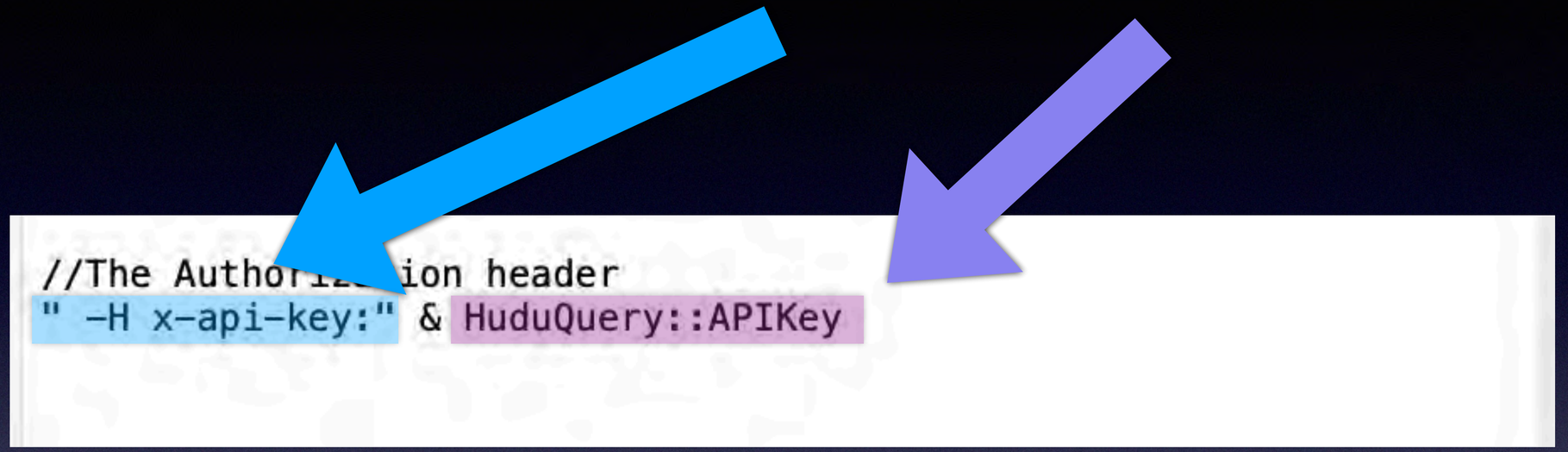
precursor.ca/slides

```
https://[YOUR_DOMAIN]/api/v1/companies? -H "x-api-key: YOUR_API_KEY_HERE"
```



precursor.ca/slides

```
https://[YOUR_DOMAIN]/api/v1/companies? -H "x-api-key: YOUR_API_KEY_HERE"
```



precursor.ca/slides

Data in URL

precursor.ca/slides

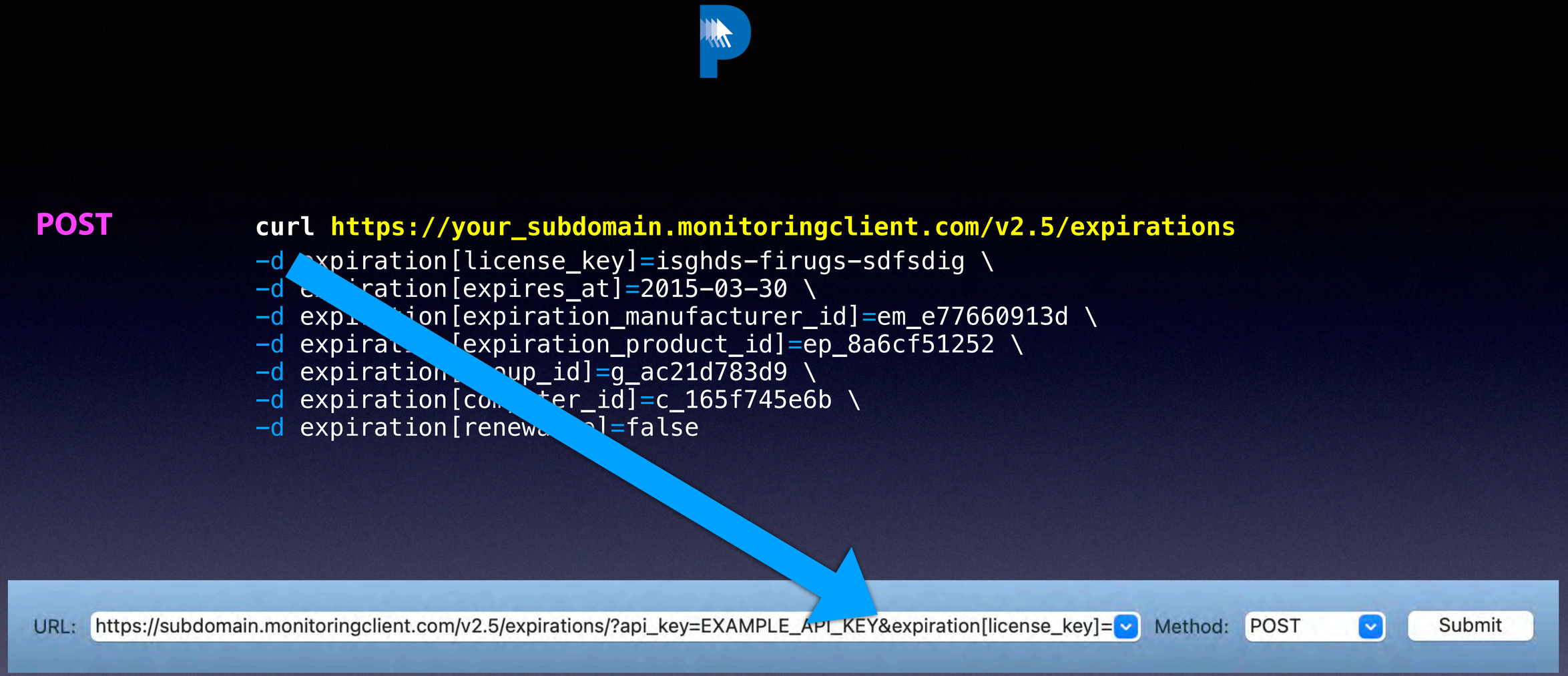
POST `curl https://your_subdomain.monitoringclient.com/v2.5/expirations`

precursor.ca/slides

POST

```
curl https://your_subdomain.monitoringclient.com/v2.5/expirations
-d expiration[license_key]=isghds-firugs-sdfsdig \
-d expiration[expires_at]=2015-03-30 \
-d expiration[expiration_manufacturer_id]=em_e77660913d \
-d expiration[expiration_product_id]=ep_8a6cf51252 \
-d expiration[group_id]=g_ac21d783d9 \
-d expiration[computer_id]=c_165f745e6b \
-d expiration[renewable]=false
```

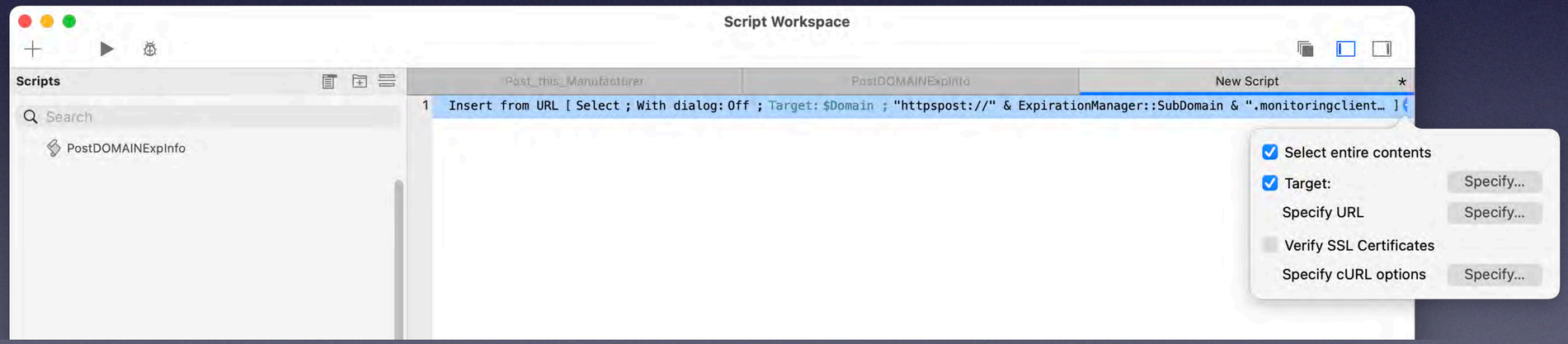
precursor.ca/slides



precursor.ca/slides

POST

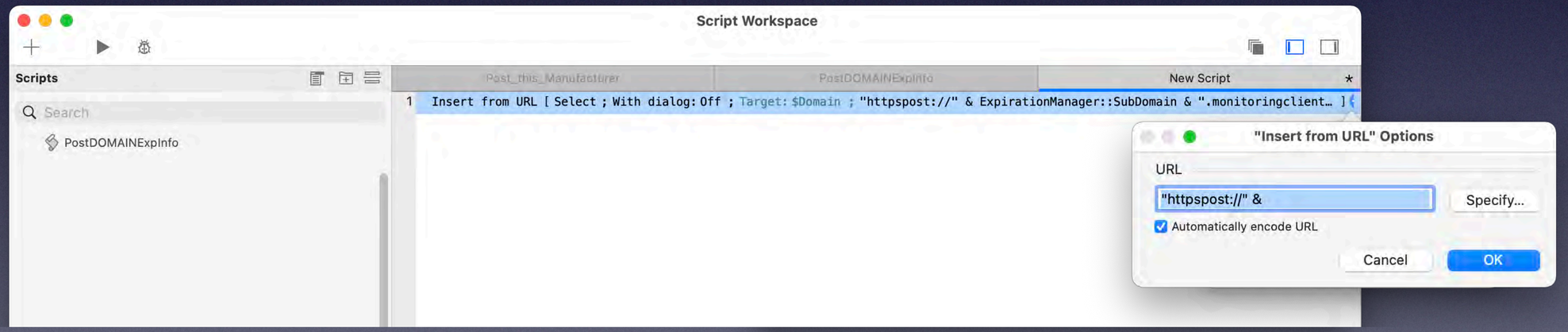
```
curl https://your_subdomain.monitoringclient.com/v2.5/expirations
-d expiration[license_key]=isghds-firugs-sdfsdig \
-d expiration[expires_at]=2015-03-30 \
-d expiration[expiration_manufacturer_id]=em_e77660913d \
-d expiration[expiration_product_id]=ep_8a6cf51252 \
-d expiration[group_id]=g_ac21d783d9 \
-d expiration[computer_id]=c_165f745e6b \
-d expiration[renewable]=false
```



precursor.ca/slides

POST

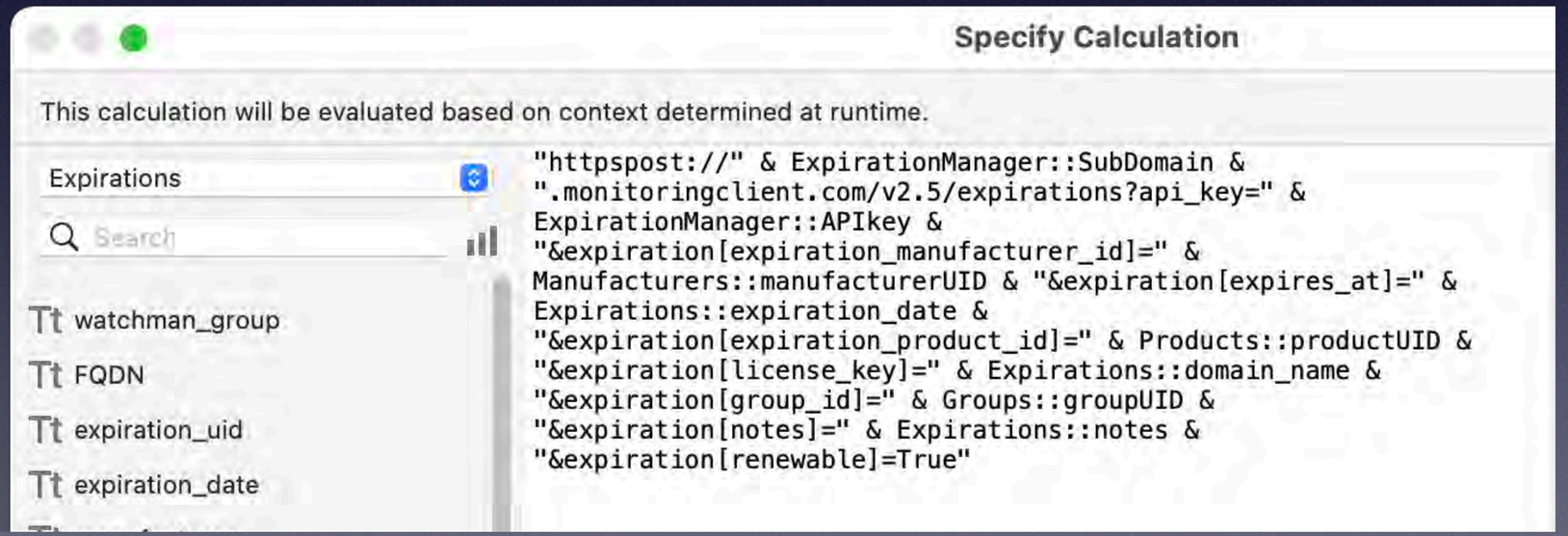
```
curl https://your_subdomain.monitoringclient.com/v2.5/expirations
-d expiration[license_key]=isghds-firugs-sdfsdig \
-d expiration[expires_at]=2015-03-30 \
-d expiration[expiration_manufacturer_id]=em_e77660913d \
-d expiration[expiration_product_id]=ep_8a6cf51252 \
-d expiration[group_id]=g_ac21d783d9 \
-d expiration[computer_id]=c_165f745e6b \
-d expiration[renewable]=false
```



precursor.ca/slides

POST

```
curl https://your_subdomain.monitoringclient.com/v2.5/expirations
-d expiration[license_key]=isghds-firugs-sdfsdig \
-d expiration[expires_at]=2015-03-30 \
-d expiration[expiration_manufacturer_id]=em_e77660913d \
-d expiration[expiration_product_id]=ep_8a6cf51252 \
-d expiration[group_id]=g_ac21d783d9 \
-d expiration[computer_id]=c_165f745e6b \
-d expiration[renewable]=false
```



precursor.ca/slides

```
“text” & table::fielddata & $variable & “&key=data”
```

precursor.ca/slides

FileMaker Ampersand

```
“text” & table::fielddata & $variable & “&key=data”
```

precursor.ca/slides

URL Ampersand

```
“text” & table::fielddata & $variable & “&key=data”
```

precursor.ca/slides

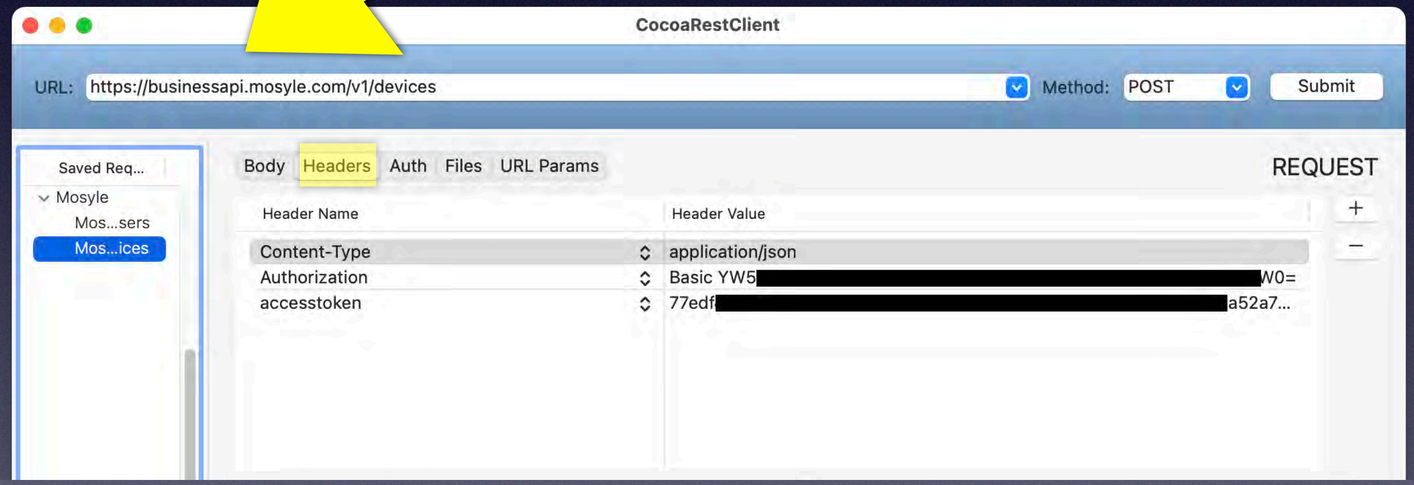
Data sent by JSON

precursor.ca/slides

```
POST  curl http://businessapi.mosyle.com/v1/devices

        {
            "operation": "list",
            "options" : { "os": "mac"}

        }
```

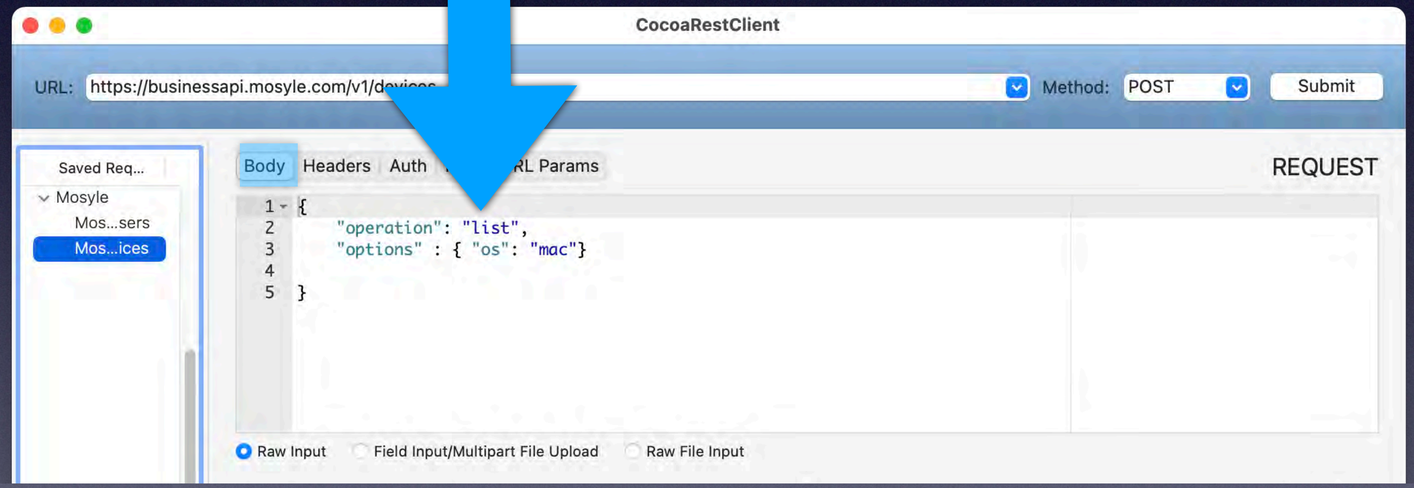
precursor.ca/slides

```
POST  curl http://businessapi.mosyle.com/v1/devices

      {
          "operati     "list",
          "optio     : { "os": "mac"}

      }
```



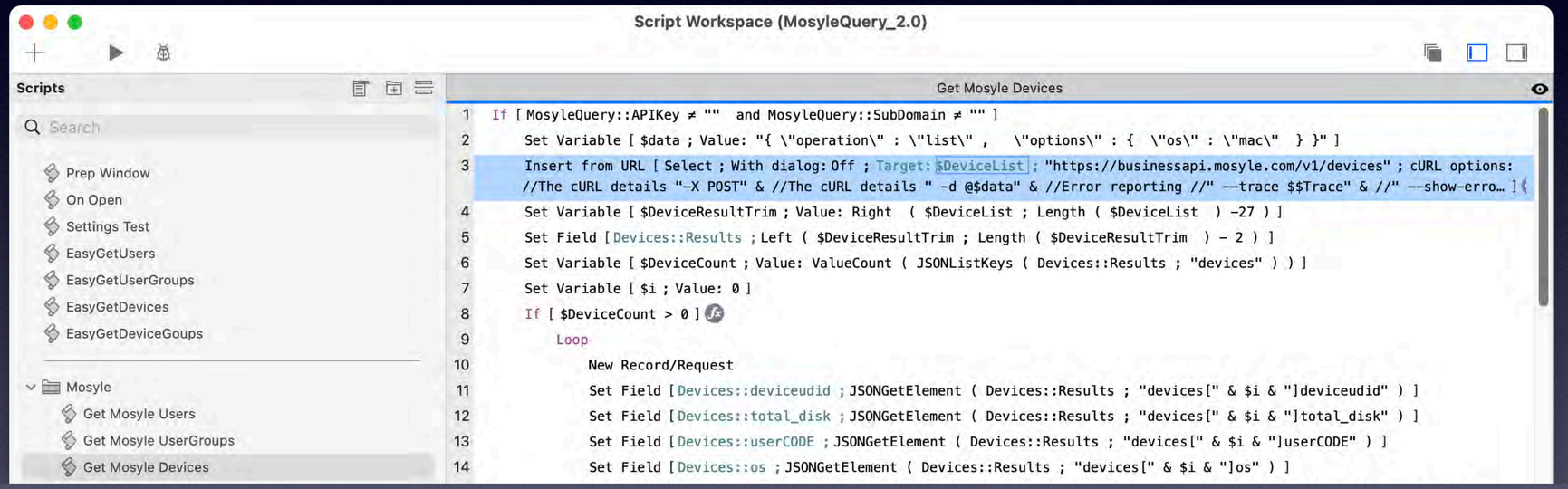
precursor.ca/slides

```
POST curl http://businessapi.mosyle.com/v1/devices

        {
            "operation": "list",
            "options" : { "os": "mac"}

        }
```



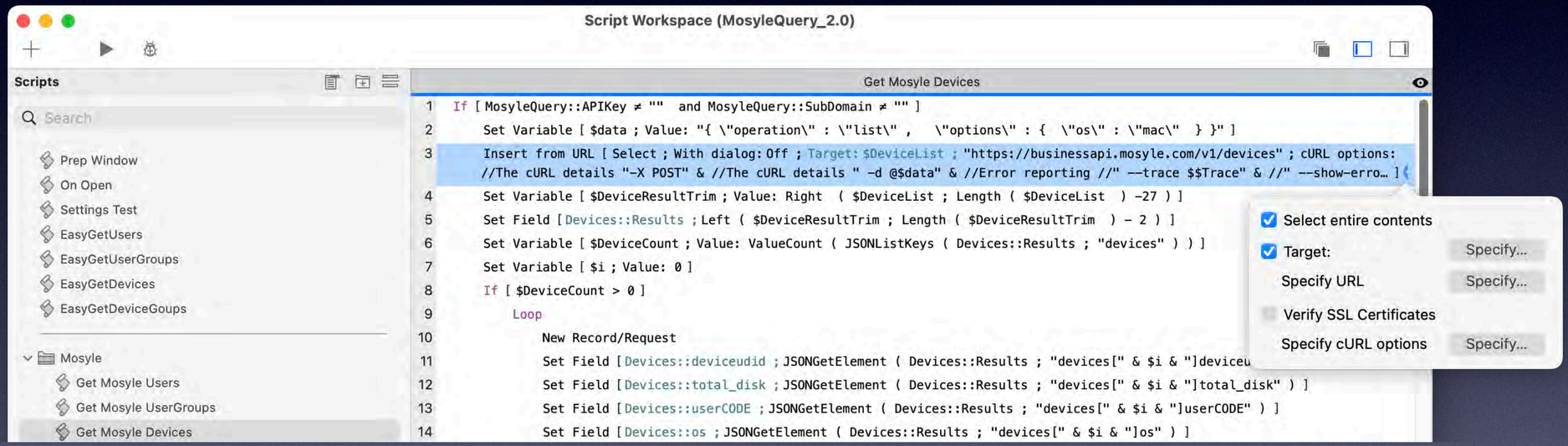
precursor.ca/slides

```
POST  curl http://businessapi.mosyle.com/v1/devices
      {
          "operation": "list",
          "options" : { "os": "mac"}

      }
```



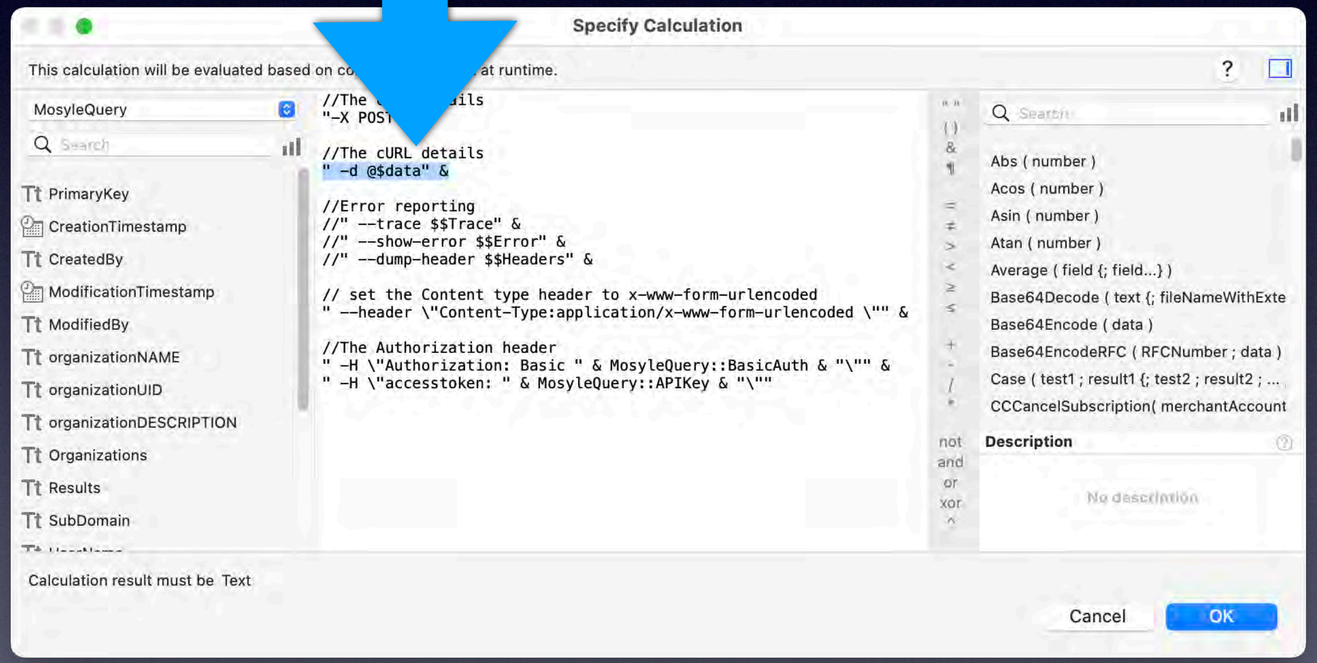
precursor.ca/slides

```
POST  curl http://businessapi.mosyle.com/v1/devices
      {
          "operation": "list",
          "options" : { "os": "mac"}

      }
```



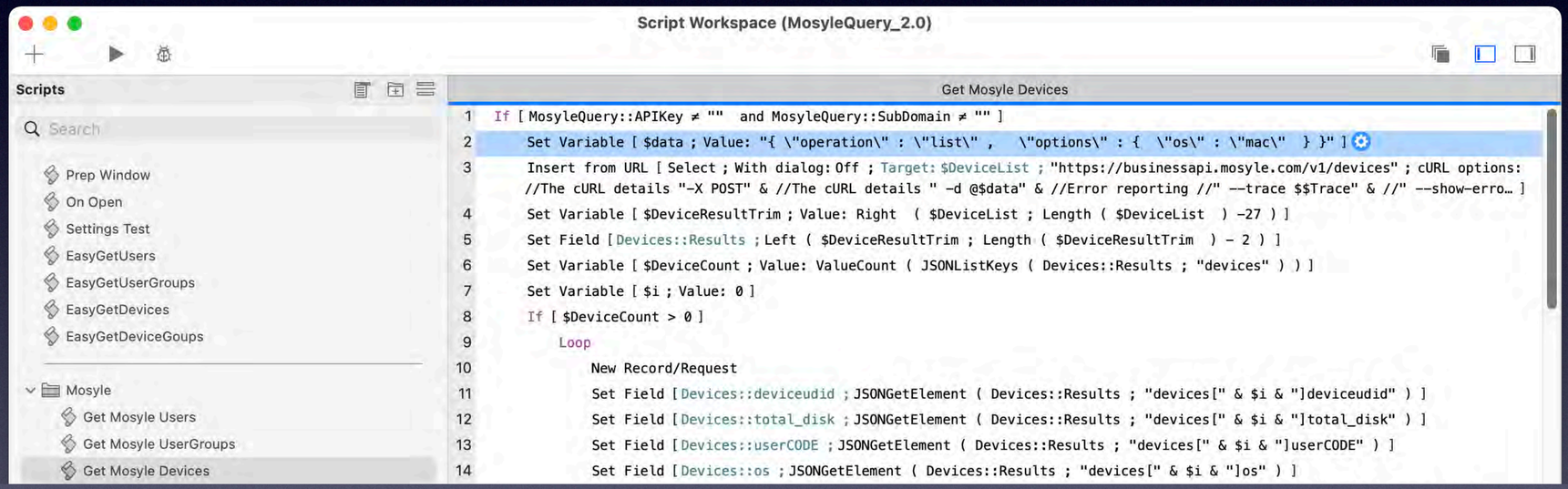
precursor.ca/slides

```
POST curl http://businessapi.mosyle.com/v1/devices
     {
         "operation": "list",
         "options" : { "os": "mac"}

     }
```



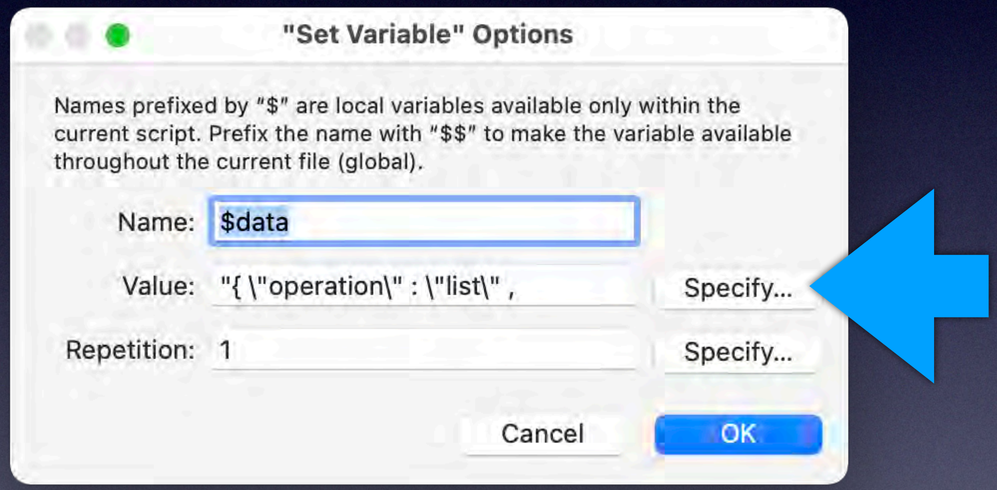
precursor.ca/slides

```
POST curl http://businessapi.mosyle.com/v1/devices
     {
         "operation": "list",
         "options" : { "os": "mac"}

     }
```



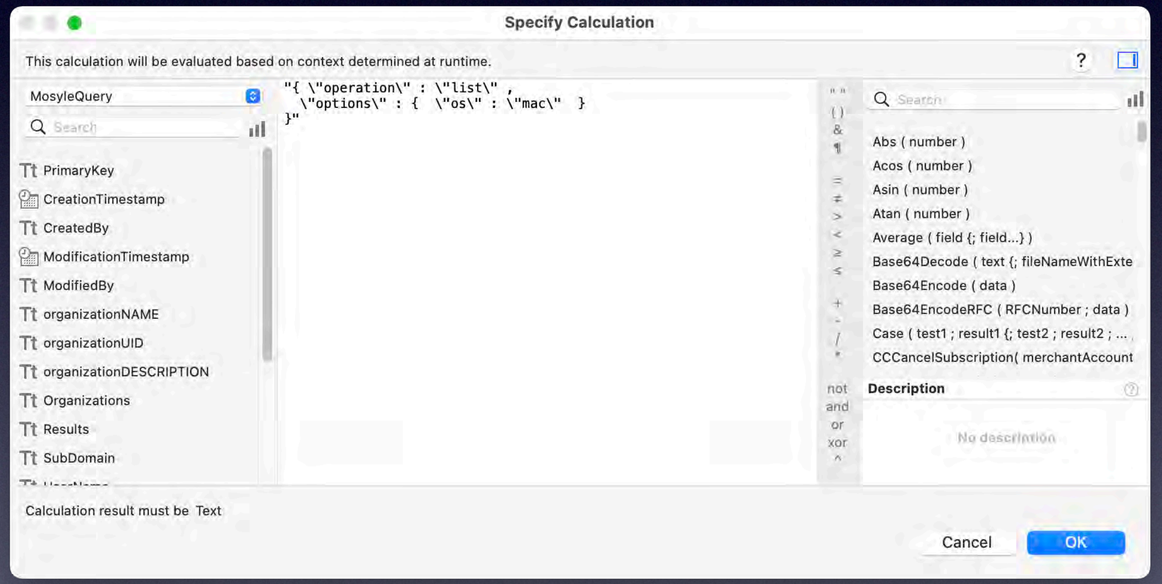
precursor.ca/slides

```
POST  curl http://businessapi.mosyle.com/v1/devices
      {
          "operation": "list",
          "options" : { "os": "mac"}

      }
```



precursor.ca/slides

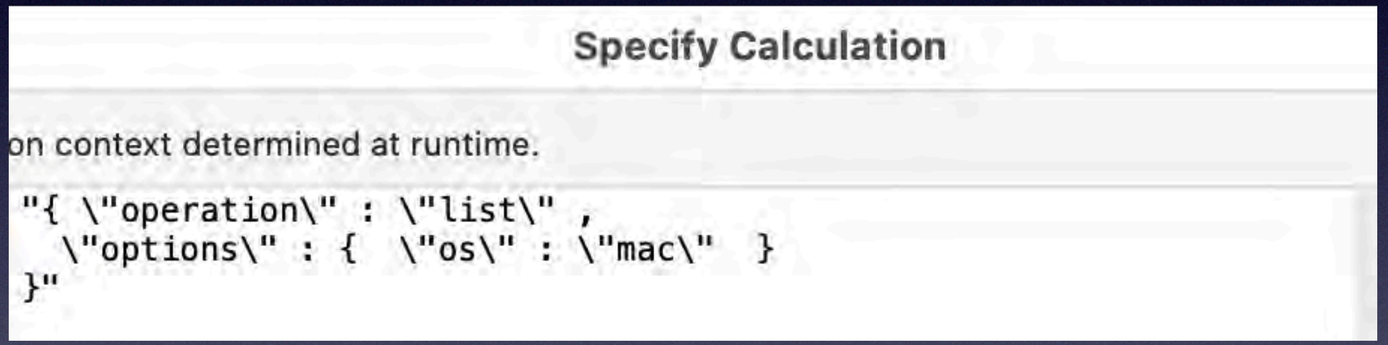
```
POST curl http://businessapi.mosyle.com/v1/devices
        {
            "operation": "list",
            "options" : { "os": "mac"}

        }
```



precursor.ca/slides

POST `curl http://businessapi.mosyle.com/v1/devices`

```
{
    "operation": "list",
    "options" : { "os": "mac"}

}
```



precursor.ca/slides

-H / --header <name: value>	Use this option multiple times to set multiple header entries.
-F / --form <name=content>	// form data (quoted?)
-F / --form <name=@[$]fmvariable>	// form data value from variable
-b / --cookie-jar <$[$]fmvariable>	Direct file access replaced with FM variables.
-d / --data <data>	Data (quoted)
-d / --data @<$[$]fmvariable>	Data from FileMaker variable (@ character prefixing a FM variable replaces direct file access)

JSON Parsing

FileMaker's JSON Parsing tools

precursor.ca/slides

JSON functions

JSON functions parse, create, modify, and format data in JSON format. See Working with the JSON functions.

This function	Returns
JSONDeleteElement	Deletes a JSON data element specified by an object name, an array index, or a path.
JSONFormatElements	Formats elements in JSON data to make them easier to read.
JSONGetElement	Queries JSON data for an element specified by an object name, an array index, or a path.
JSONListKeys	Lists the object names (keys) or array indexes in JSON data for an element specified by an object name, an array index, or a path.
JSONListValues	Lists the values in JSON data for an element specified by an object name, an array index, or a path.
JSONSetElement	Adds or modifies an element in JSON data specified by an object name, an array index, or a path.

precursor.ca/slides

`JSONFormatElements`

precursor.ca/slides


```
{"firstName":"John","
lastName":"doe","age"
:"26","address":
{"streetAddress":"123
Main street”,”
city":"Anytown","post
alCode":"12345"},"pho
neNumbers":
[{"type":"iPhone","nu
mber":"123-456-8888"}
,
{"type":"home","numbe
r":"123-557-8910"}]}
```


precursor.ca/slides

JSONFormatElements

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : "26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

precursor.ca/slides

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : "26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

https://www.soliantconsulting.com/blog/parsing-json-filemaker-16/

precursor.ca/slides

Keys

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : "26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

precursor.ca/slides

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : “26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

Values

precursor.ca/slides

Arrays

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : “26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

precursor.ca/slides

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : “26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

```
JSONGetElement( JSON ; "firstName" )
```

```
"John"
```

precursor.ca/slides

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : “26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

```
JSONGetElement( JSON ; "address" )
```

```
{
    "streetAddress": "123 Main street",
    "city"          : "Anytown",
    "postalCode"    : "12345"
}
```

precursor.ca/slides

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : “26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

```
JSONGetElement( JSON ; "address.streetAddress" )
```

```
"123 Main Street"
```

precursor.ca/slides

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : “26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

```
JSONGetElement( JSON ; "phoneNumbers" )

    [
        {
          "type"  : "iPhone",
          "number": "123-456-8888"
        },
        {
          "type"  : "home",
          "number": "123-557-8910"
        }
    ]
```

precursor.ca/slides

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : “26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

```
JSONGetElement( JSON ; "phoneNumbers[0]" )
```

```
{"number":"123-456-8888","type":"iPhone"}
```

precursor.ca/slides

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : “26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

```
JSONGetElement( JSON ; "phoneNumbers[0].number" )
```

```
"123-456-8888"
```

precursor.ca/slides

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : “26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

```
JSONGetElement( JSON ; “phoneNumbers[1].number" )
```

“123-557-8910”

precursor.ca/slides

```
{
  "firstName": "John",
  "lastName" : "doe",
  "age"      : “26",
  "address"  : {
    "streetAddress": "123 Main street",
    "city"         : "Anytown",
    "postalCode"   : "12345"
  },
  "phoneNumbers": [
    {
      "type"  : "iPhone",
      "number": "123-456-8888"
    },
    {
      "type"  : "home",
      "number": "123-557-8910"
    }
  ]
}
```

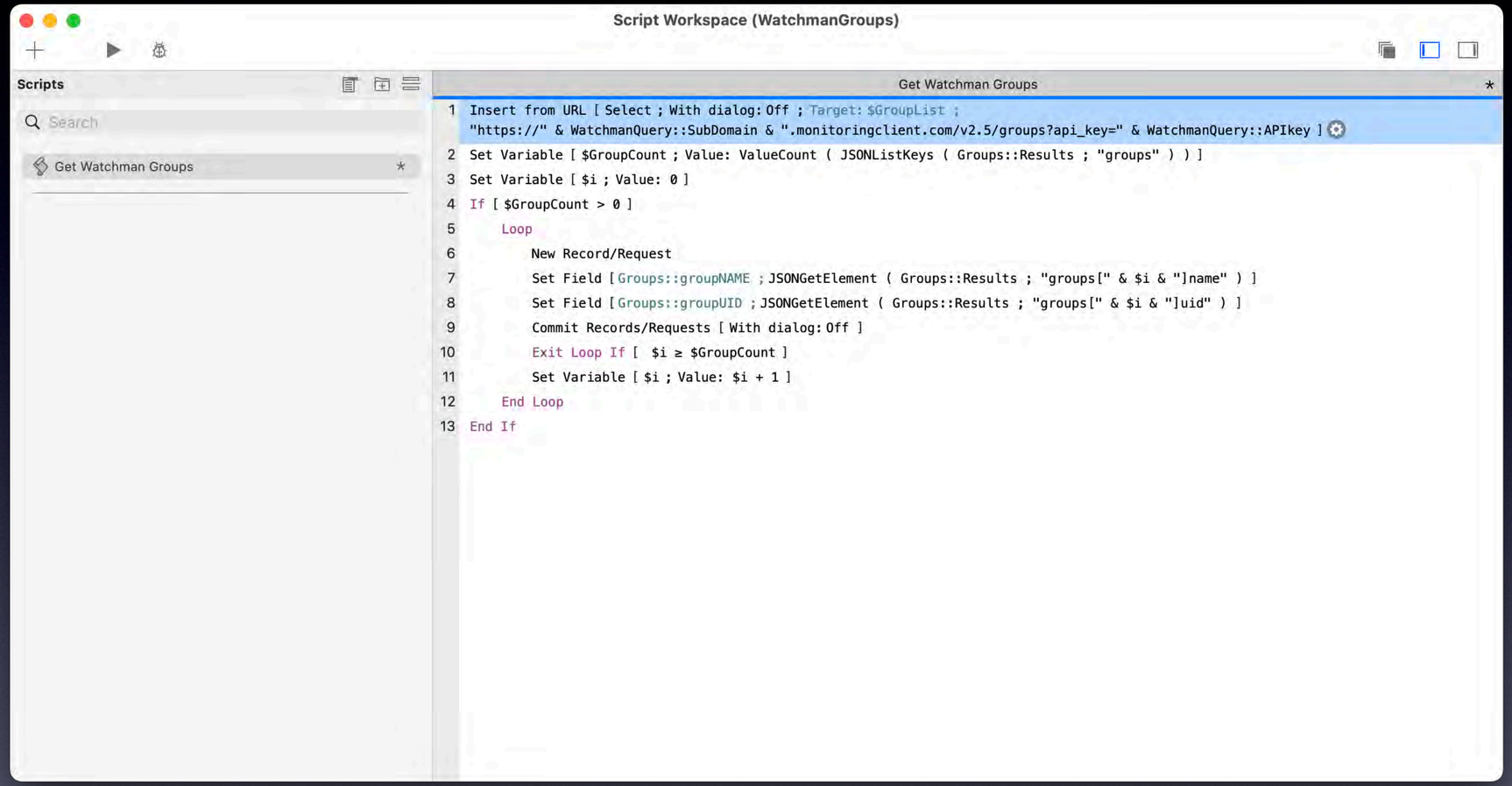
```
ValueCount( JSONListKeys( JSON; "phoneNumbers" )
```

```
"2"
```

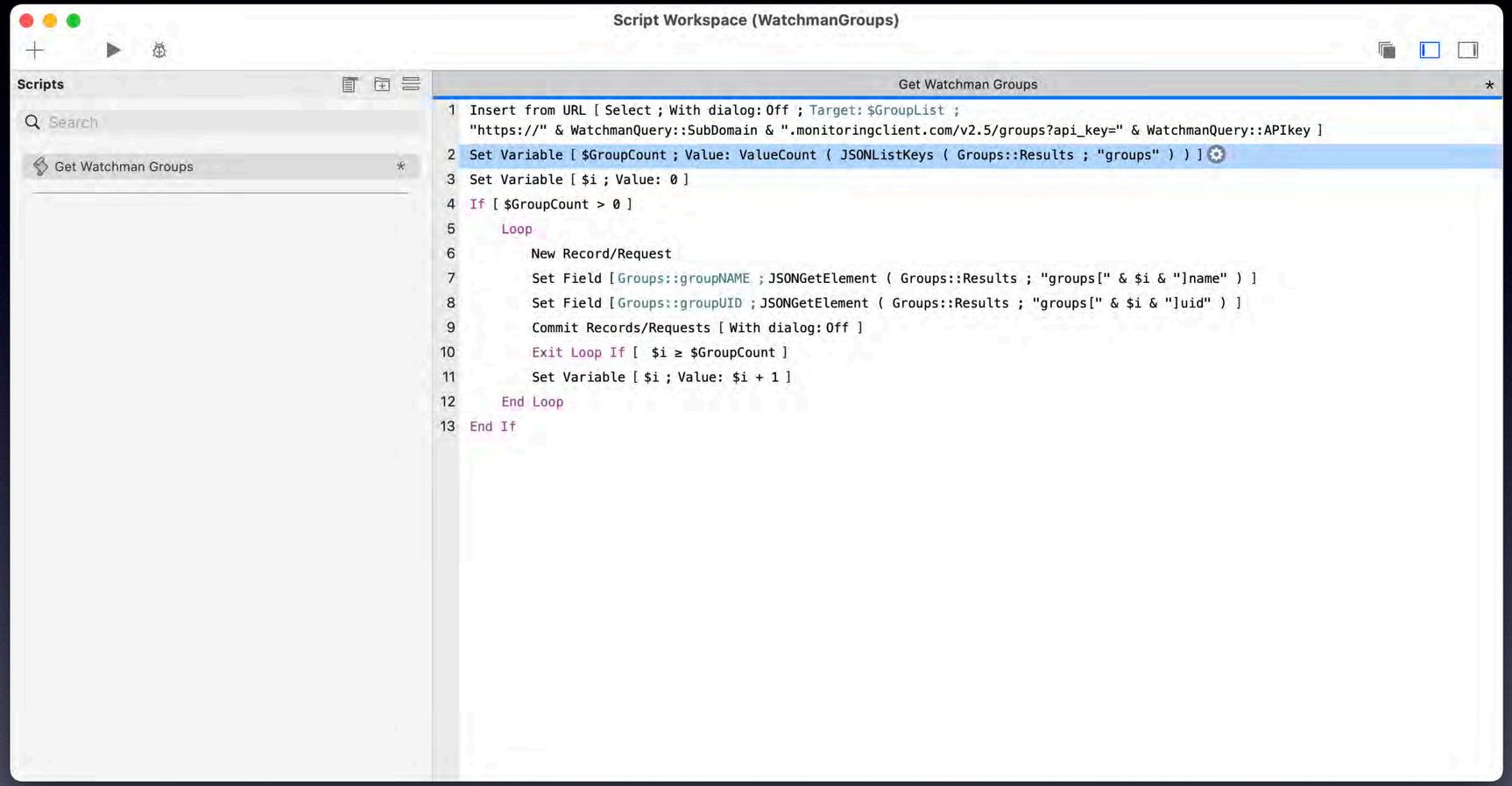
precursor.ca/slides

```
JSONGetElement( JSON ; "phoneNumbers[" & $i & "].number" )
```

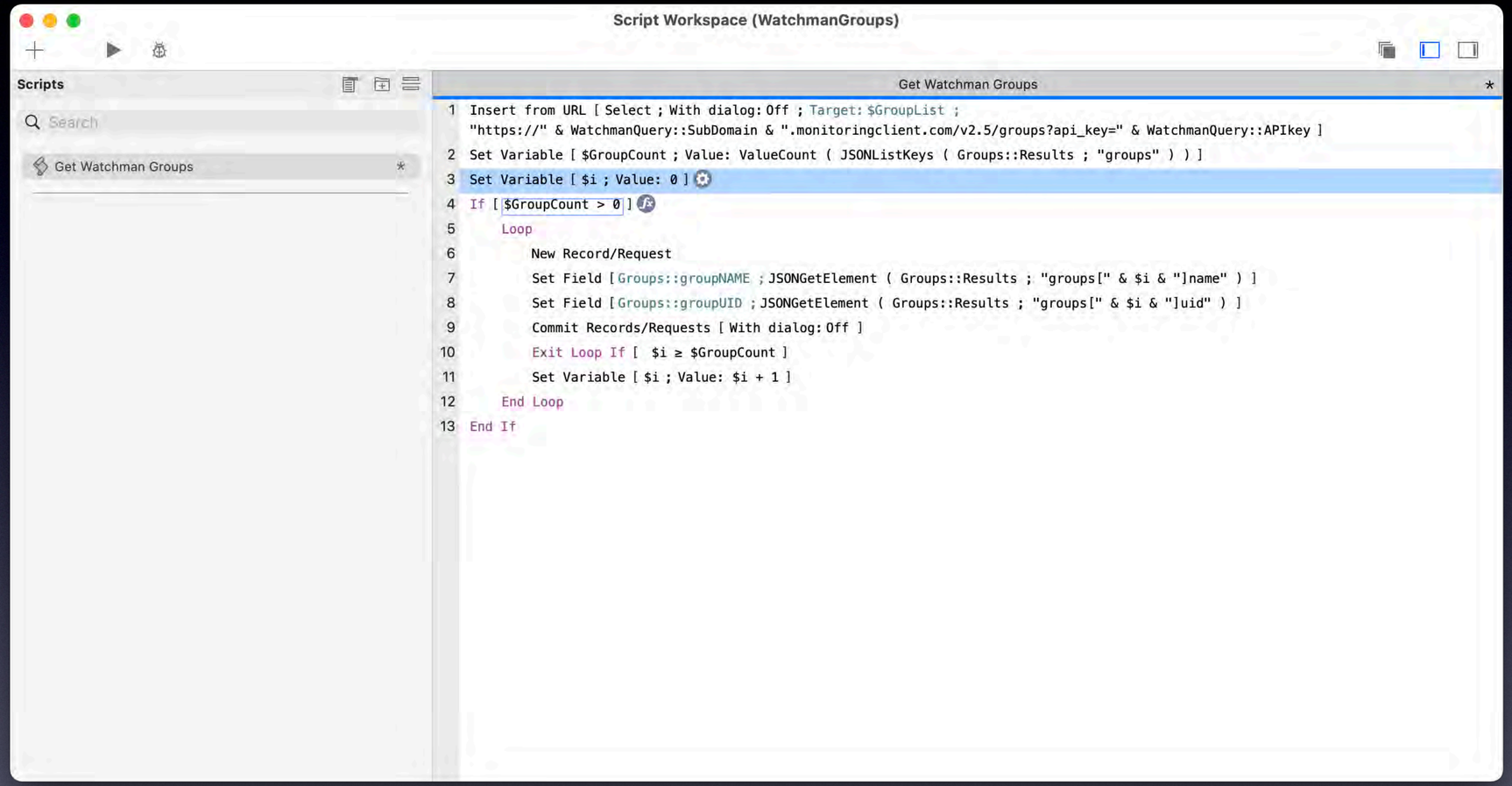
precursor.ca/slides



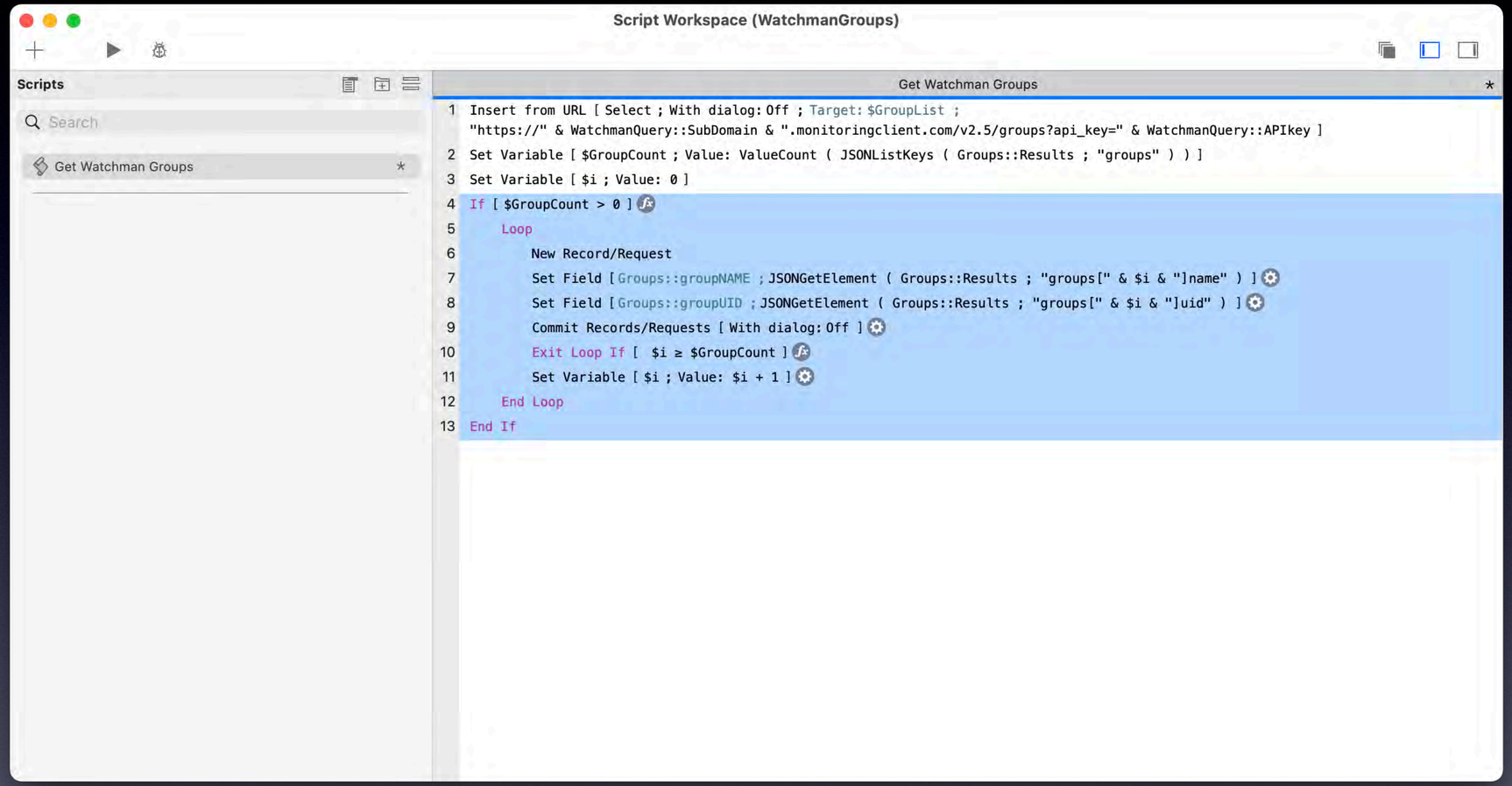
precursor.ca/slides



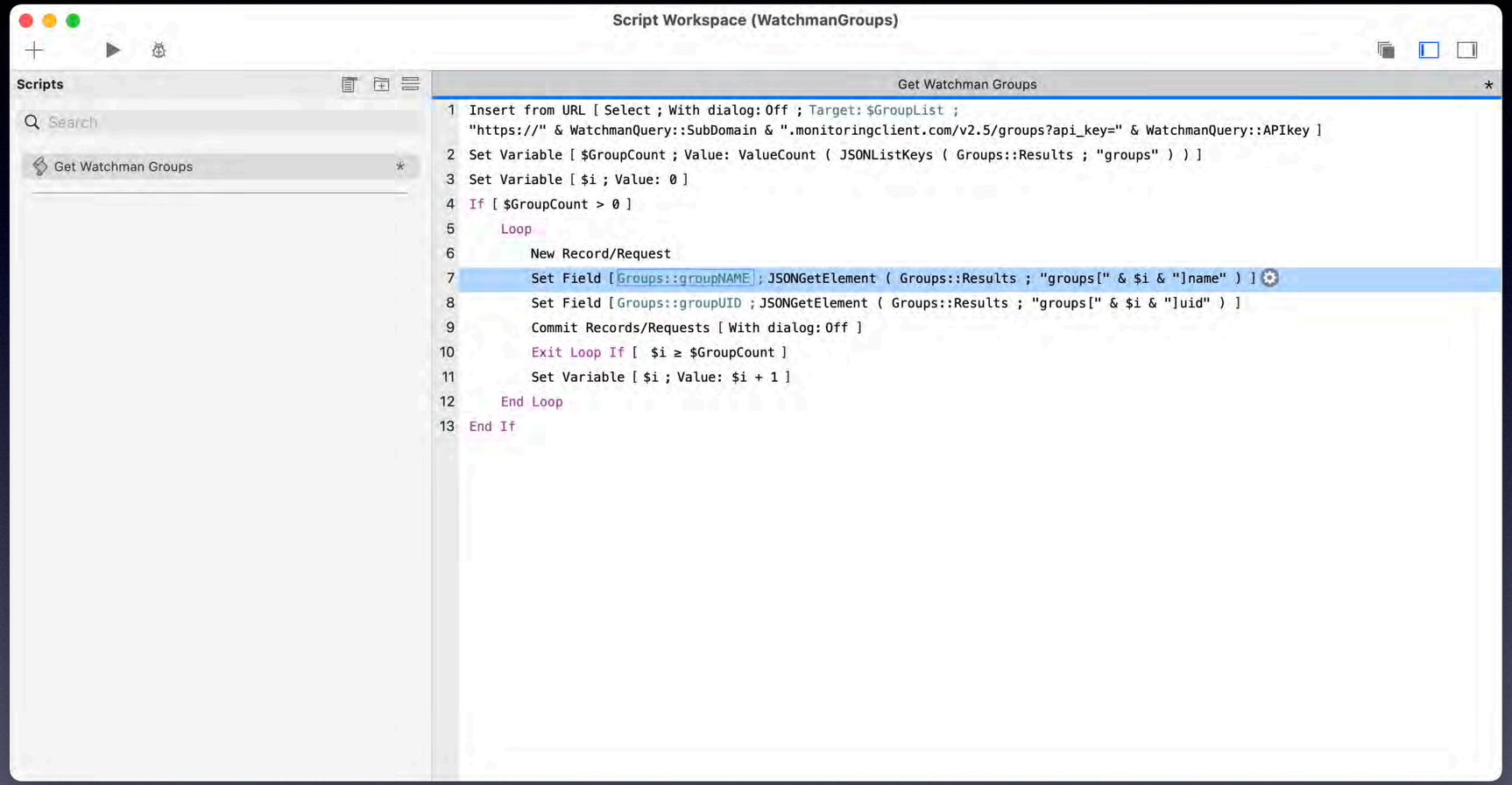
precursor.ca/slides



precursor.ca/slides



precursor.ca/slides



precursor.ca/slides

https://www.soliantconsulting.com/blog/parsing-json-filemaker-16/

https://dbservices.com/articles/filemaker-json-functions/

https://github.com/SoliantMike/PHP-HTTPDebug

precursor.ca/slides

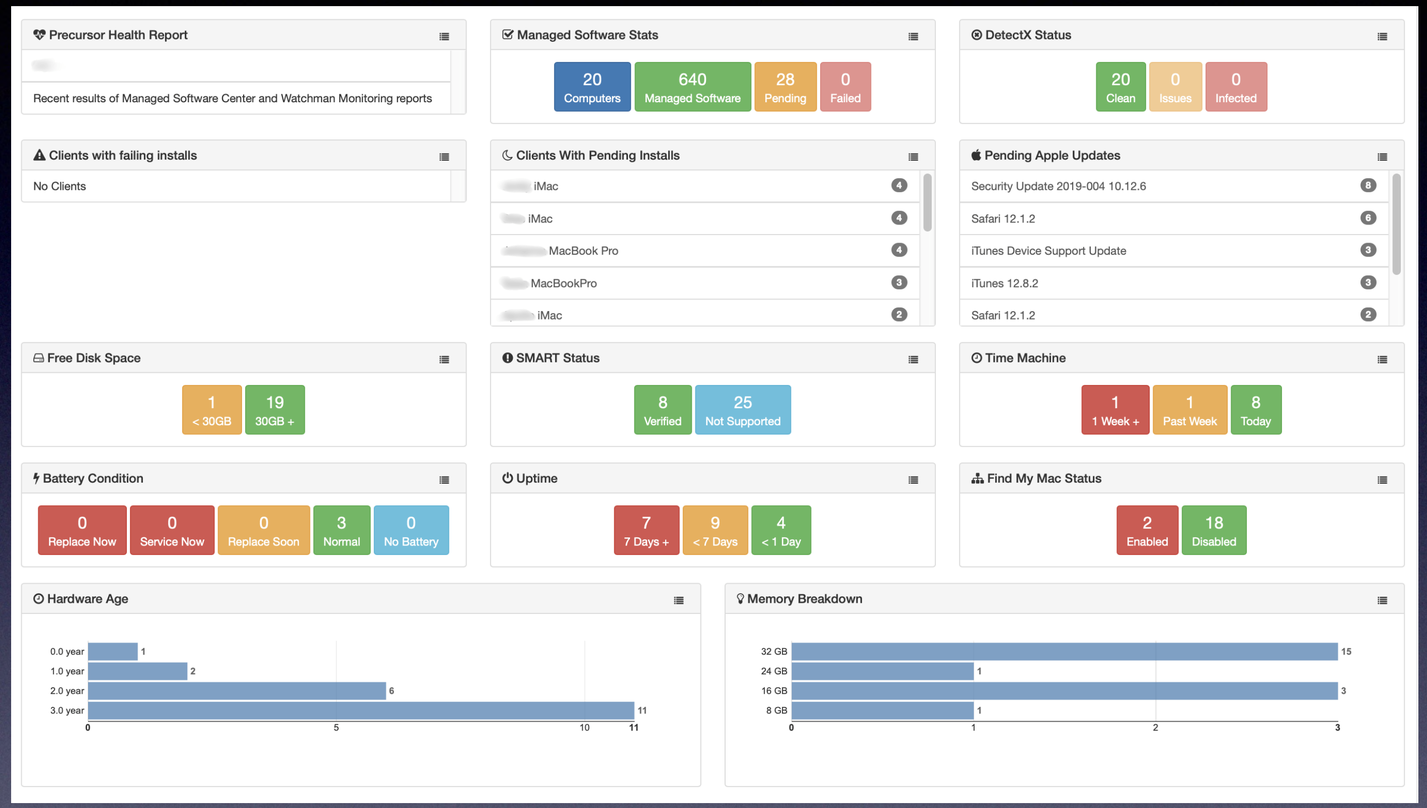
Winnipeg

precursor.ca/slides

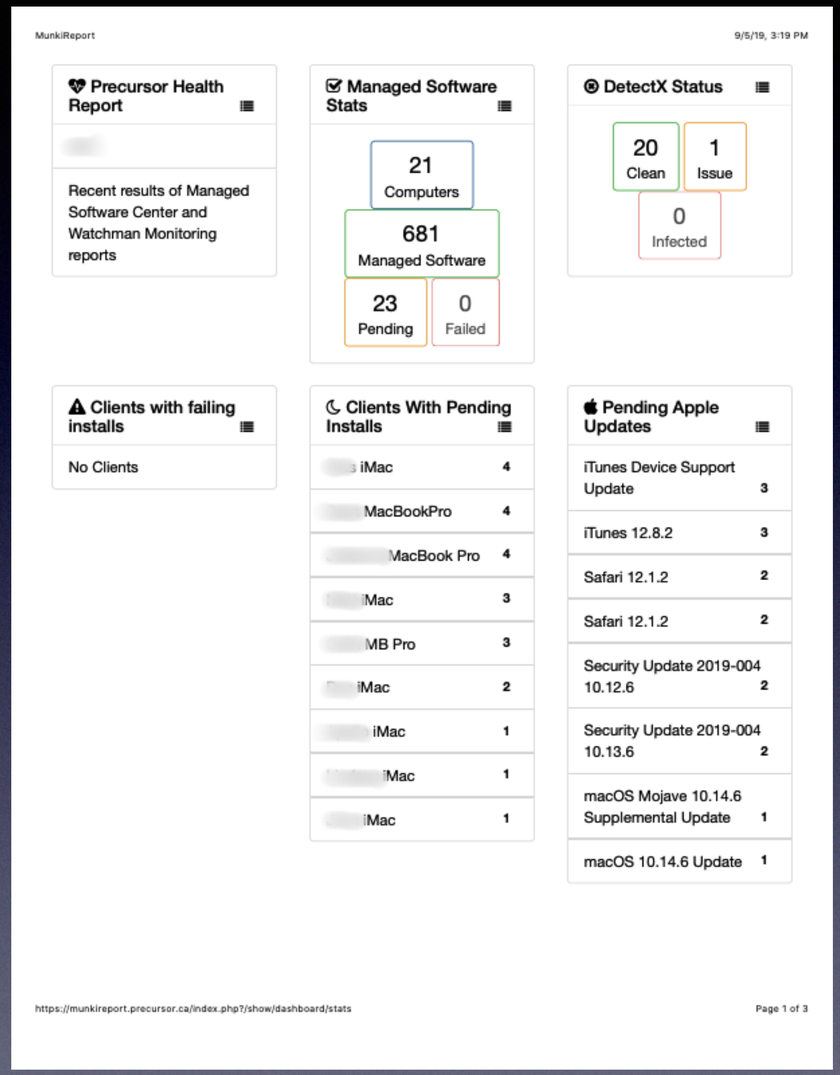

HONEY
RUNS ALSO

Monthly Reports

precursor.ca/slides



precursor.ca/slides



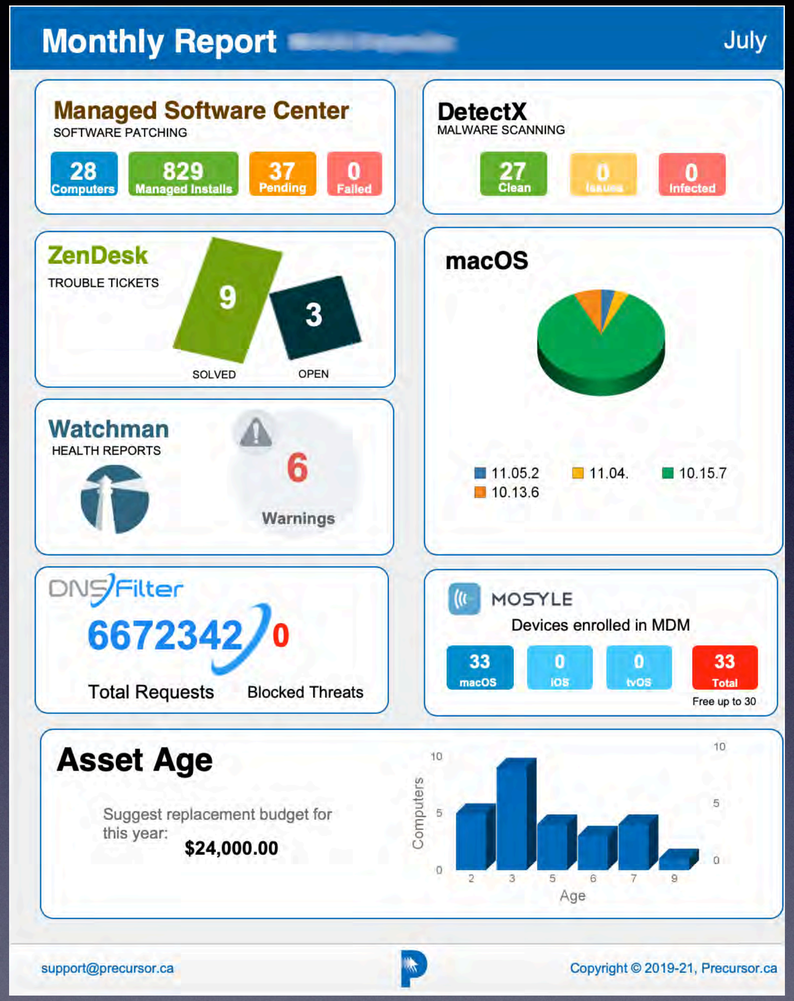
precursor.ca/slides

Each Month!

- **Log in to each client's**
- **Screenshot each client**
- **Rename screenshot**
- **Attach to email**
- **Send mail**

"Like an animal !"

precursor.ca/slides



precursor.ca/slides

Each Month!

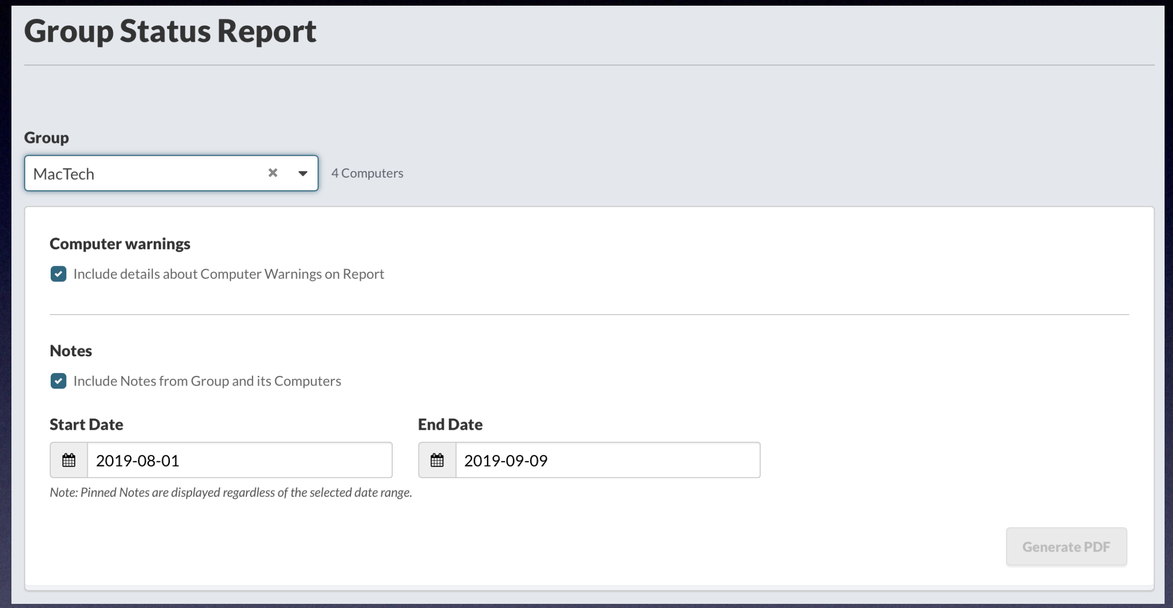
- **Do nothing**

😎

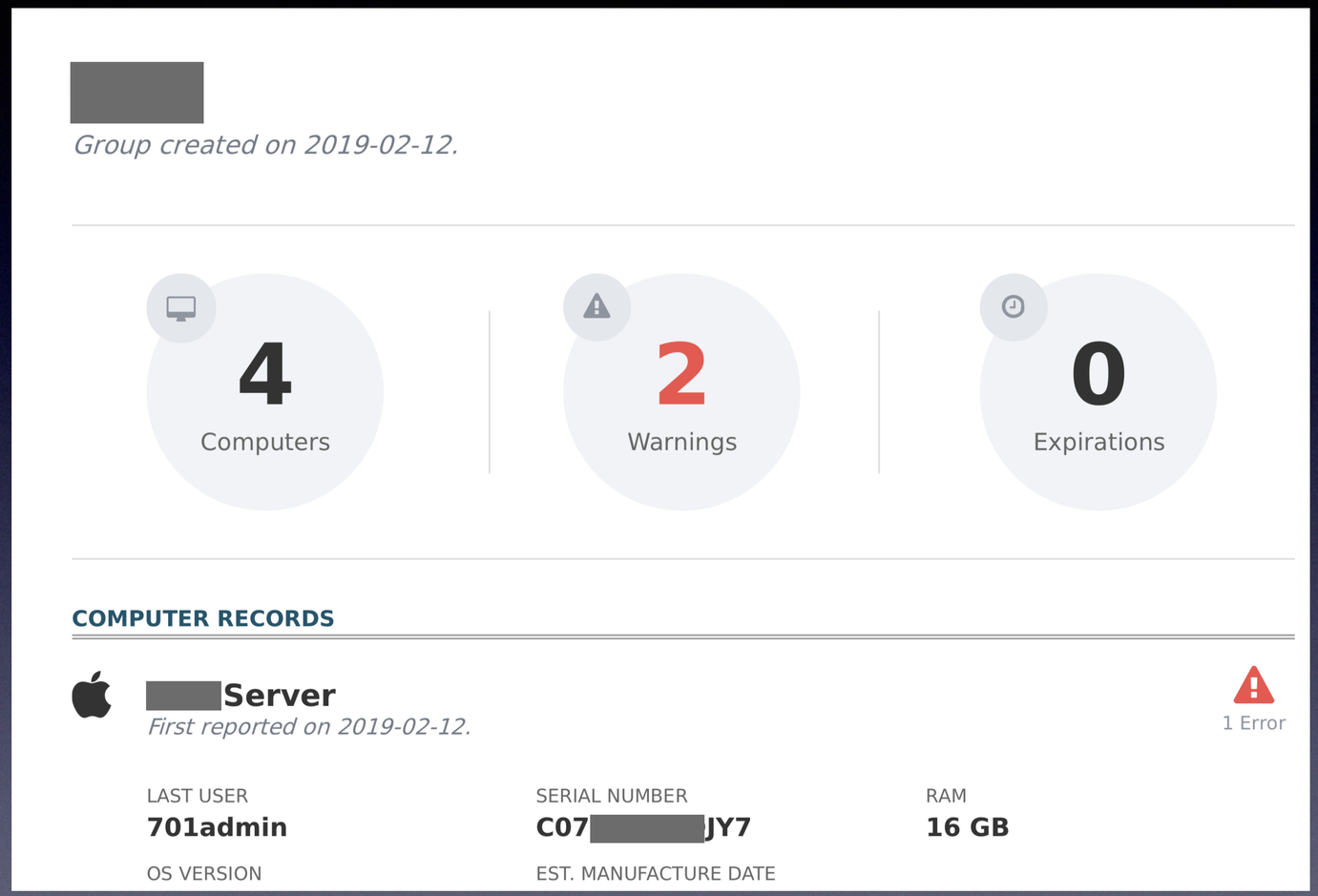
precursor.ca/slides

Watchman Monitoring

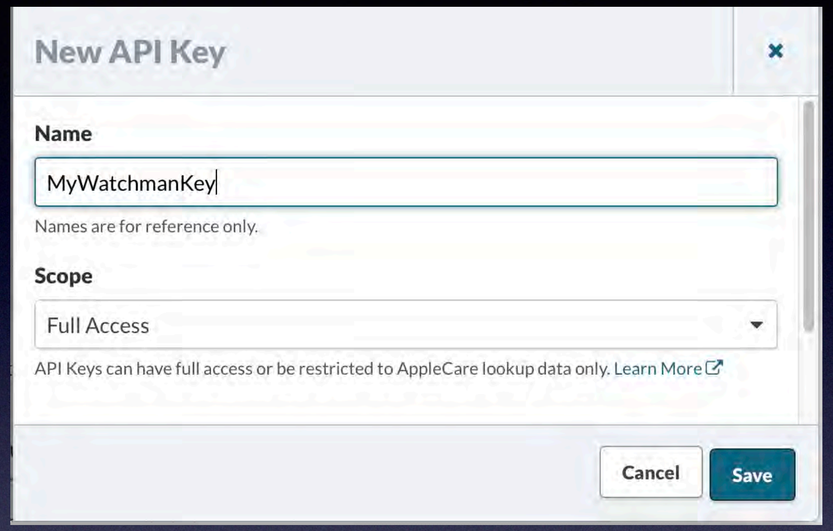
precursor.ca/slides



precursor.ca/slides

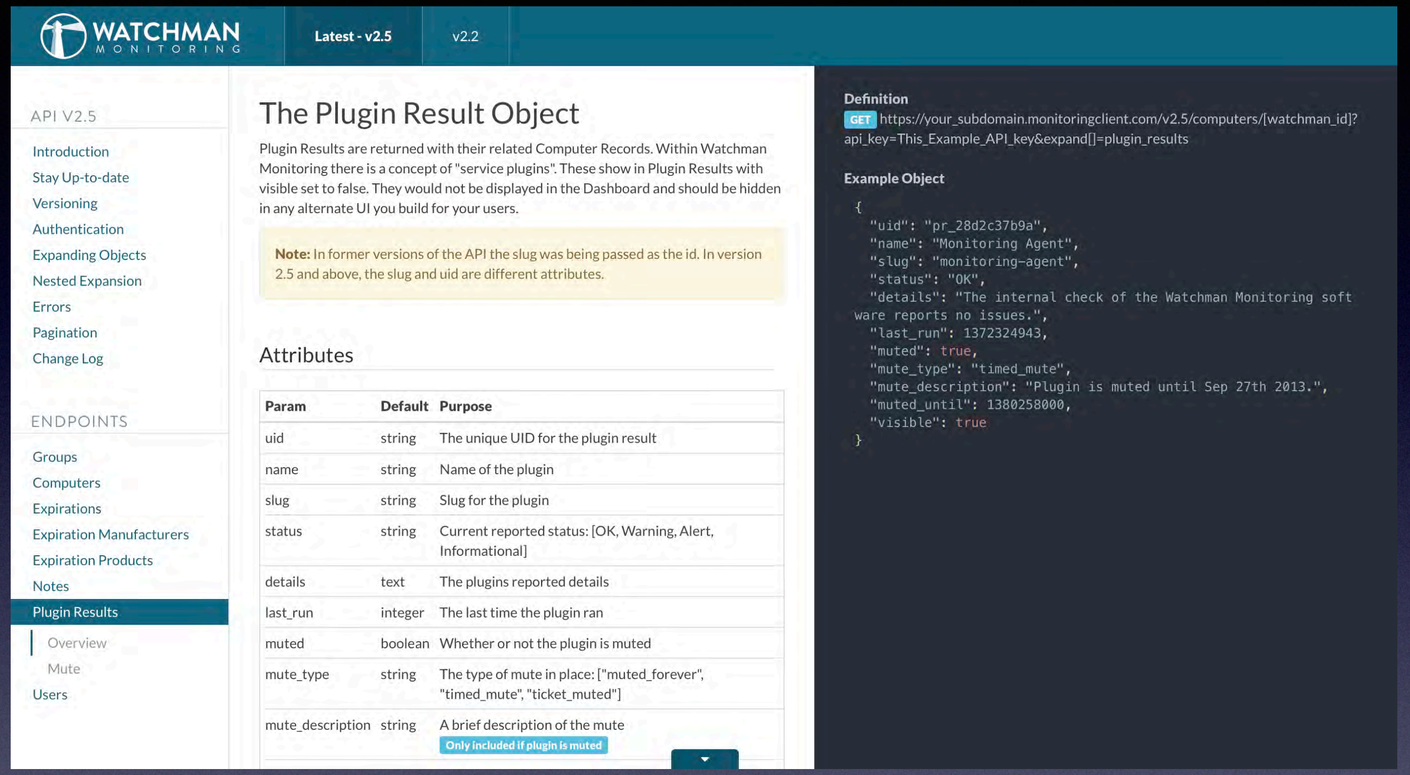


precursor.ca/slides



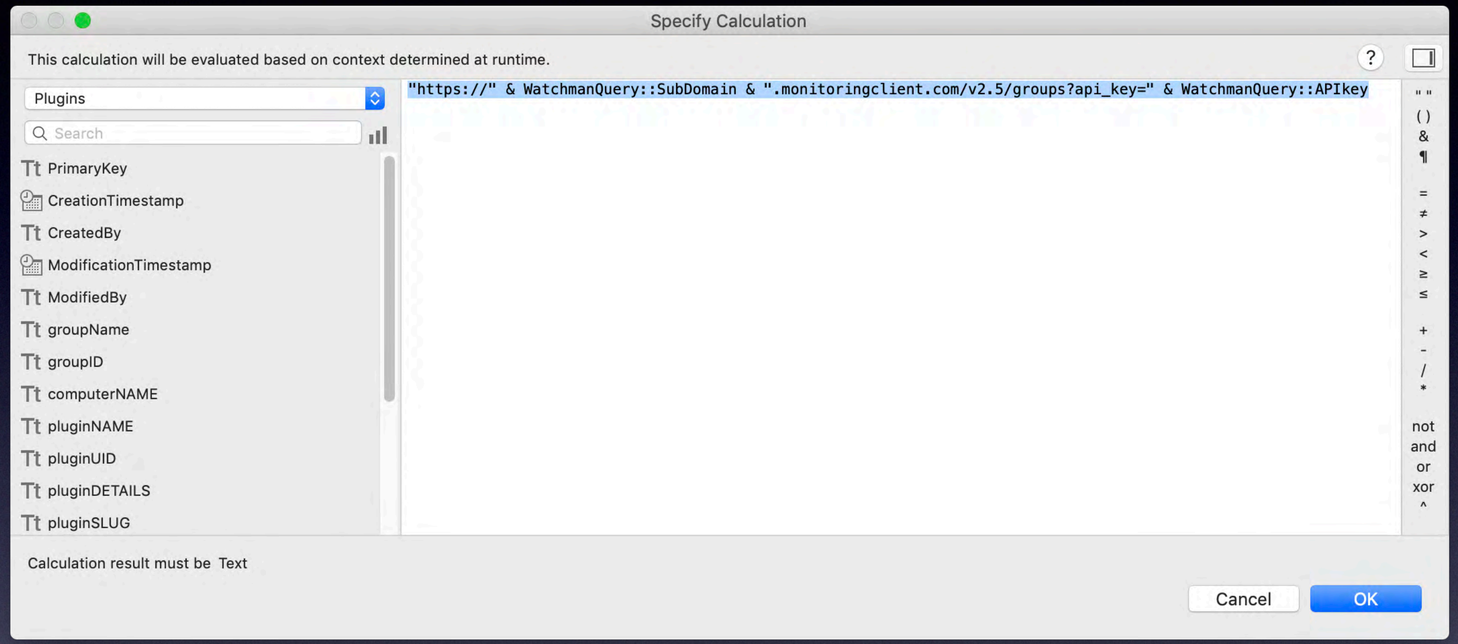
https:/your_subdomain.monitoringclient.com/settings/api

precursor.ca/slides

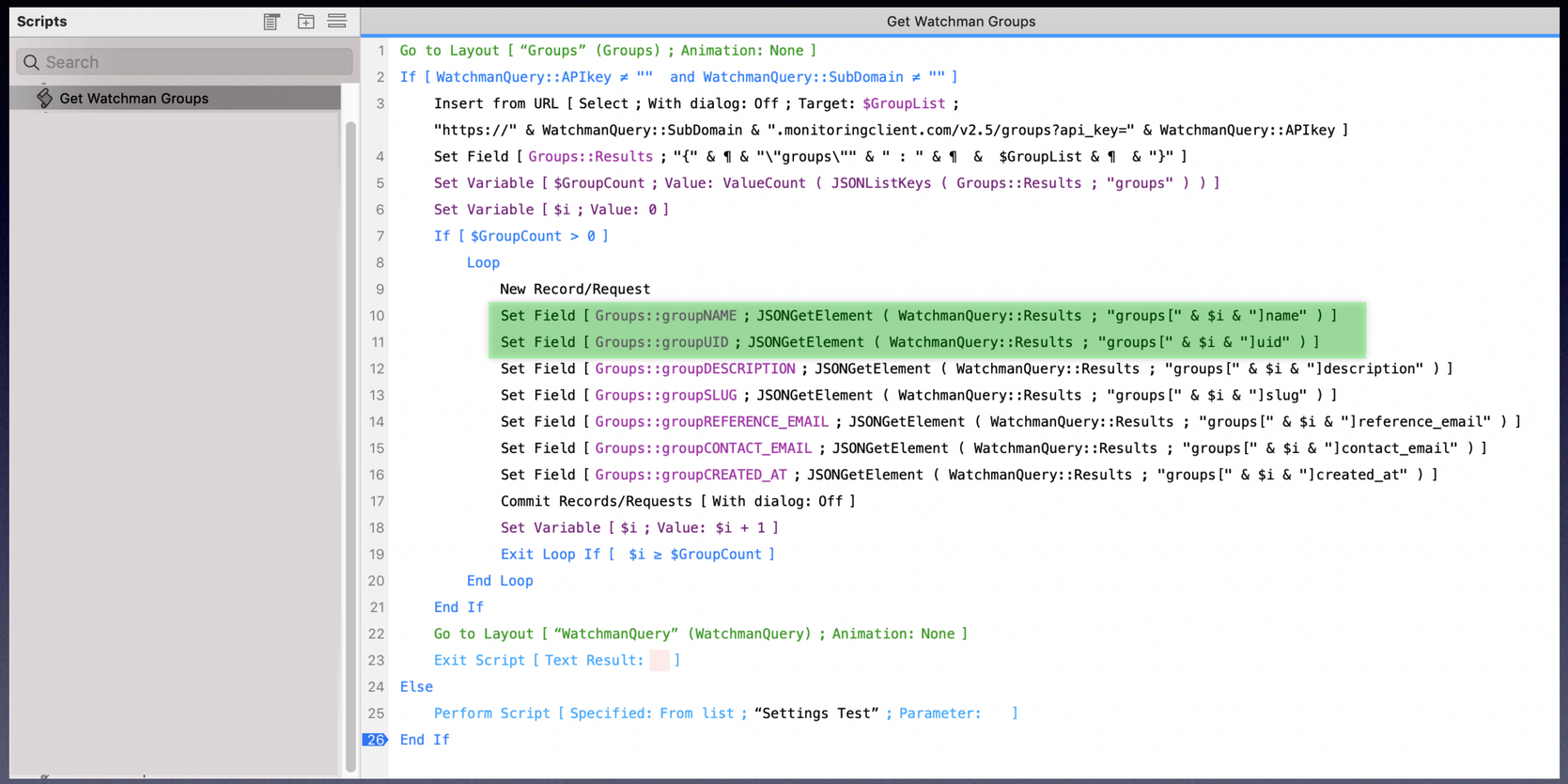


https://api.watchmanmonitoring.com/#plugin_result

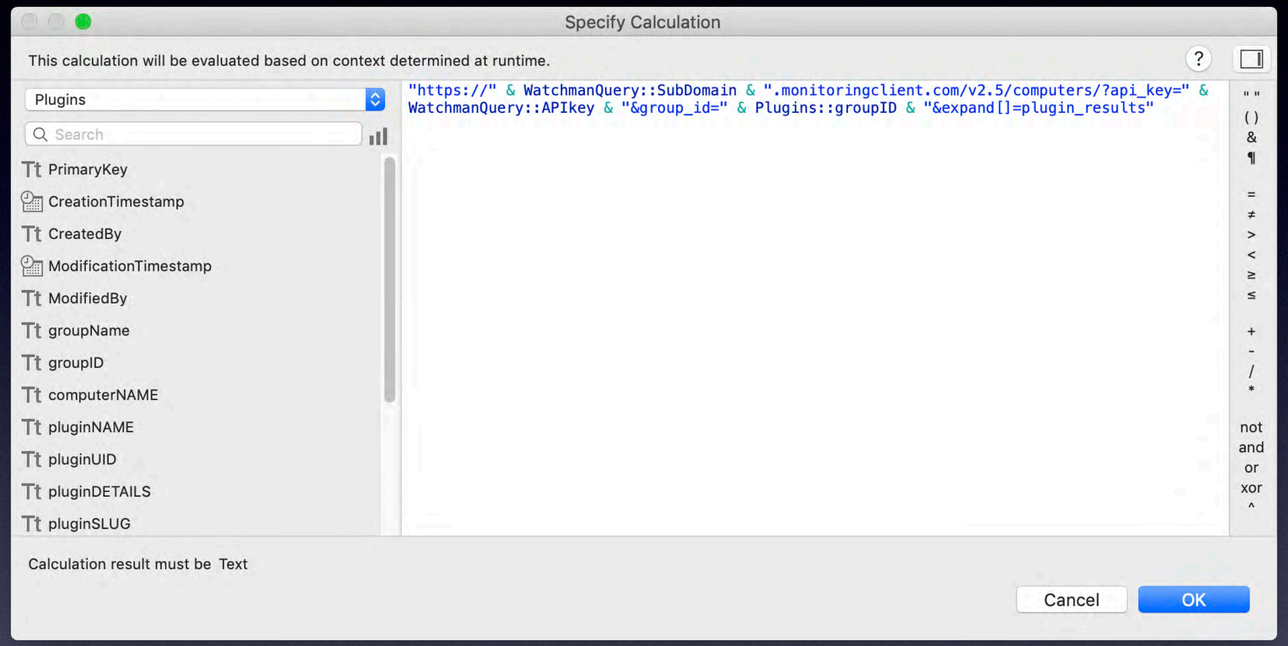
precursor.ca/slides



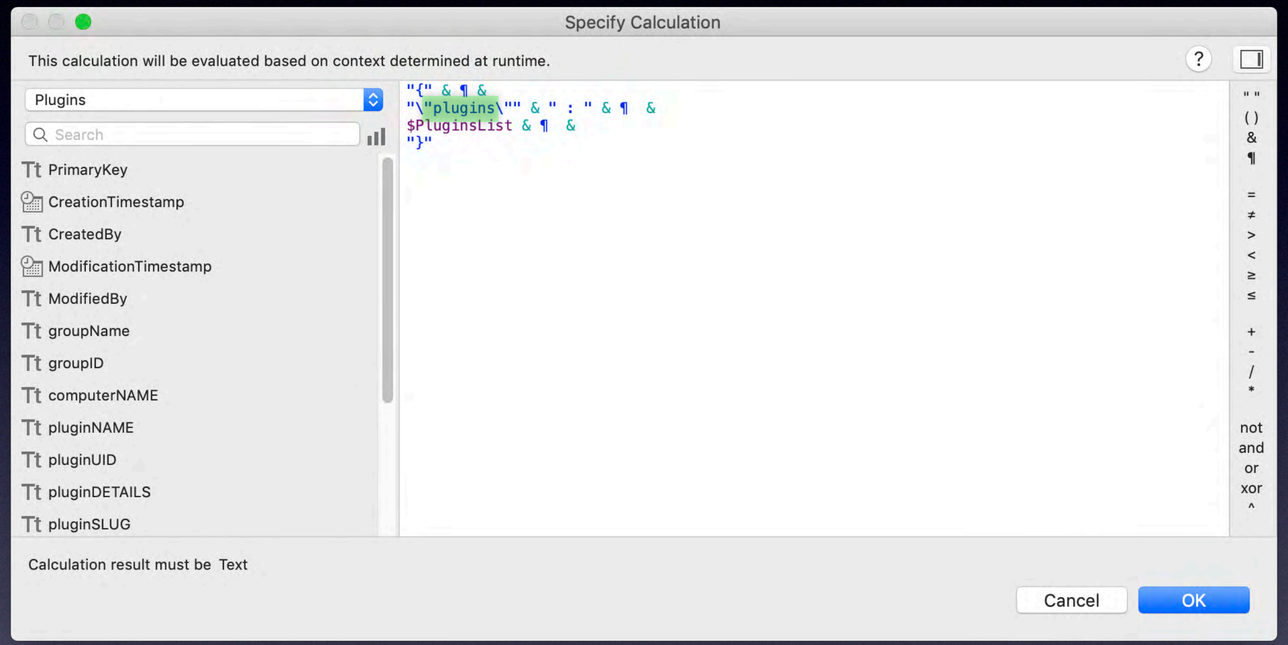
precursor.ca/slides



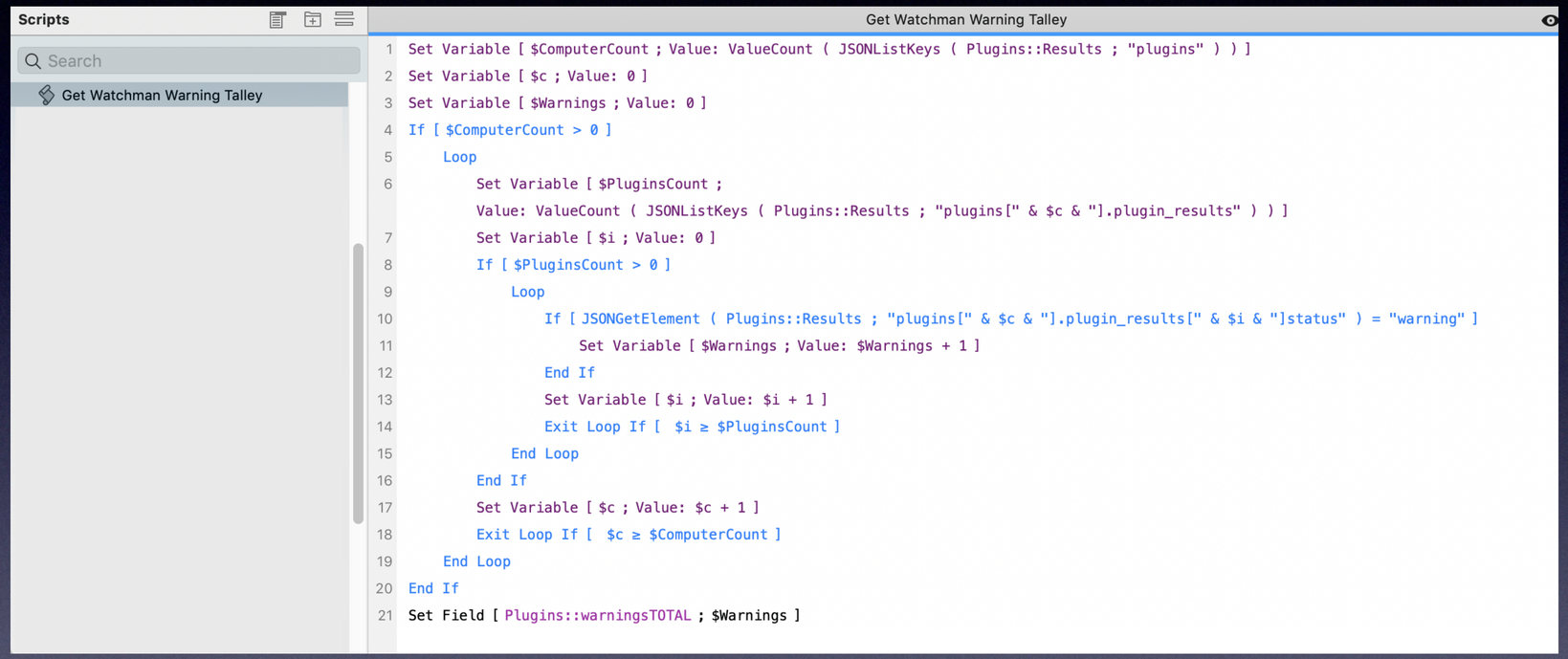
precursor.ca/slides



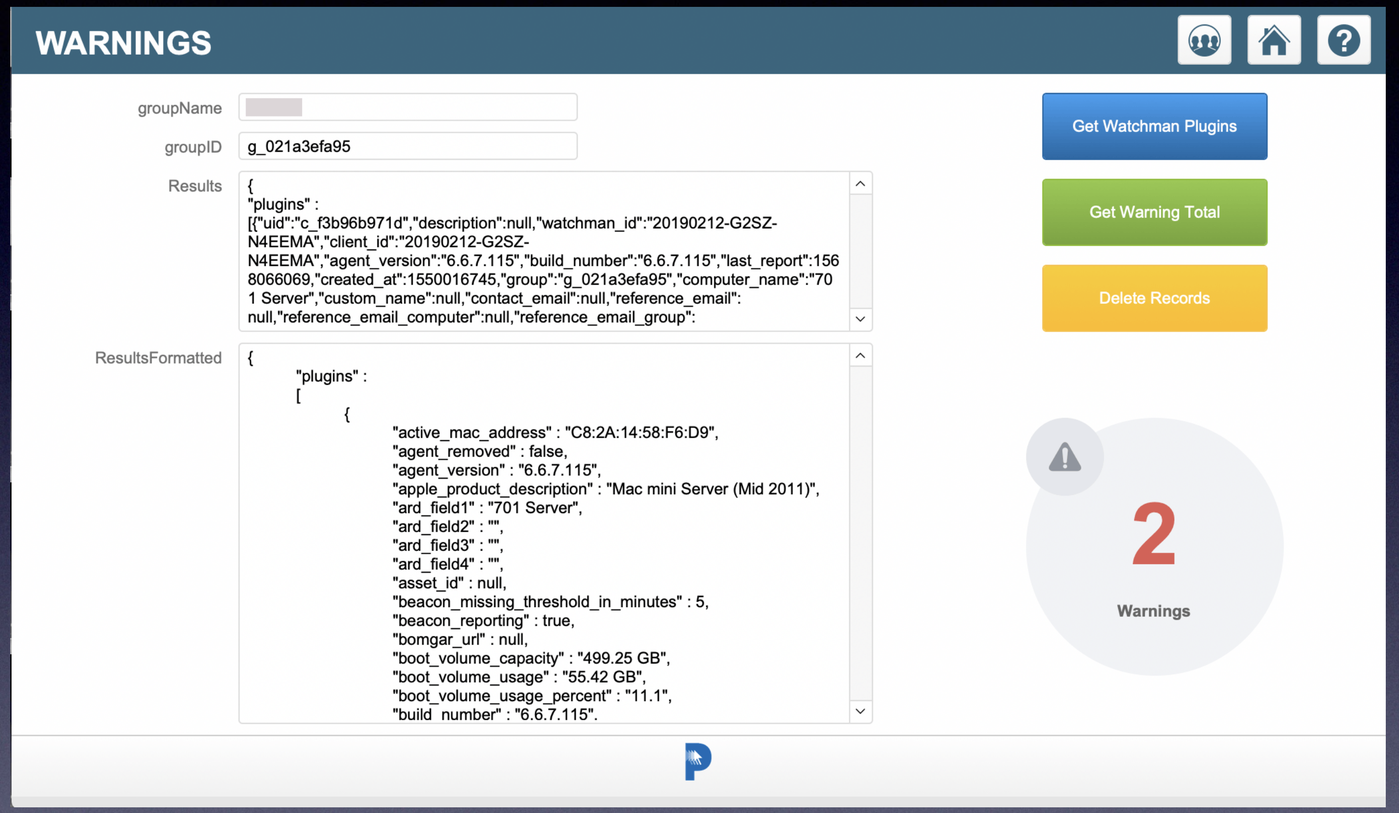
precursor.ca/slides



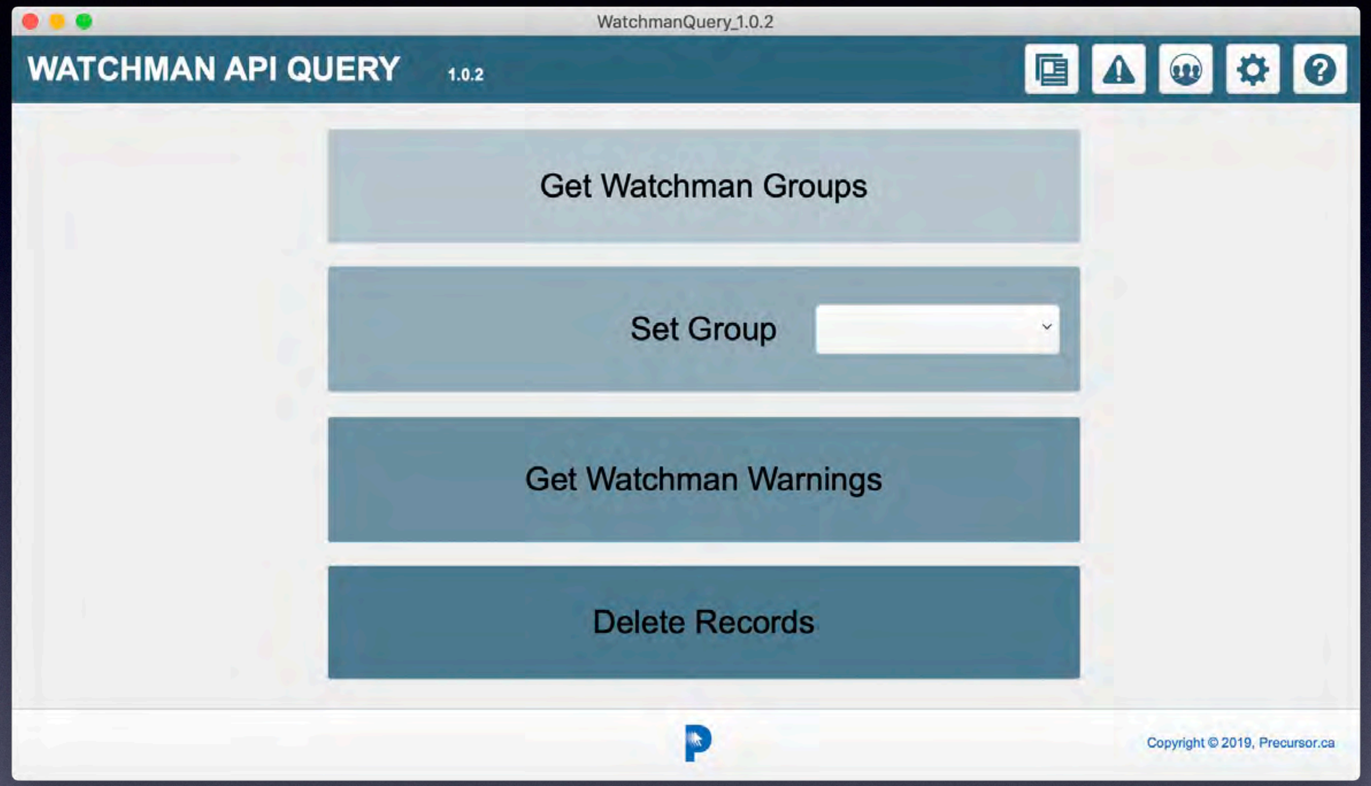
precursor.ca/slides



precursor.ca/slides



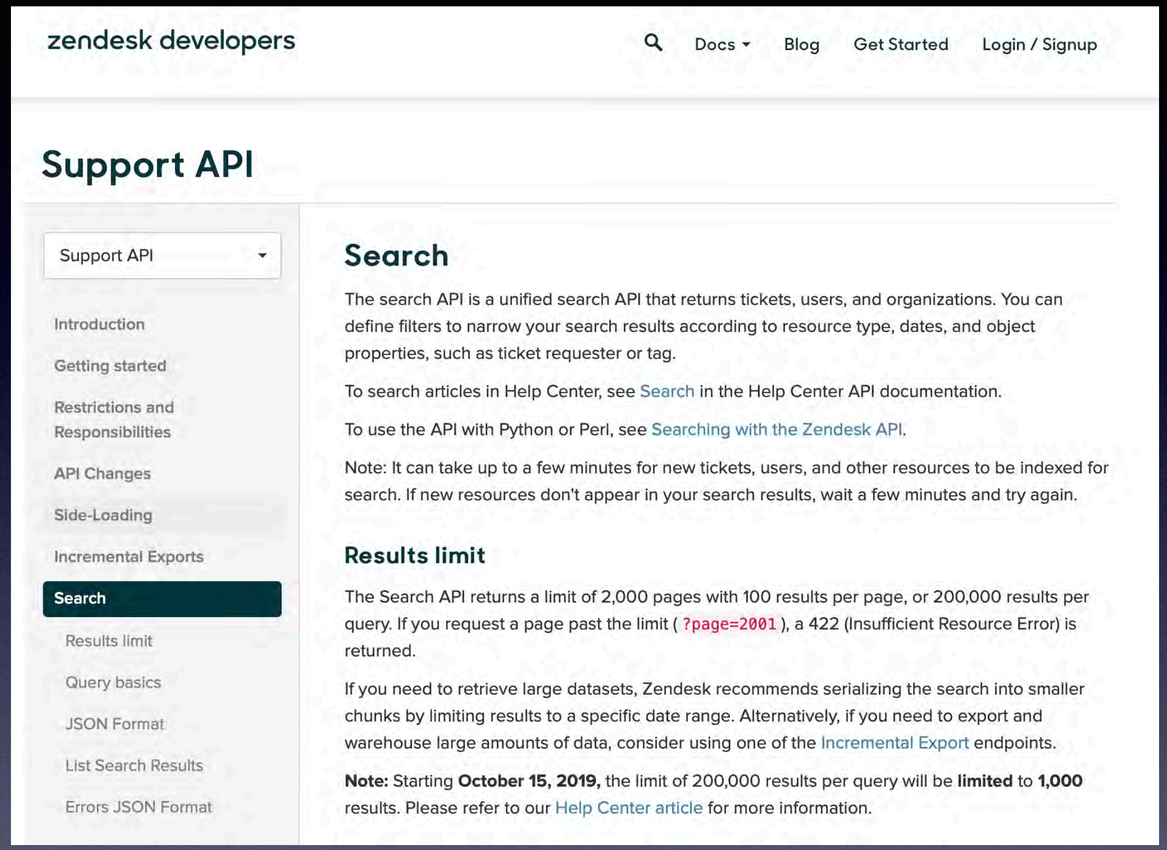
precursor.ca/slides



precursor.ca/slides

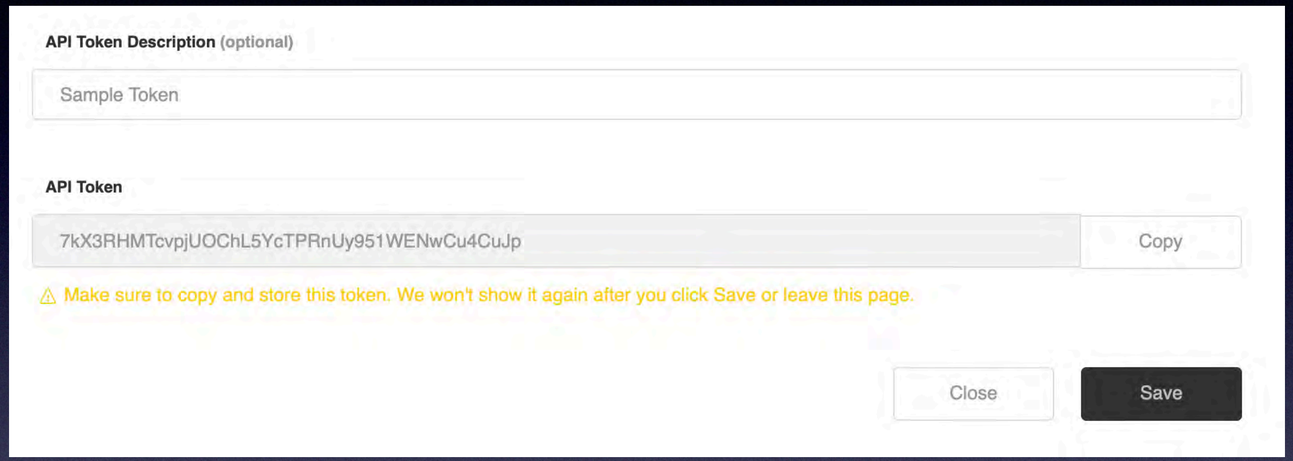
Zendesk

precursor.ca/slides



https://developer.Zendesk.com/rest_api/docs/support/

precursor.ca/slides



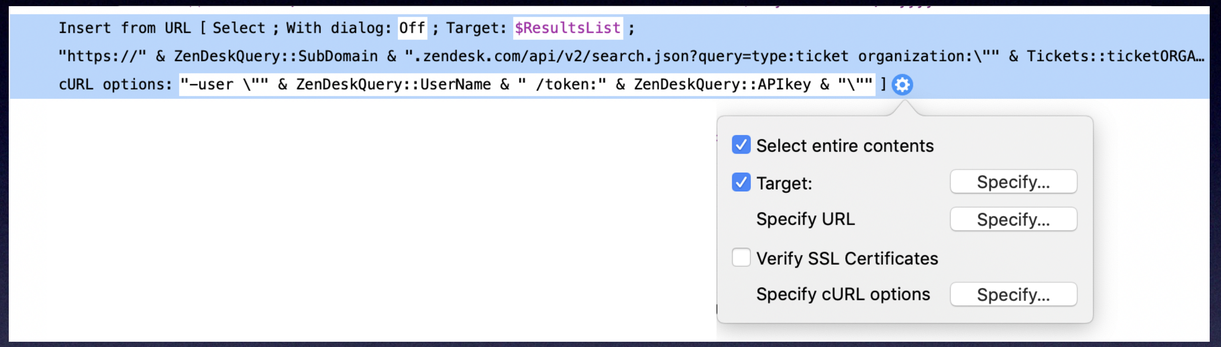
https://your_subdomain.zendesk.com/agent/admin/api/settings/tokens

precursor.ca/slides

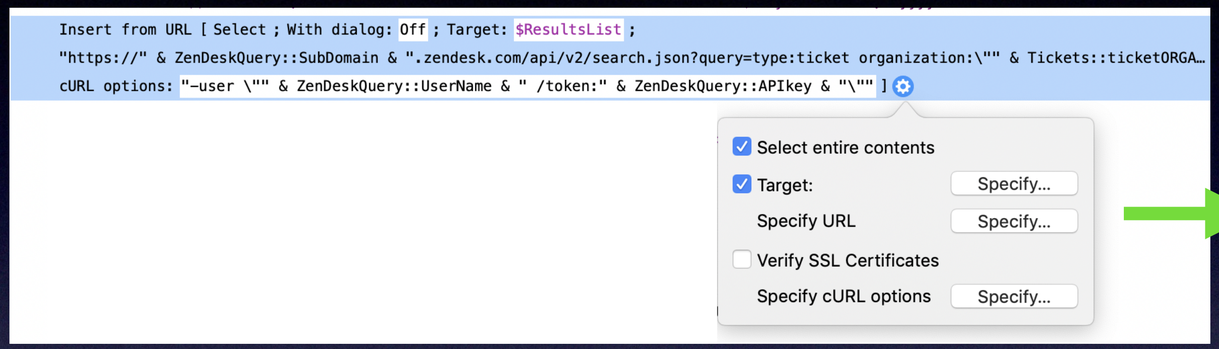
```
“https://subdomain.Zendesk.com/api/v2/search.json?
query=type:ticket organization:orgName solved>YYYY-MM-DD“
```

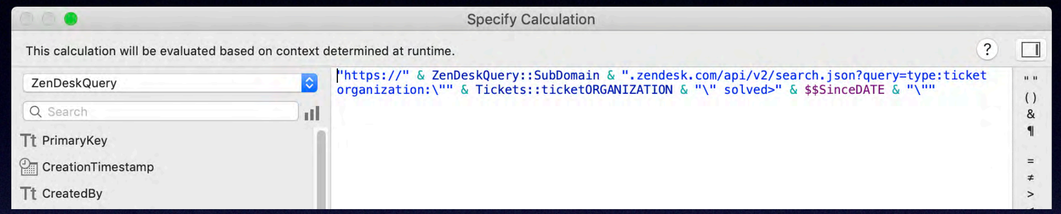
```
"https://" & ZendeskQuery::SubDomain & ".Zendesk.com/api/v2/
search.json?query=type:ticket organization:\"" &
Tickets::ticketORGANIZATION & "\" solved>" & $$Period & "\""
```

precursor.ca/slides



precursor.ca/slides

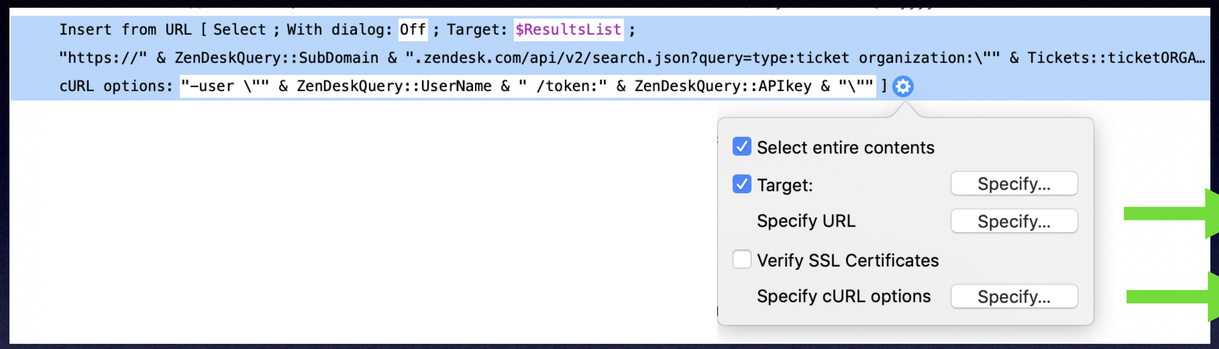


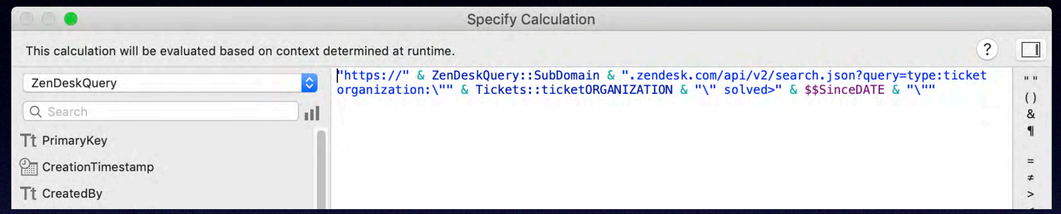


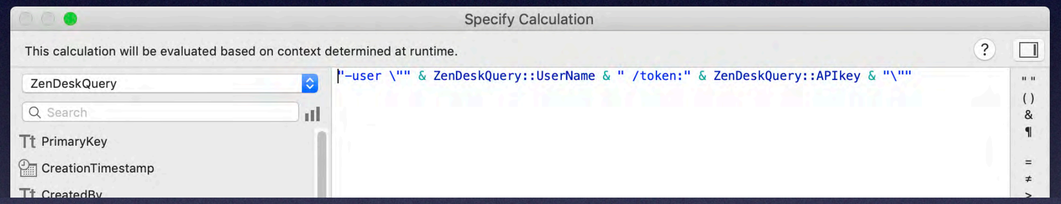
Insert from URL

```
"https://" & ZendeskQuery::SubDomain & ".Zendesk.com/api/v2/
search.json?query=type:ticket organization:\"" &
Tickets::ticketORGANIZATION & "\" solved>" & $$Period & "\""
```

precursor.ca/slides





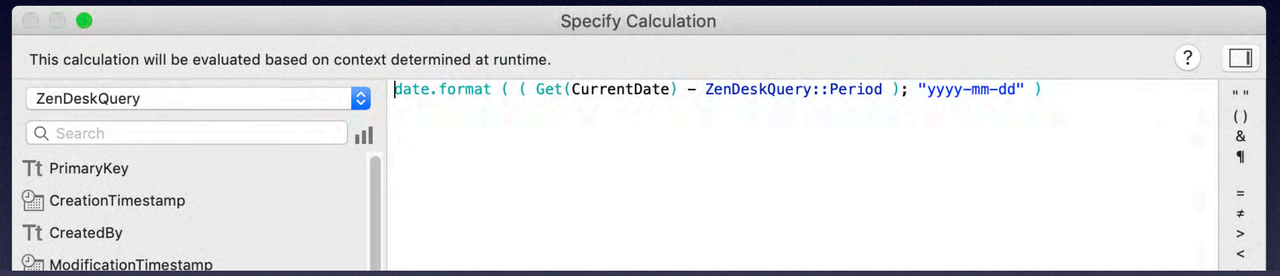


cURL Options

```
"-user \"" & ZendeskQuery::UserName & " /token:" &
ZendeskQuery::APIkey & "\""
```

precursor.ca/slides

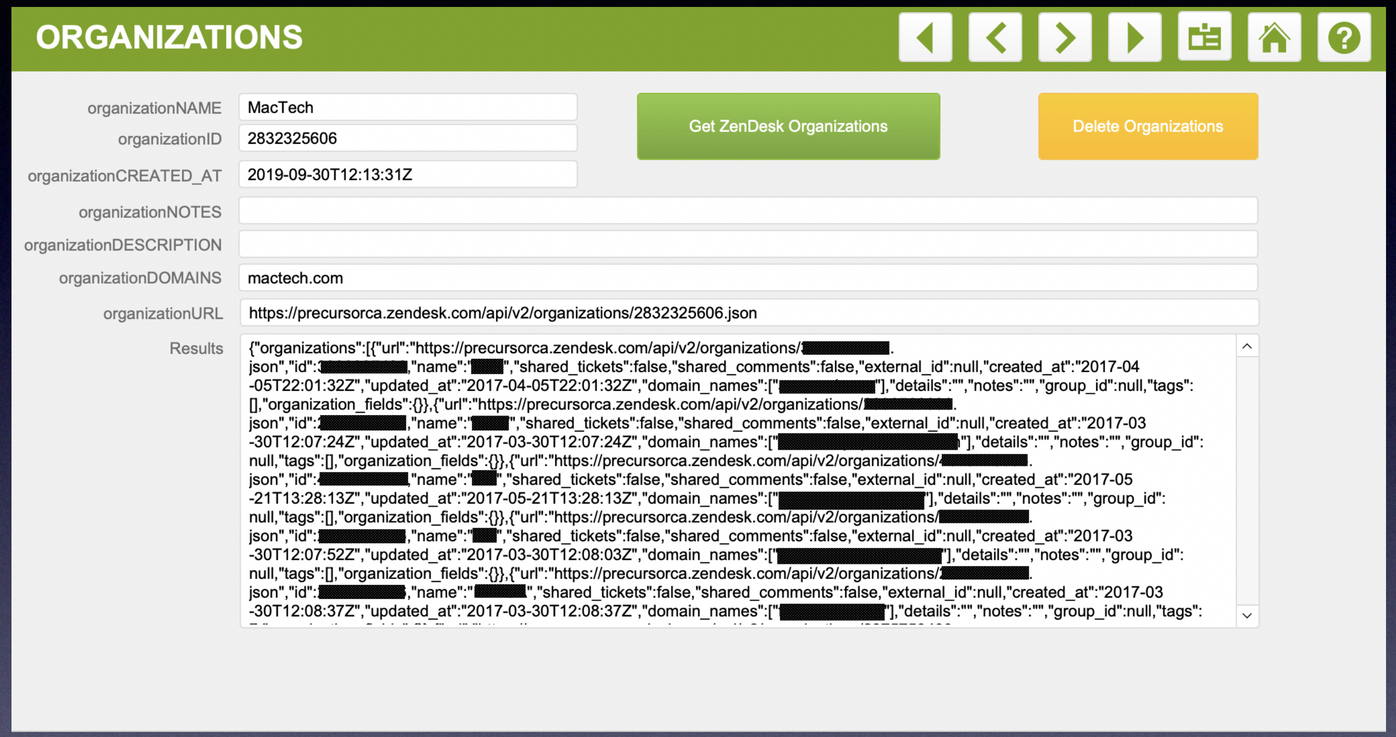
```
date.format ( ( Get(CurrentDate) - ZendeskQuery::Period ); "yyyy-mm-dd" )
```



date.format custom function by Tim Anderson Group

https://www.briandunning.com/cf/893

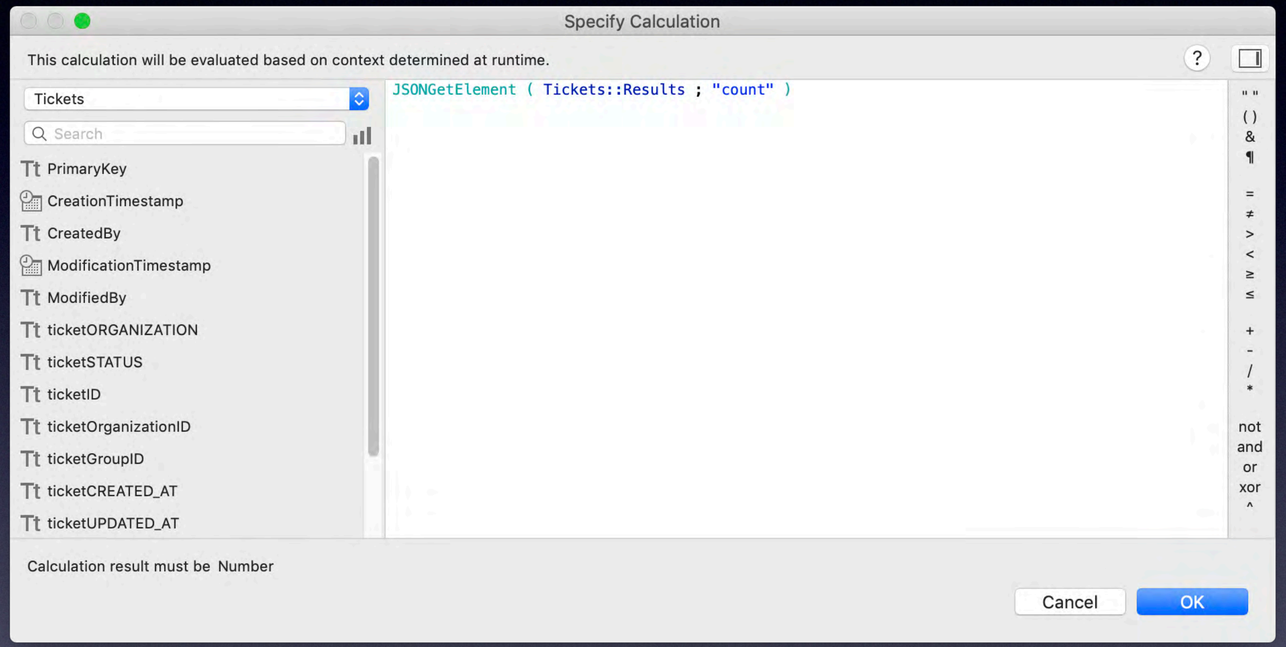
precursor.ca/slides



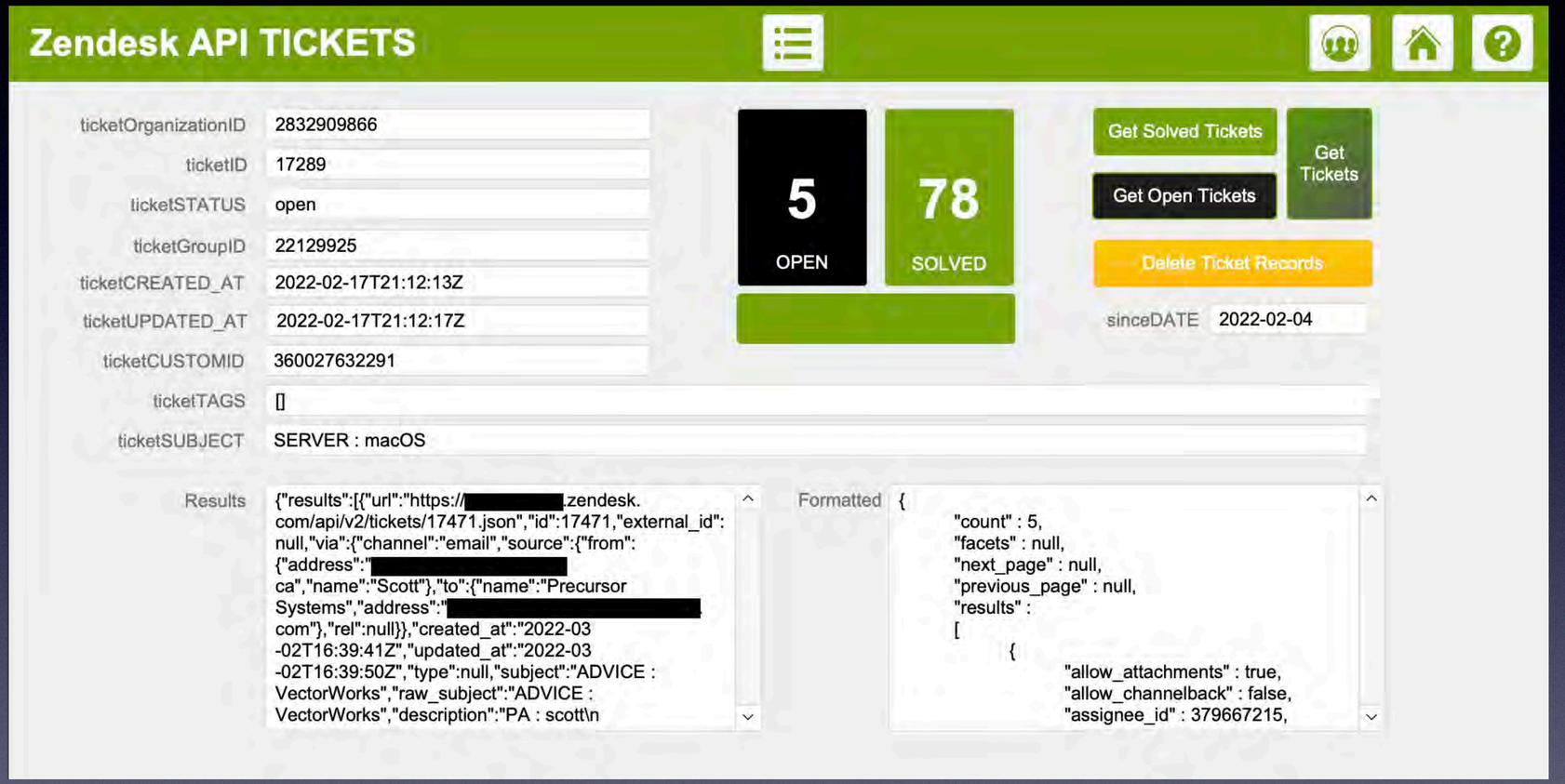
precursor.ca/slides

```
If [ ZenDeskQuery::APIkey ≠ ""  and ZenDeskQuery::SubDomain ≠ "" ]
    Insert from URL [ Select ; With dialog: Off ; Target: $OrganizationList ;
    "https://" & ZenDeskQuery::SubDomain & ".zendesk.com/api/v2/organizations.json" ; cURL options:
    "-user \"" & ZenDeskQuery::UserName & " /token:" & ZenDeskQuery::APIkey & "\"" ]
    Set Field [ Organizations::Results ; $OrganizationList ]
    Set Variable [ $OrganizationCount ; Value: ValueCount ( JSONListKeys ( Organizations::Results ; "organizations" ) ) ]
    Set Variable [ $i ; Value: 0 ]
    If [ $OrganizationCount > 0 ]
        Loop
            New Record/Request
            Set Field [ Organizations::organizationNAME ; JSONGetElement ( Organizations::Results ; "organizations[" & $i & "]name" ) ]
            Set Field [ Organizations::organizationID ; JSONGetElement ( Organizations::Results ; "organizations[" & $i & "]id" ) ]
            Set Field [ Organizations::organizationDESCRIPTION ;
            JSONGetElement ( Organizations::Results ; "organizations[" & $i & "]details" ) ]
            Set Field [ Organizations::organizationURL ; JSONGetElement ( Organizations::Results ; "organizations[" & $i & "]url" ) ]
            Set Field [ Organizations::organizationDOMAINS ;
            JSONGetElement ( Organizations::Results ; "organizations[" & $i & "]domain_names" ) ]
            Set Field [ Organizations::organizationNOTES ; JSONGetElement ( Organizations::Results ; "organizations[" & $i & "]notes" ) ]
            Set Field [ Organizations::organizationCREATED_AT ;
            JSONGetElement ( Organizations::Results ; "organizations[" & $i & "]created_at" ) ]
            Commit Records/Requests [ With dialog: Off ]
            Set Variable [ $i ; Value: $i + 1 ]
            Exit Loop If [  $i ≥ $OrganizationCount ]
        End Loop
    End If
    Exit Script [ Text Result:  ]
Else
    Perform Script [ Specified: From list ; “Settings Test” ; Parameter:    ]
End If
```

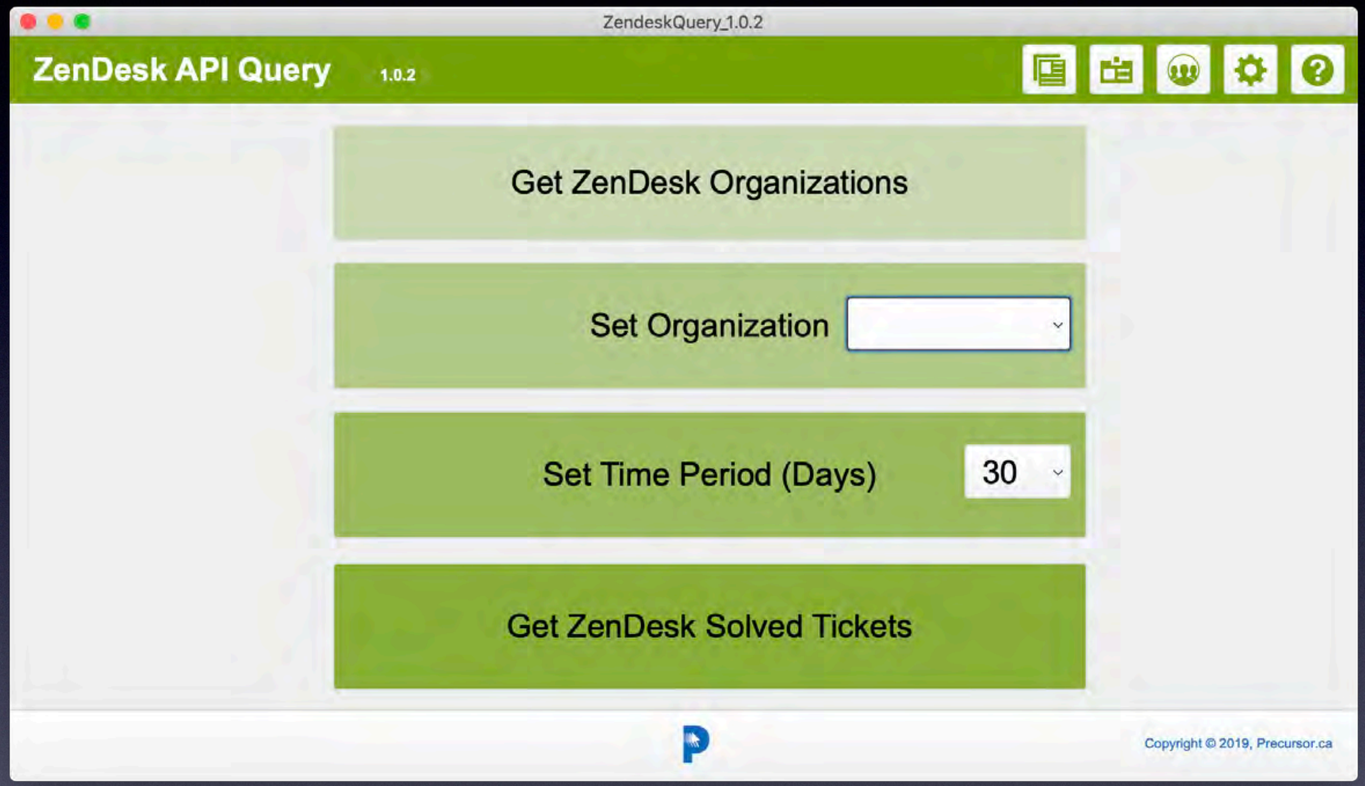
precursor.ca/slides



precursor.ca/slides

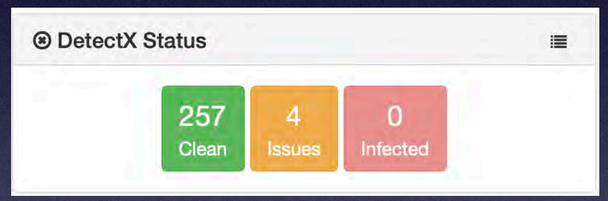


precursor.ca/slides

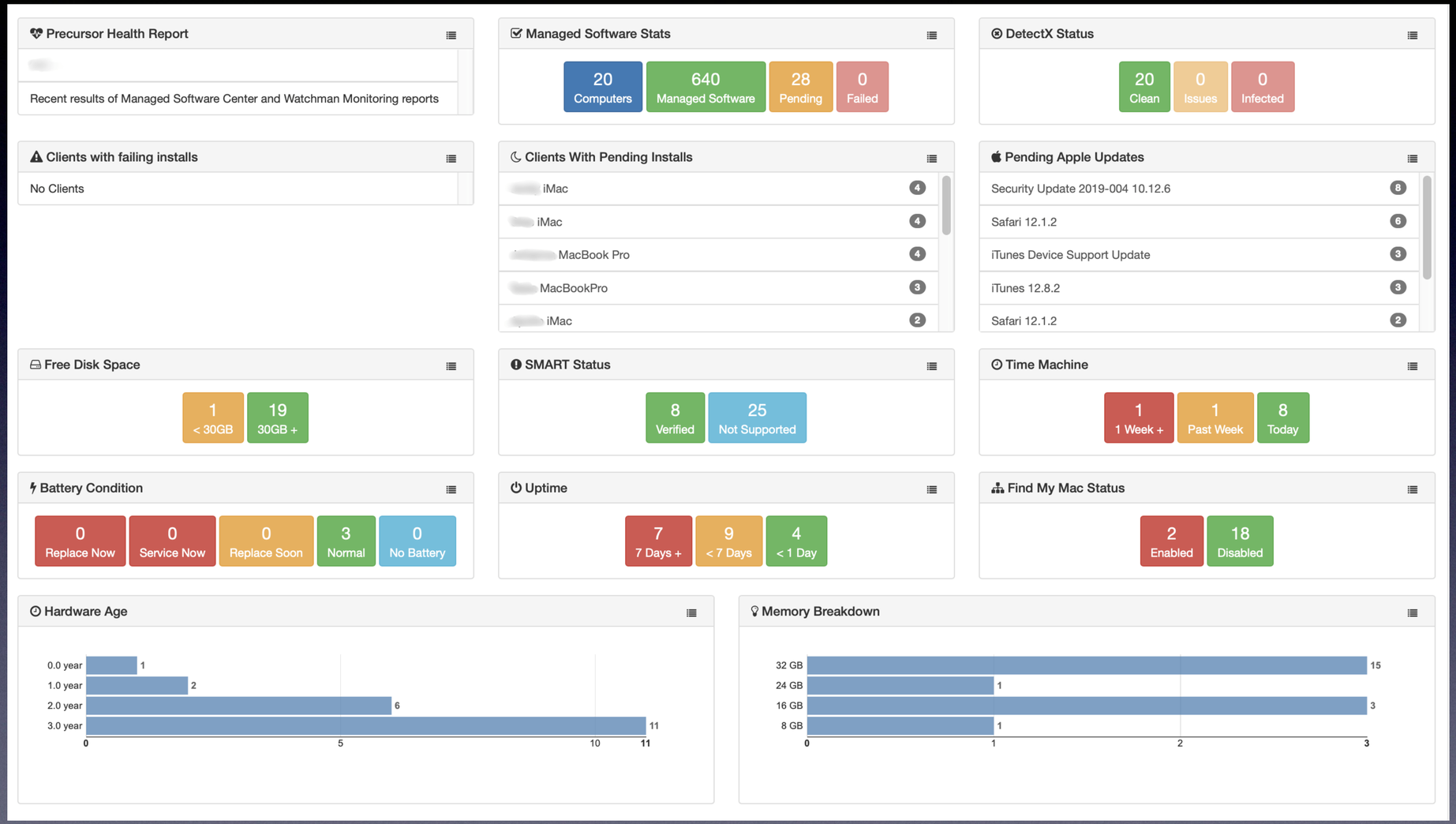


precursor.ca/slides

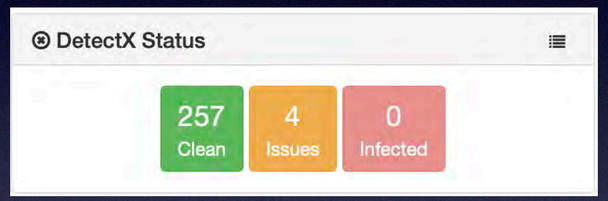
munkireport



precursor.ca/slides



precursor.ca/slides



precursor.ca/slides

https://sqwarq.com/detectx/

precursor.ca/slides

API

Arjen van Bochoven edited this page on Apr 12, 2017 · 6 revisions

To get data out of Munkireport there is a Datatables API that is also used for the listing reports. To help you get started use the small shell script or python script below. To get the data to CSV format, pipe the output to JSON to CSV parser (see below).

Shell Version

```
#!/bin/sh
#
# Script to run automated queries against the munkireport datatables API
#
# Results are returned in JSON format
# The actual entries are in the 'data' variable
#
# To make this work, set up a regular user in munkireport and adjust the
# proper values below
#
# Author: Arjen van Bochoven
# Date: 2015-11-06

# Retrieve data from munkireport
# DEBUG=1
MR_BASE_URL='https://yourmunkireportserver/index.php?'
MR_DATA_QUERY='/datatables/data'
MR_LOGIN='datauser'
MR_PASSWORD='secretpassword'

CLIENT_COLUMNS=(
    "machine.serial_number"
    "machine.hostname"
```

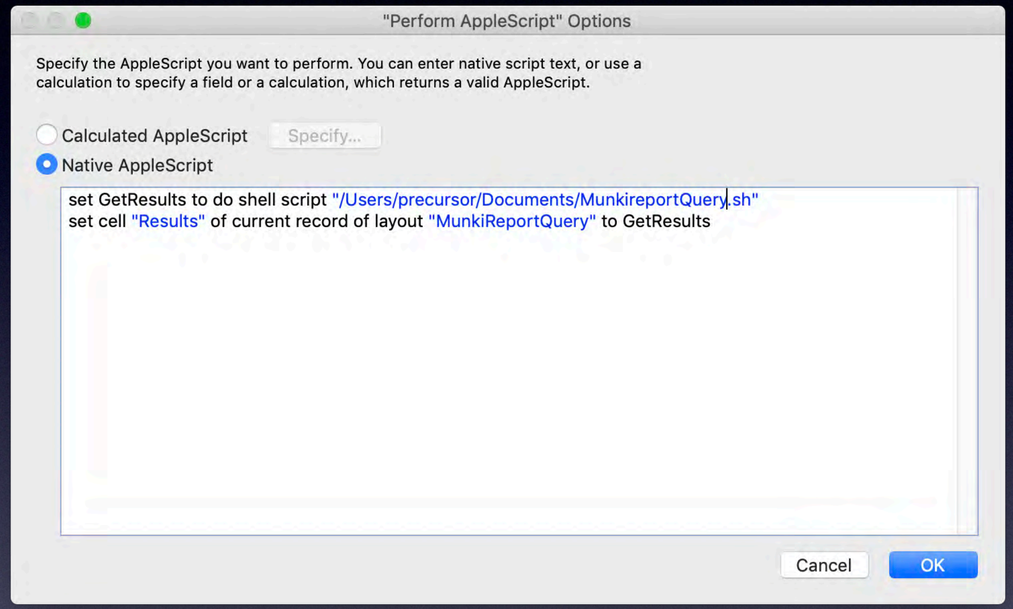
https://github.com/munkireport/munkireport-php/wiki/API

precursor.ca/slides

```
CLIENT_COLUMNS=(
    "machine.serial_number"
    "machine.hostname"
    "machine.machine_desc"
    "reportdata.timestamp"
    "reportdata.console_user"
    "machine.os_version"
    "reportdata.remote_ip"
    "munkireport.manifestname"
)
```

“detectx.status”
“detectx.numberofissues”

precursor.ca/slides



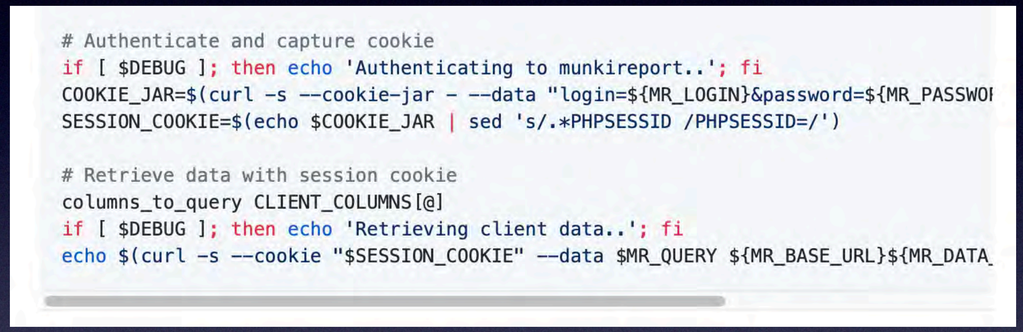
precursor.ca/slides

https://munkireport.precursor.ca/index.php?/module/detectx/get_stats/

```
$.getJSON( appUrl + '/module/detectx/get_stats', function( data ) {
  if(data.error){
    //alert(data.error);
    return;
  }

  $.each(['Clean', 'Issues', 'Infected'], function(index, type){
```

Status: Clean Issues Infected

precursor.ca/slides



```
# Authenticate and capture cookie
if [ $DEBUG ]; then echo 'Authenticating to munkireport..'; fi
COOKIE_JAR=$(curl -s --cookie-jar - --data "login=${MR_LOGIN}&password=${MR_PASSWOI
SESSION_COOKIE=$(echo $COOKIE_JAR | sed 's/.*PHPSESSID /PHPSESSID=/')

# Retrieve data with session cookie
columns_to_query CLIENT_COLUMNS[@]
if [ $DEBUG ]; then echo 'Retrieving client data..'; fi
echo $(curl -s --cookie "$SESSION_COOKIE" --data $MR_QUERY ${MR_BASE_URL}${MR_DATA_
```

precursor.ca/slides

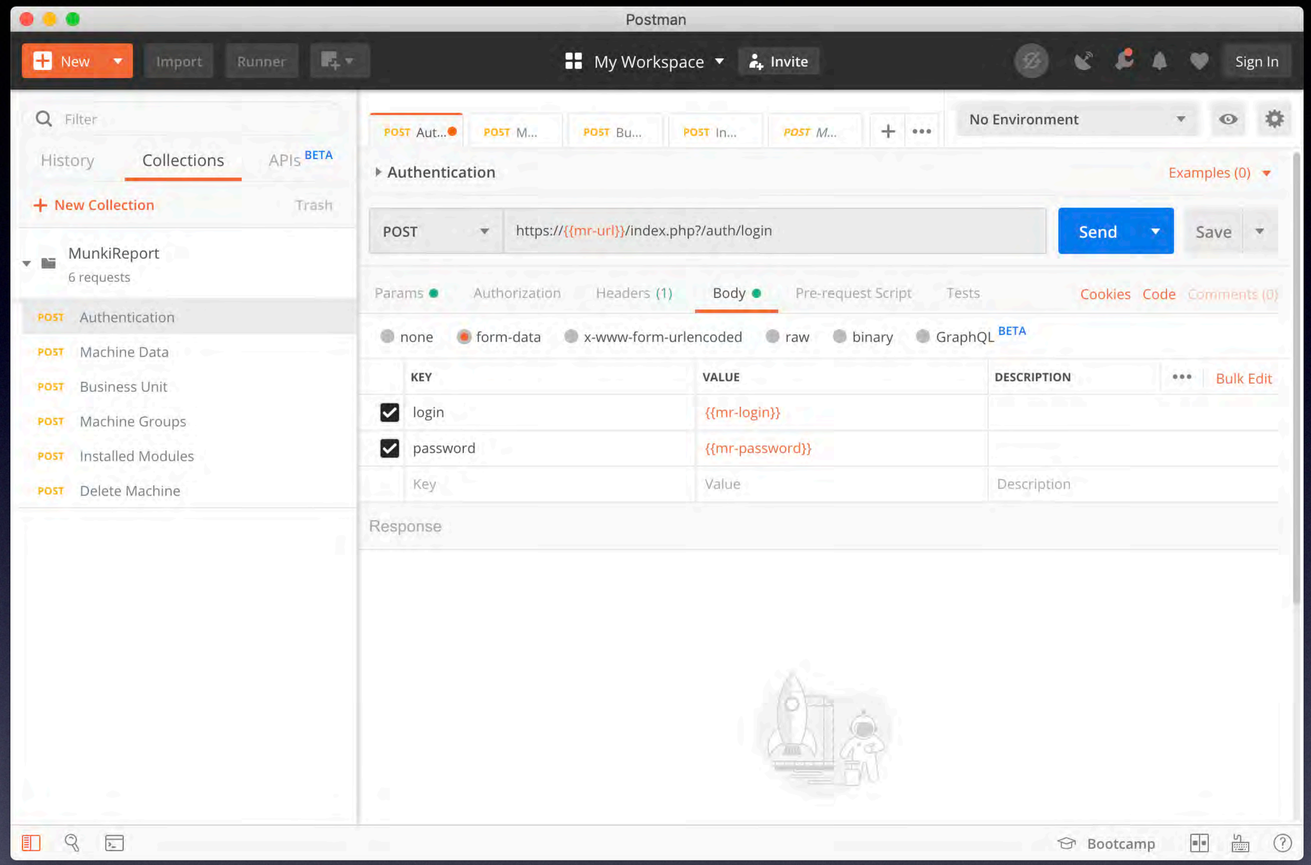
```
// the authentication data we wish to send
"-d @$data " &amp;

// set the Content type header to JSON
"--header Content-Type:application/json " &amp;

// create a cookie jar to store cookies in
"--cookie-jar $$CookieJar "
```

```
Set Variable [ $CookieString ; Value : Substitute ( $$CookieJar ; "¶" ; Char(10) )
```

```
// for debugging ONLY set error reporting on
"--show-error " &amp;
"--dump-header $$Headers " &amp;

// set the Contet type header to JSON
"--header Content-Type:application/json " &amp;

// send the cookies with the request
"--cookie $CookieString"
```

https://msdev.co.uk/adventures-in-filemaker-curl-the-cookie-jar/

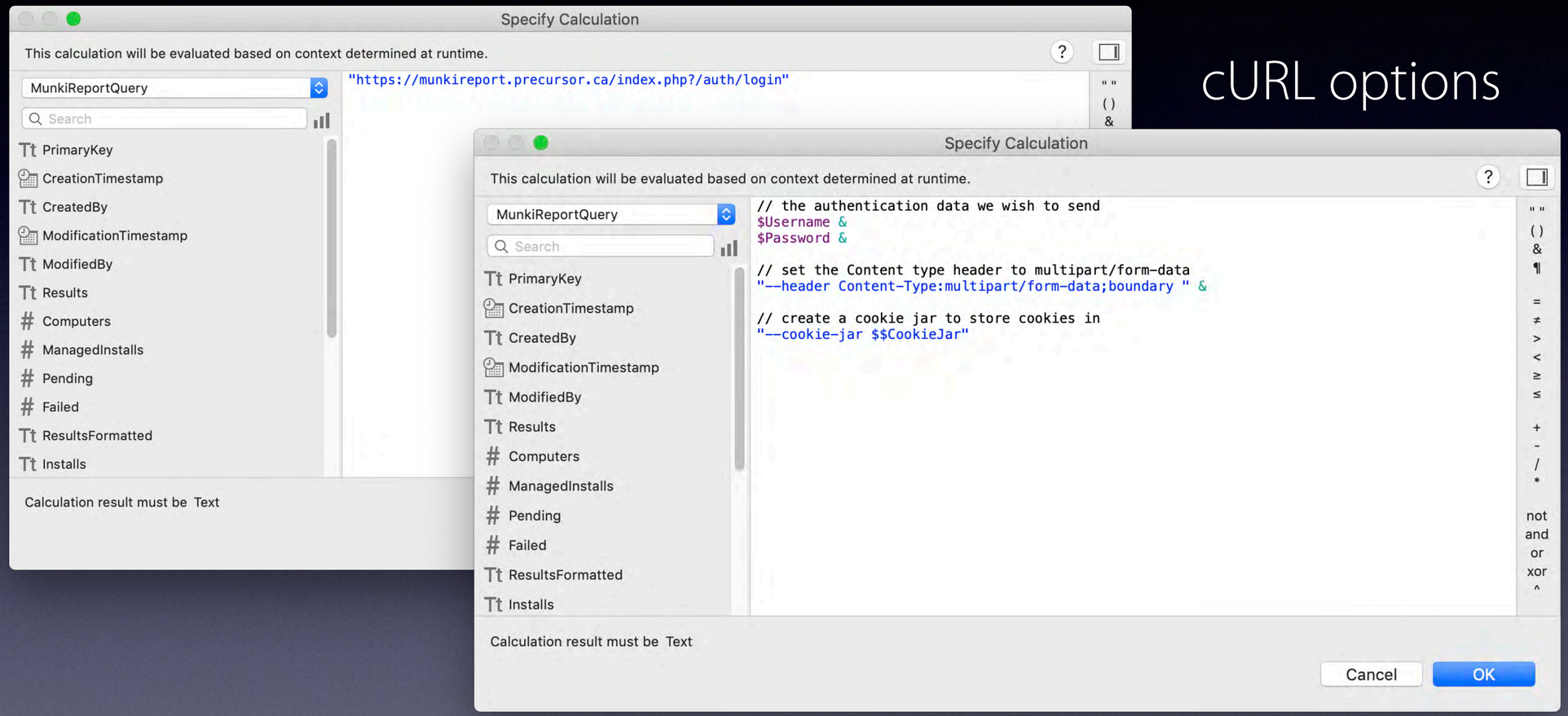
precursor.ca/slides



https://github.com/joncrain/munkireport-postman-collection

precursor.ca/slides

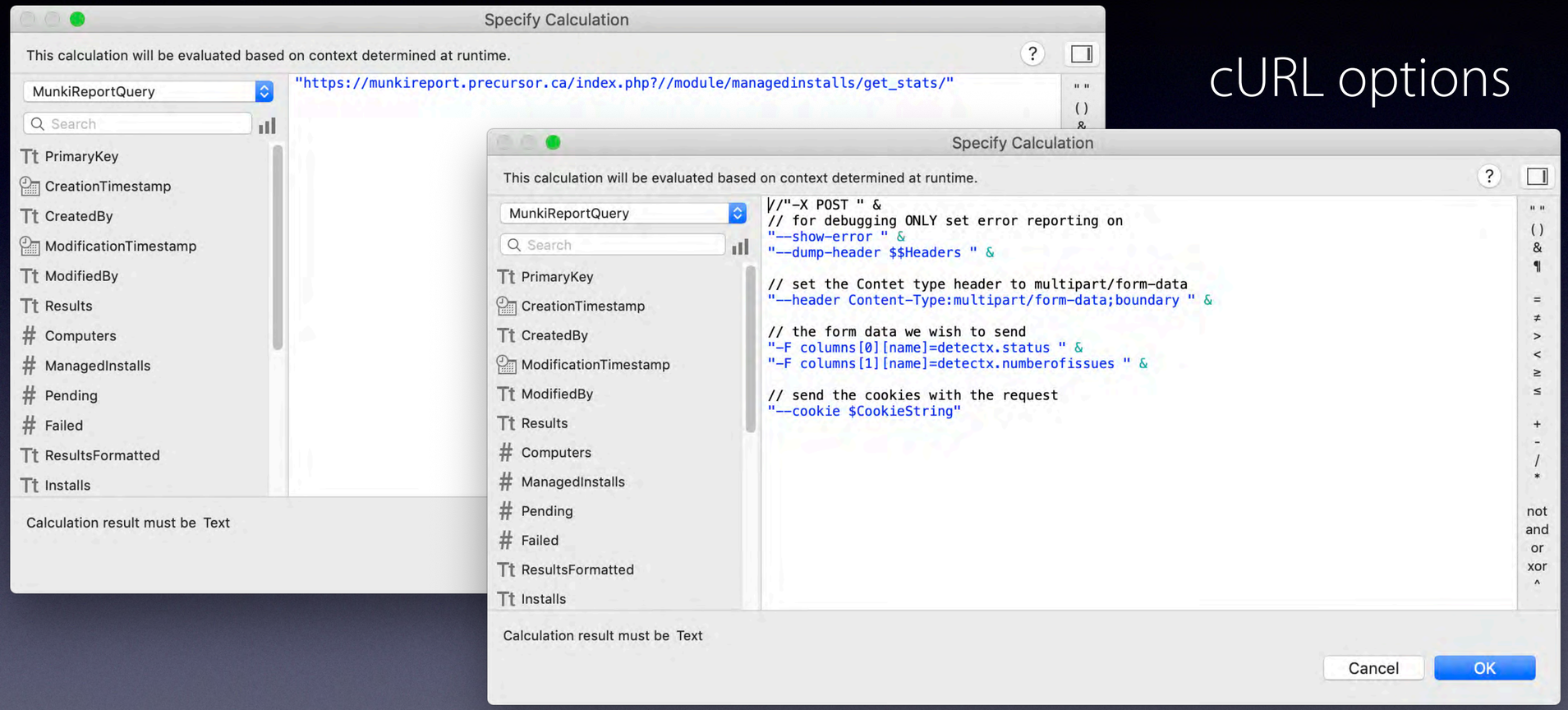
Specify URL



cURL options

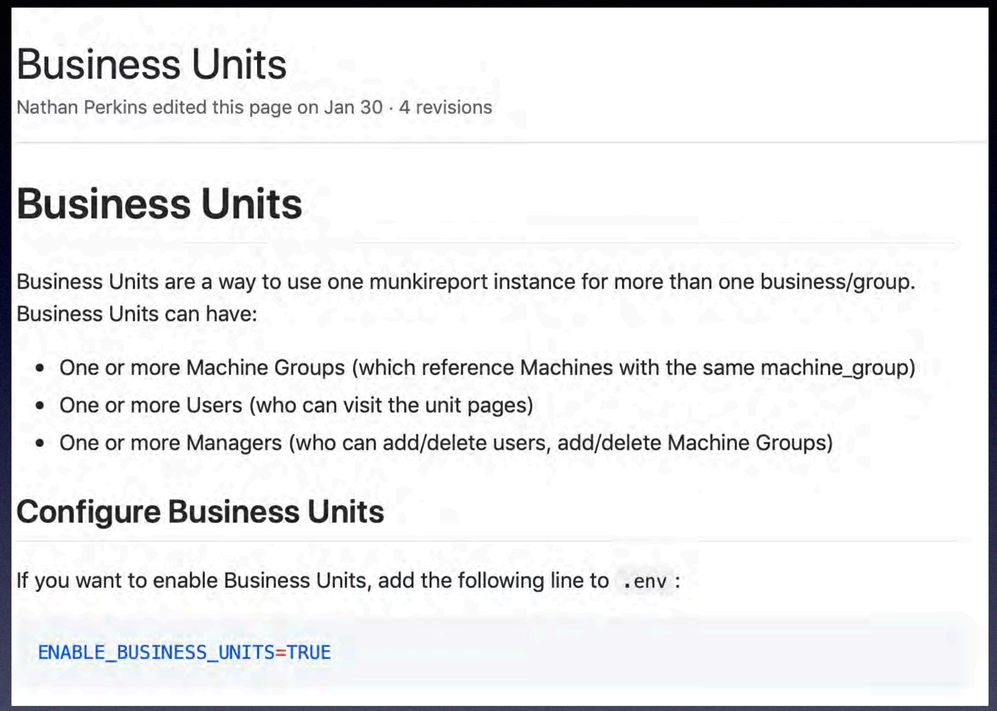
precursor.ca/slides

Specify URL



cURL options

precursor.ca/slides



Business Units

Nathan Perkins edited this page on Jan 30 · 4 revisions

Business Units

Business Units are a way to use one munkireport instance for more than one business/group.
Business Units can have:

- One or more Machine Groups (which reference Machines with the same machine_group)
- One or more Users (who can visit the unit pages)
- One or more Managers (who can add/delete users, add/delete Machine Groups)

Configure Business Units

If you want to enable Business Units, add the following line to `.env` :

```
ENABLE_BUSINESS_UNITS=TRUE
```

https://github.com/munkireport/munkireport-php/wiki/Business-Units

precursor.ca/slides

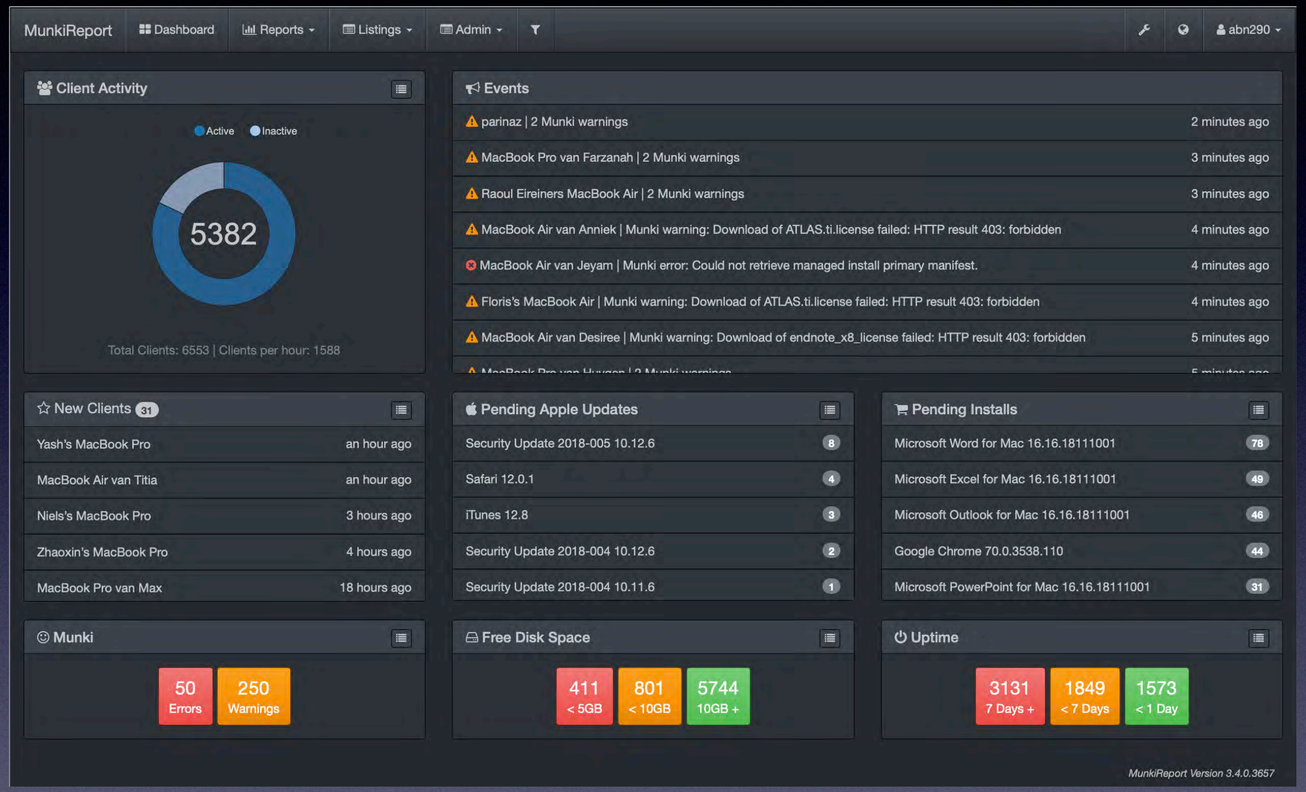
```
Set Variable [ $theCount ; Value: JSONListValues ( MunkiReportQuery::Results ; "data" ) ]
Set Field [ MunkiReportQuery::DetectXclean ; PatternCount ( $theCount ; "Clean" ) ]
Set Field [ MunkiReportQuery::DetectXissues ; PatternCount ( $theCount ; "Issues" ) ]
Set Field [ MunkiReportQuery::DetectXinfected ; PatternCount ( $theCount ; "Infected" ) ]
```

precursor.ca/slides

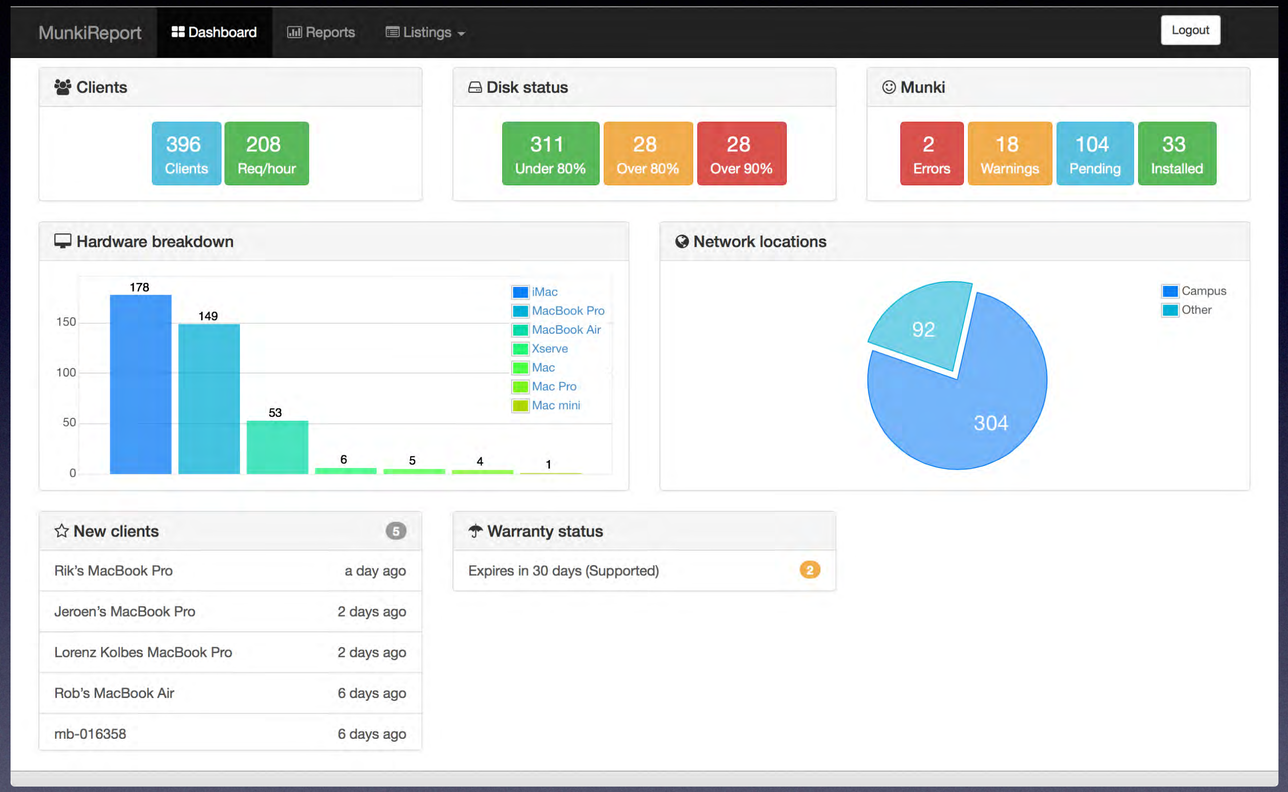
precursor.ca/slides

Managed Software Center

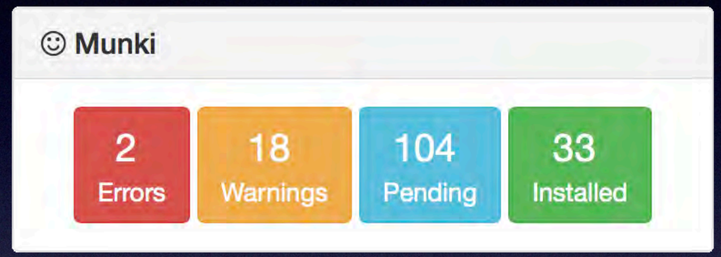
precursor.ca/slides



precursor.ca/slides



precursor.ca/slides



precursor.ca/slides

https://github.com/precursorca/munki-stats-widget

Thanks to Arjen van Bochoven!

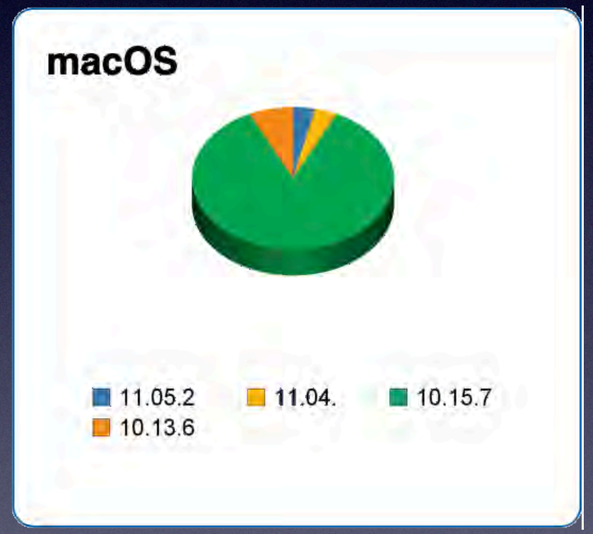
precursor.ca/slides

munkireport.example.com/index.php?//show/dashboard/stats

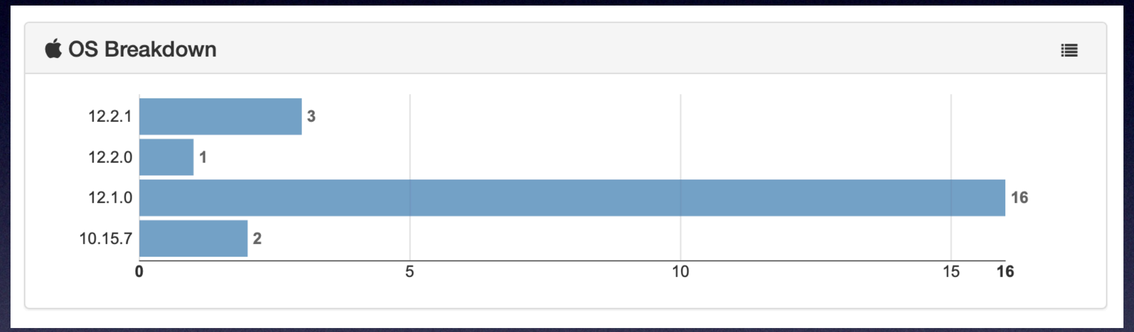
Display as a FileMaker Buttons

precursor.ca/slides

macOS

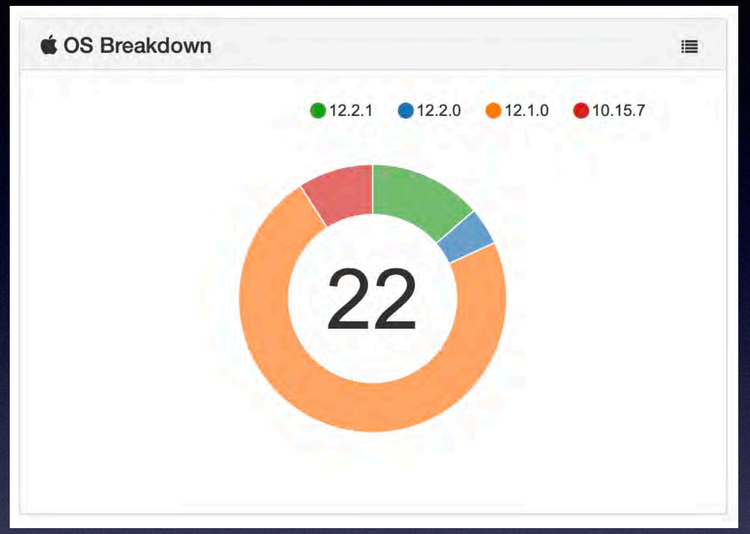


precursor.ca/slides



Munkireport OS Breakdown

precursor.ca/slides

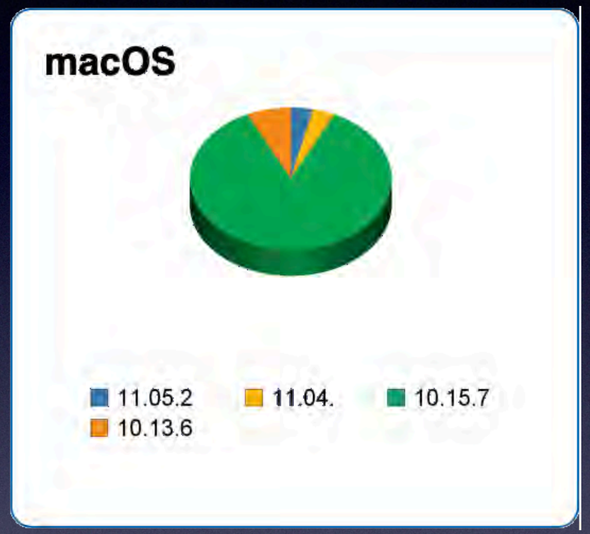


Munkireport OS Breakdown Donut

https://github.com/precursorca/os_donut_widget

precursor.ca/slides

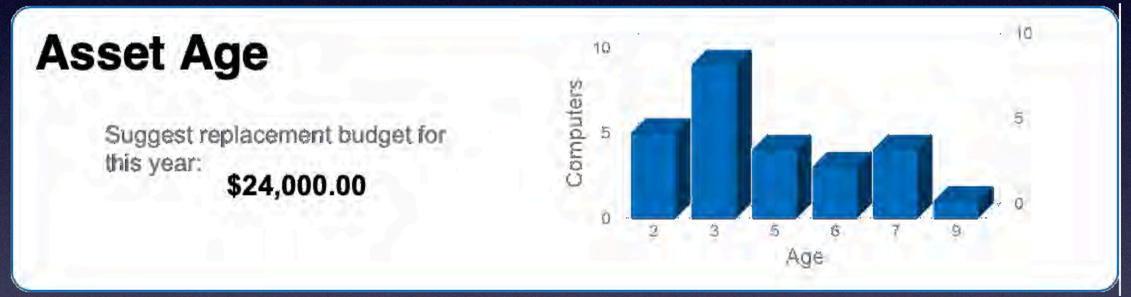
munkireport.example.com/index.php?//module/machine/os



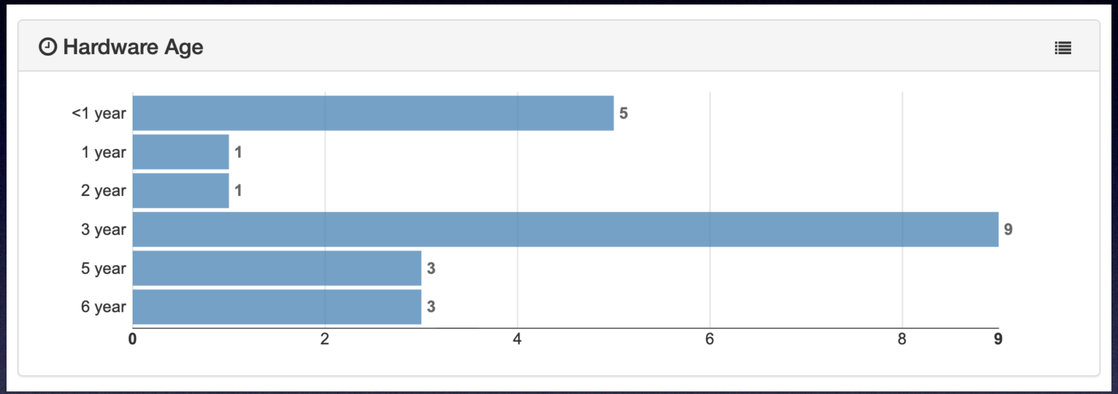
Display as a FileMaker Chart

precursor.ca/slides

Asset Age

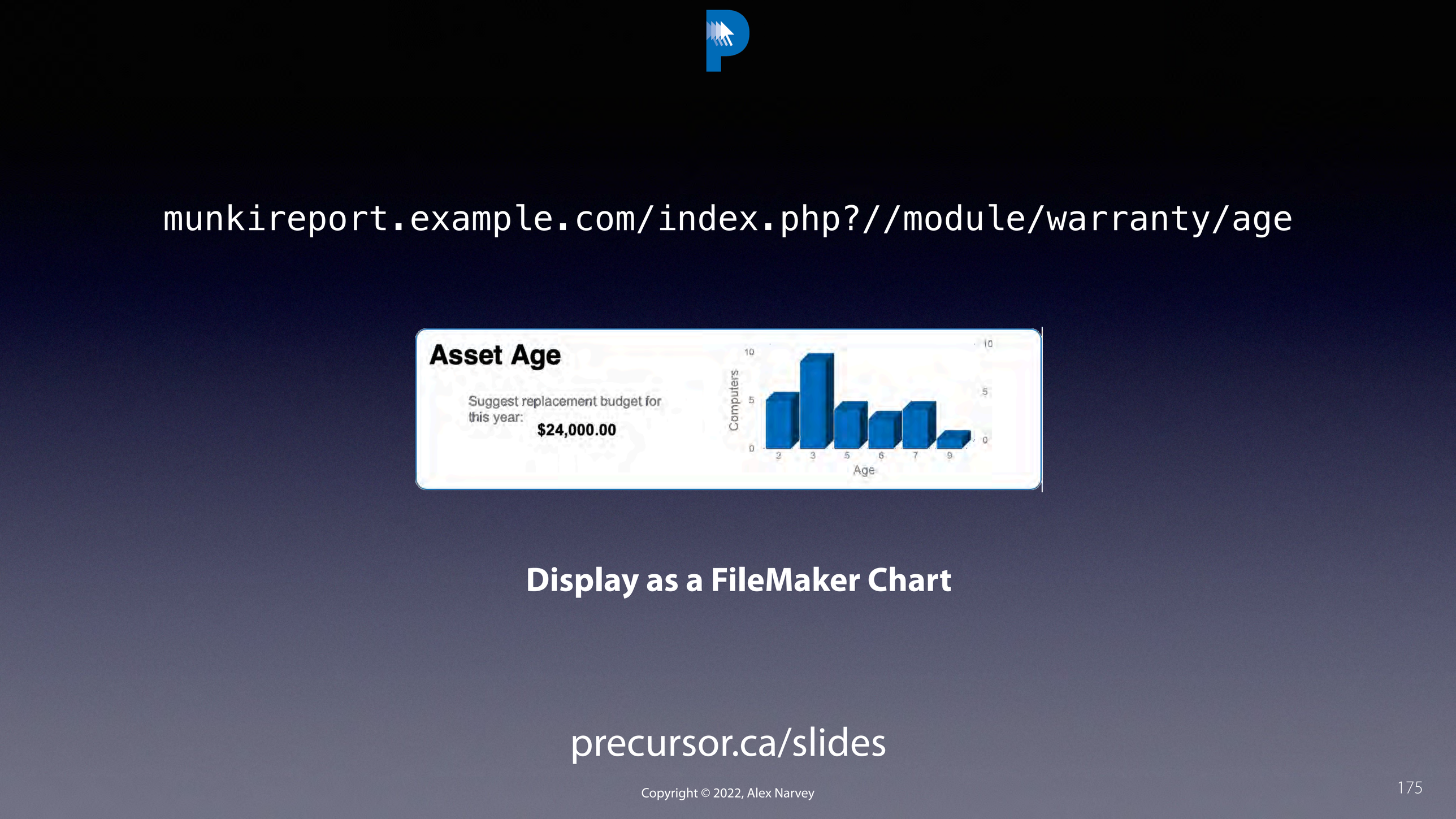


precursor.ca/slides



Munkireport Hardware Age

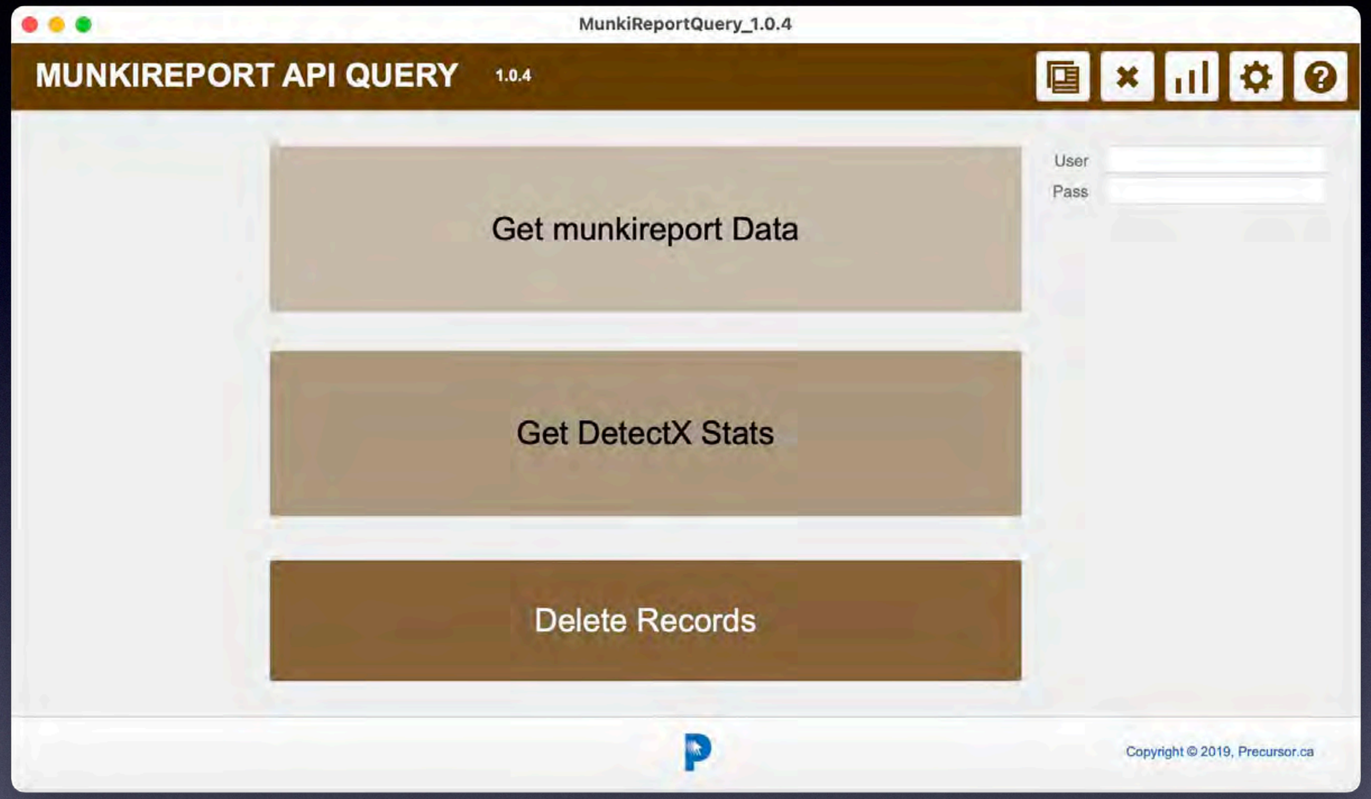
precursor.ca/slides



munkireport.example.com/index.php?//module/warranty/age

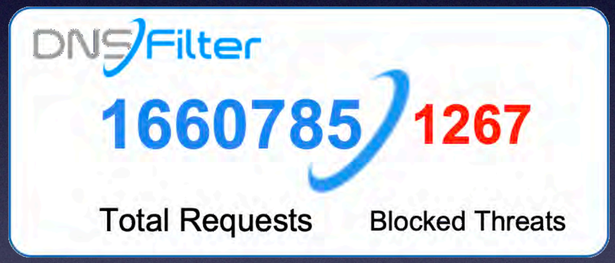
Display as a FileMaker Chart

precursor.ca/slides

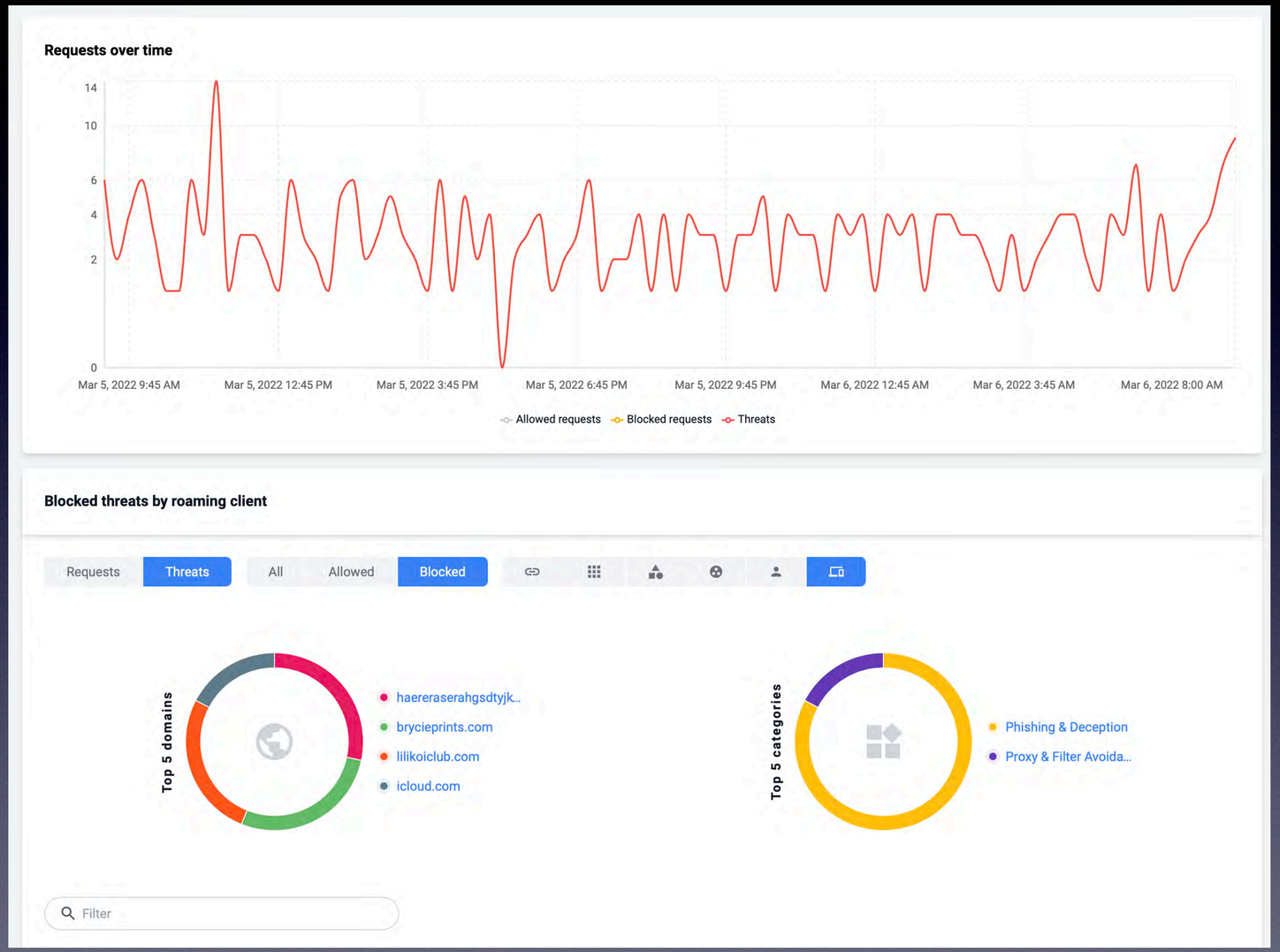


precursor.ca/slides

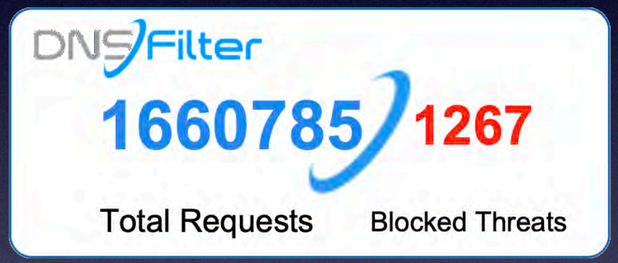
DNSFilter



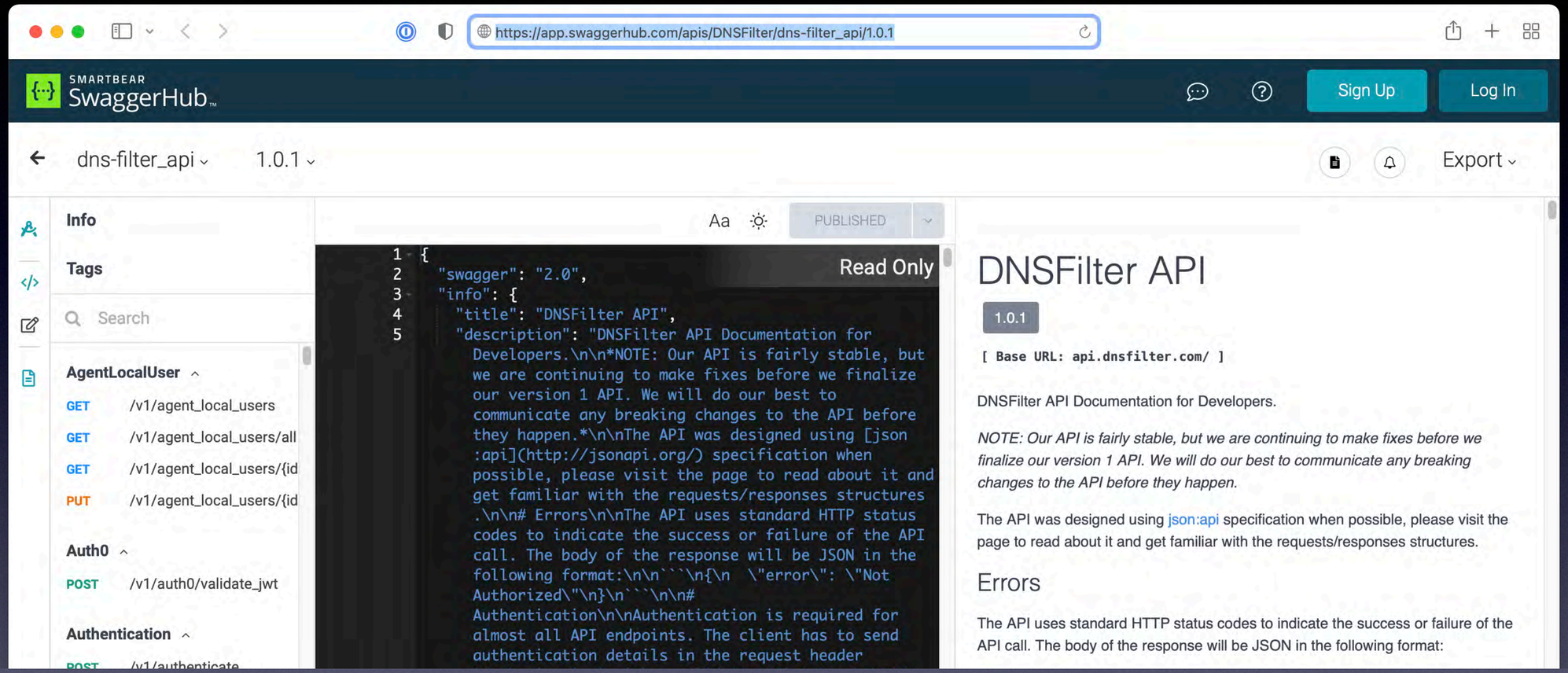
precursor.ca/slides



precursor.ca/slides



precursor.ca/slides



https://app.swaggerhub.com/apis/DNSFilter/dns-filter_api/1.0.1

precursor.ca/slides

1) Obtain a key for authentication

2) Get the Orgization IDs (clients)

3) Set a time period (30 days back)

4) Get total number of requests (aggregate 30 buckets)

5) Get total number of blocked threats (aggregate 30 buckets)

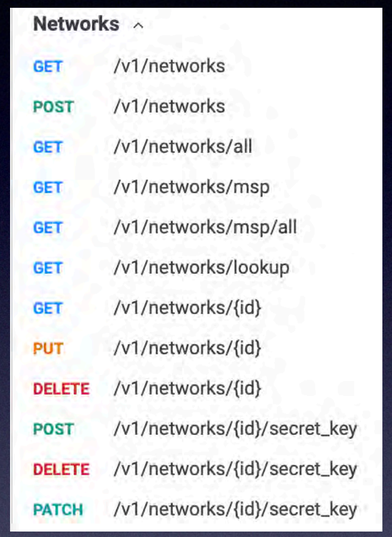
precursor.ca/slides

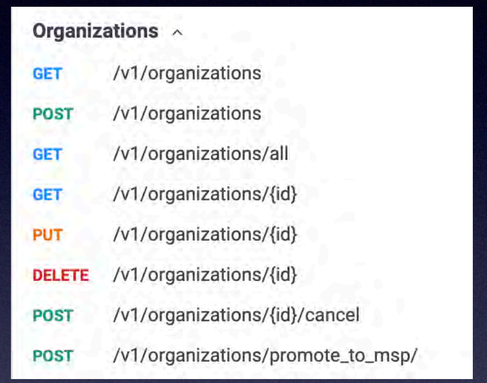
Obtaining a Token

Authentication tokens can be obtained from Auth0 using their Authenticate API `/oauth/token` endpoint with the following parameters:

- `client_id` : `zJ1WJHavuUFx89cConwlipxoOc2J3TVQ`
- `realm` : `Username-Password-Authentication`
- `audience` : `https://dnsfilter.auth0.com/mfa/`
- `grant_type` : `http://auth0.com/oauth/grant-type/password-realm`
- `scope` : `enroll read:authenticators remove:authenticators offline_access openid picture name email`
- `username` : `<your username>`
- `password` : `<your password>`

precursor.ca/slides





https://api.dnsfilter.com/v1/clients

https://api.dnsfilter.com/v1/organizations

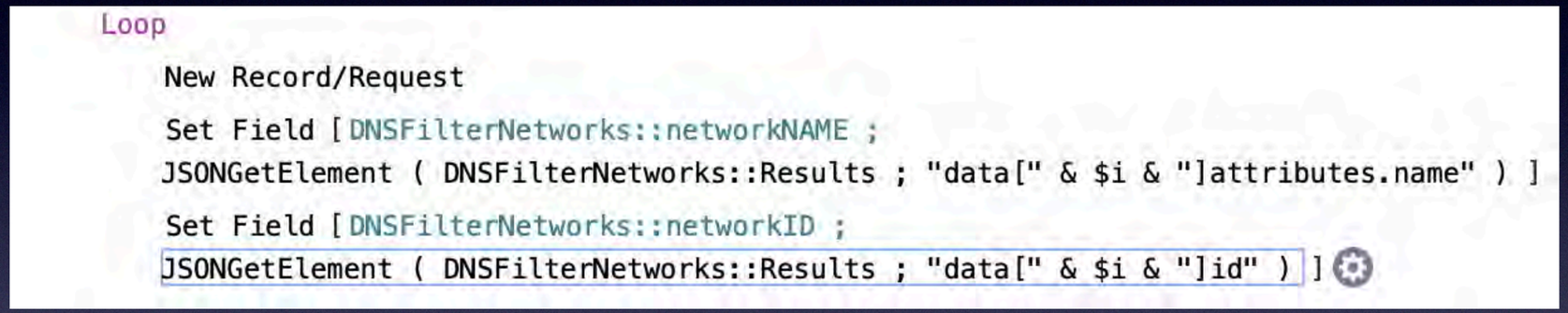
precursor.ca/slides

GET

https://api.dnsfilter.com/v1/organizations


```
{
 "data": [
  {
   "id": "string",
   "type": "string",
   "attributes": {
    "id": 0,
    "name": "string",
    "stripe_customer_id": "string",
    "trial_days": 0,
    "update_payment": true,
    "billing_contact_name": "string",
    "billing_contact_phone": "string",
    "billing_contact_email": "string",
    "address": "string",
```


precursor.ca/slides



precursor.ca/slides

```
"https://api.dnsfilter.com/v1/traffic_reports/total_requests?
organization_ids="
& DNSFilterStats::ORGANIZATION_ID
& "&from="
& $$DNSFilterDATE
& "T01:01:01" & "\""
```

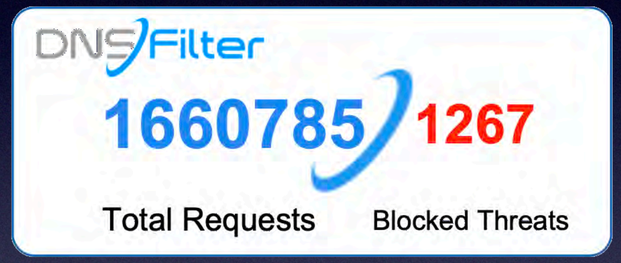
precursor.ca/slides

bucket_size **string** *(query)*	Desired bucket size, possible values: *auto*, *15min*, *1day*, defaults to *auto* *Auto*: If report from/to range is less than 24 hours (ex.: "Today" and "Yesterday"), bucket size is fifteen minutes If report from/to range is greater than 24 hours (ex.: "Last 7 days" and "Last 14 days"), bucket size is one day

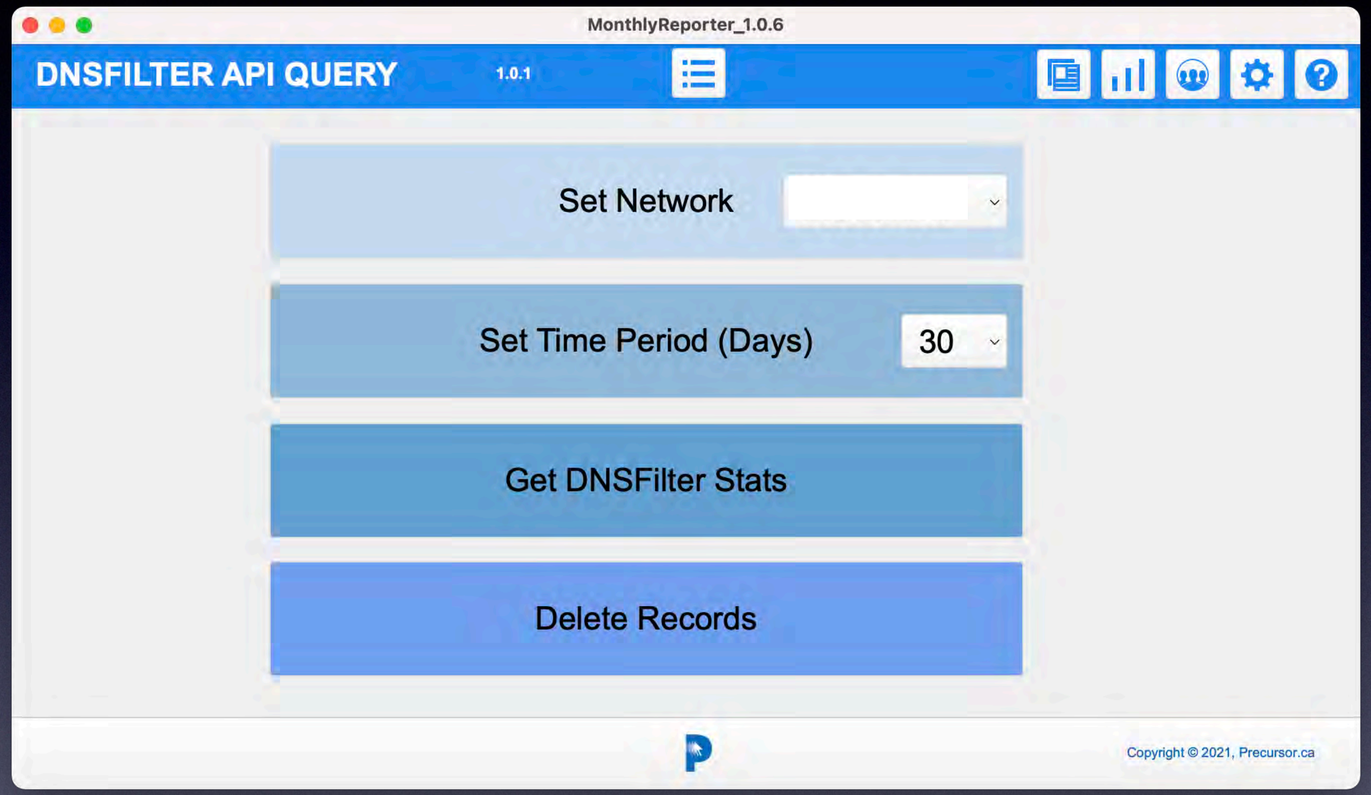
precursor.ca/slides

```
If [ $BucketCount > 0 ]
    Loop
        Set Variable [ $Requests ; Value: $Requests + JSONGetElement ( DNSFilterStats::Results ; "data.values[" & $i & "].total" ) ]
        Set Variable [ $i ; Value: $i + 1 ]
        Exit Loop If [  $i ≥ $BucketCount ]
    End Loop
    Set Field [ DNSFilterStats::TotalRequests ; $Requests ]
```

precursor.ca/slides



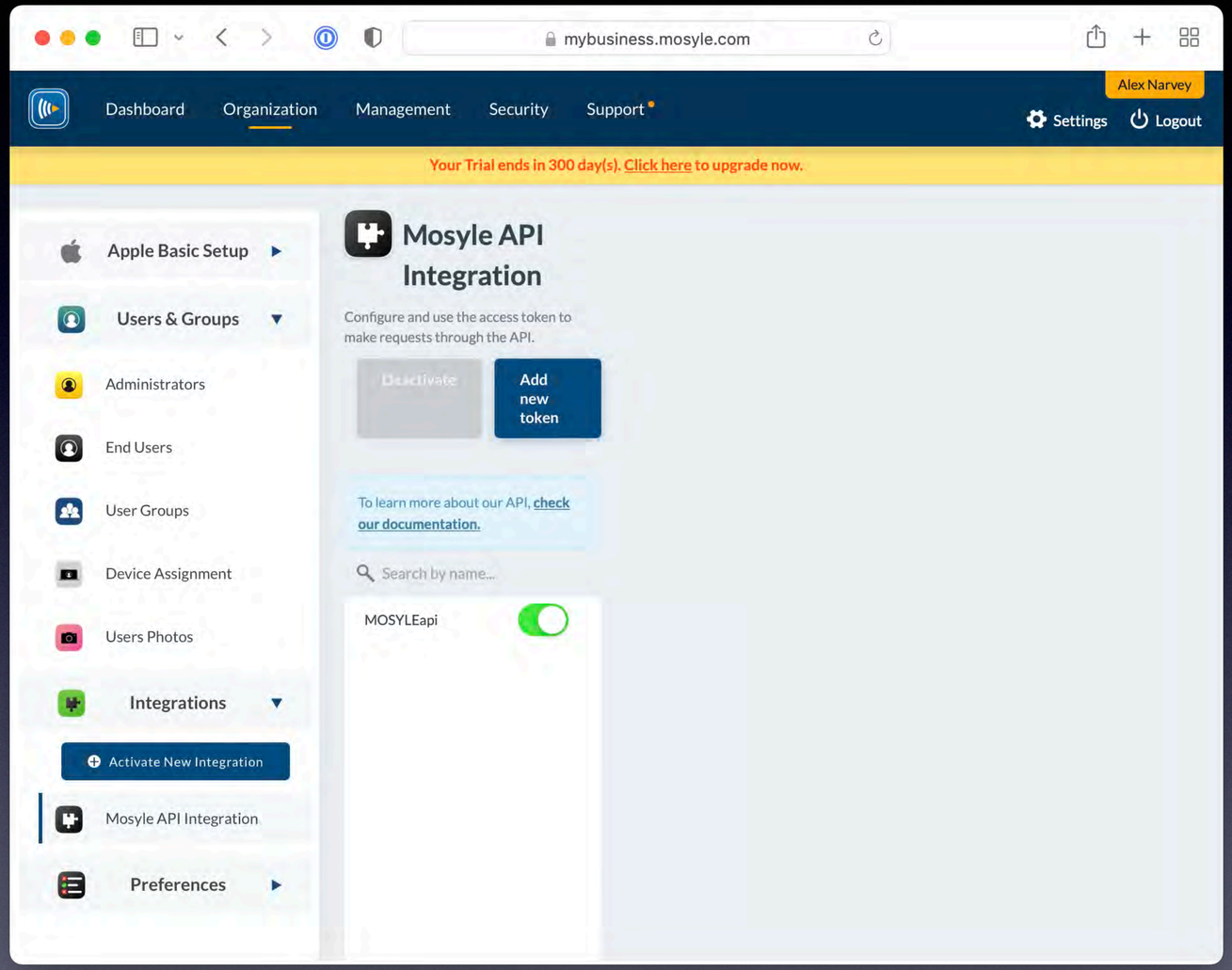
precursor.ca/slides



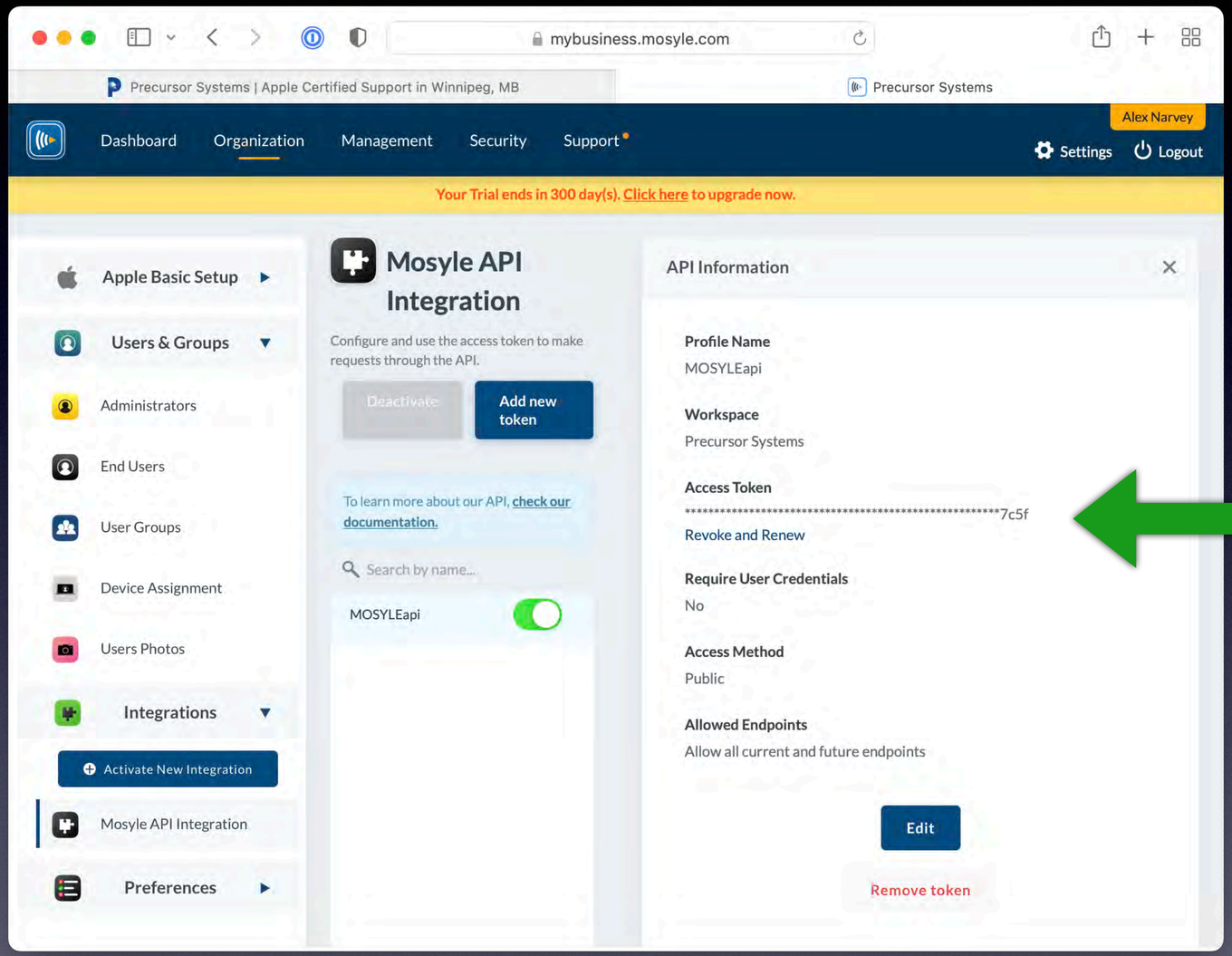
precursor.ca/slides

Mosyle

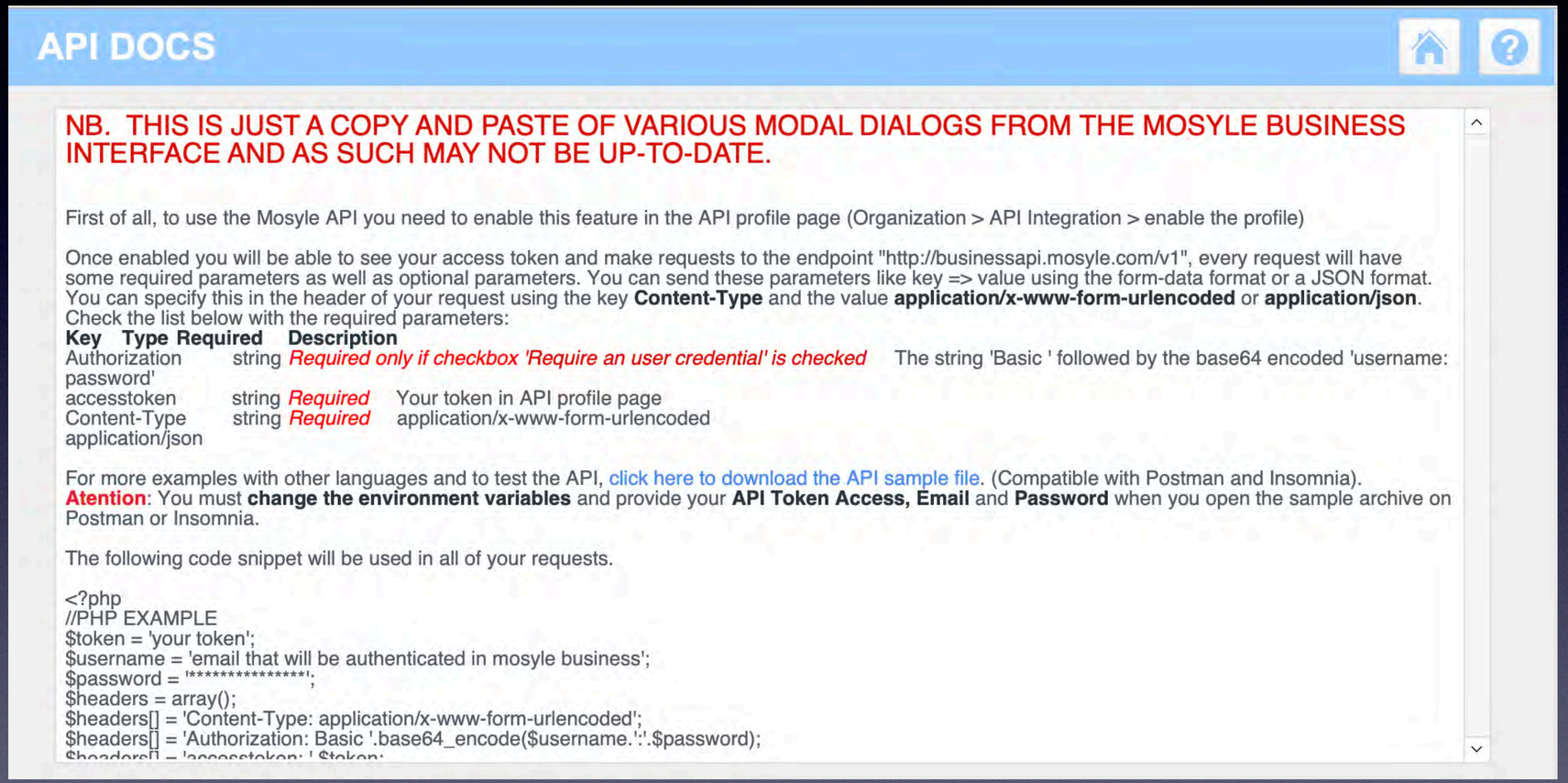
precursor.ca/slides



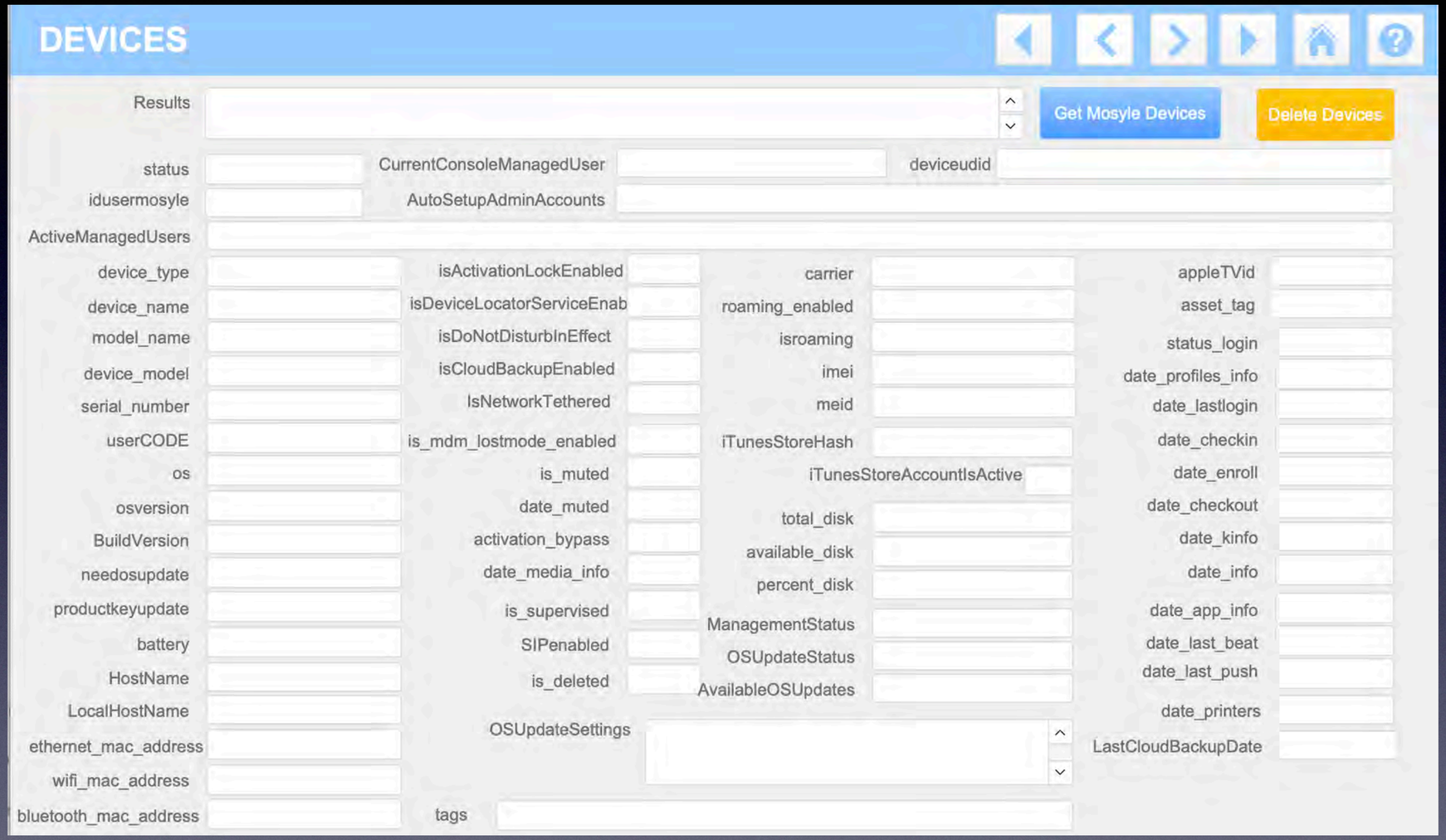
precursor.ca/slides



precursor.ca/slides



precursor.ca/slides

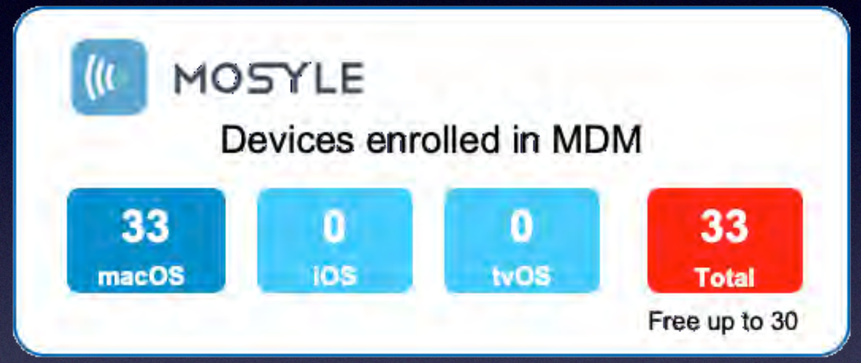


precursor.ca/slides

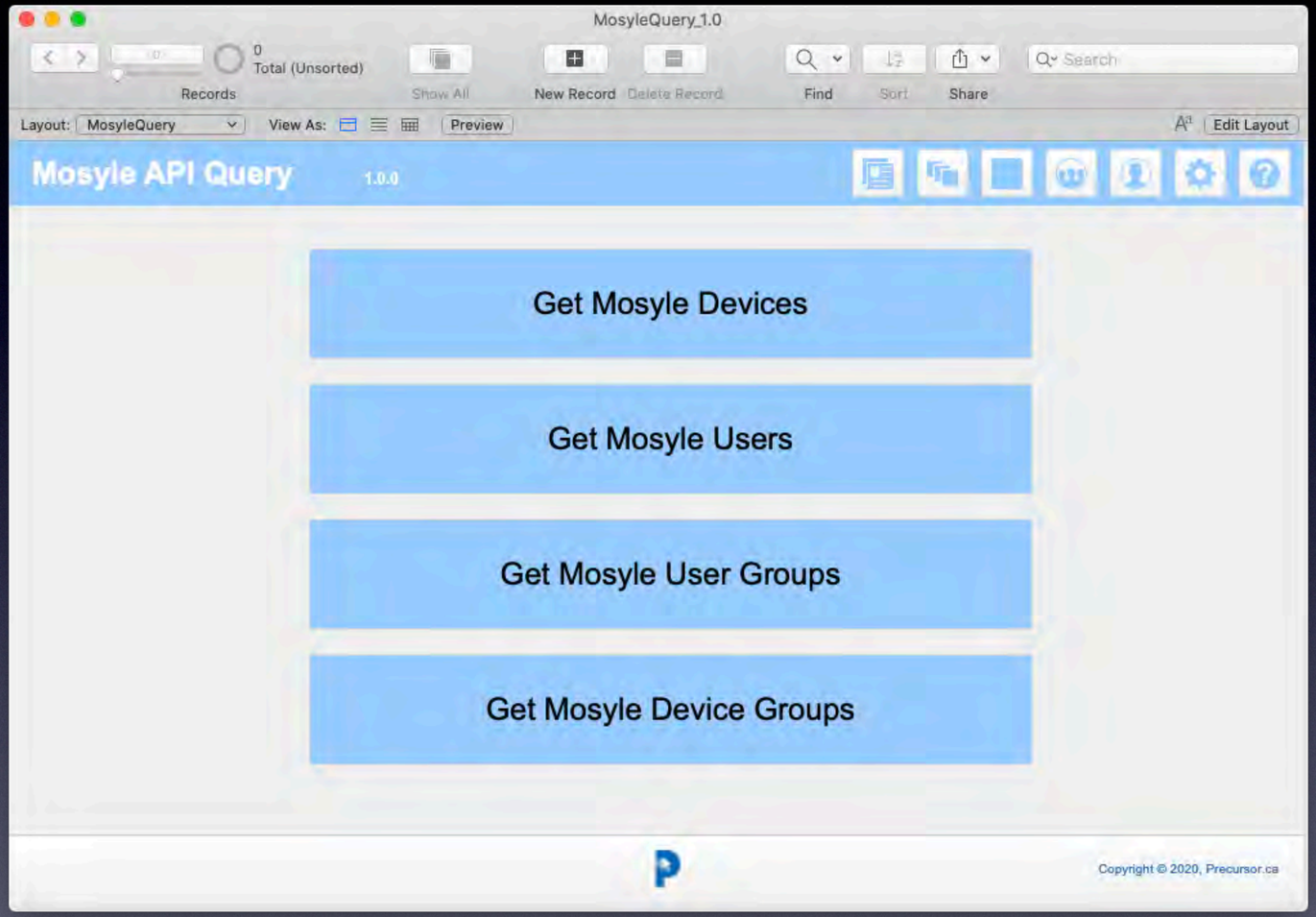
"https://businessapi.mosyle.com/v1/devices"

```
//The cURL details
" -d @$MacDeviceJSON"
//" -d {\"operation\": \"list\",\"options\" : { \"os\": \"mac\"} }"
//" -d {\"operation\": \"list\",\"options\" : { \"os\": \"ios\"} }"
//" -d {\"operation\": \"list\",\"options\" : { \"os\": \"tvos\"} }"
```

precursor.ca/slides

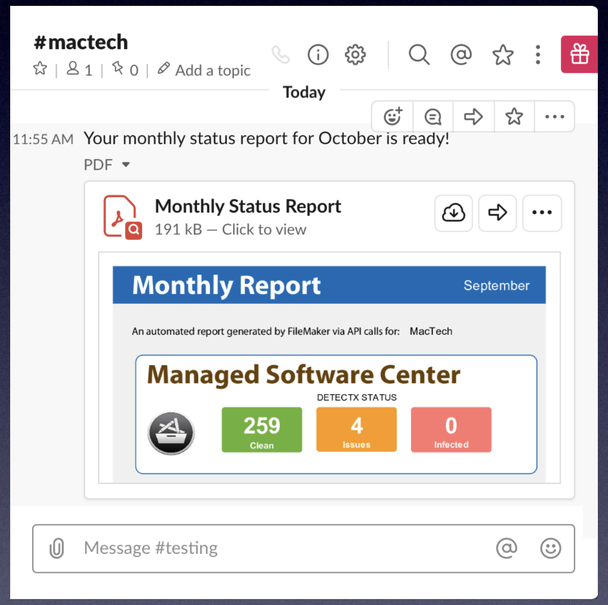


precursor.ca/slides



precursor.ca/slides

Slack

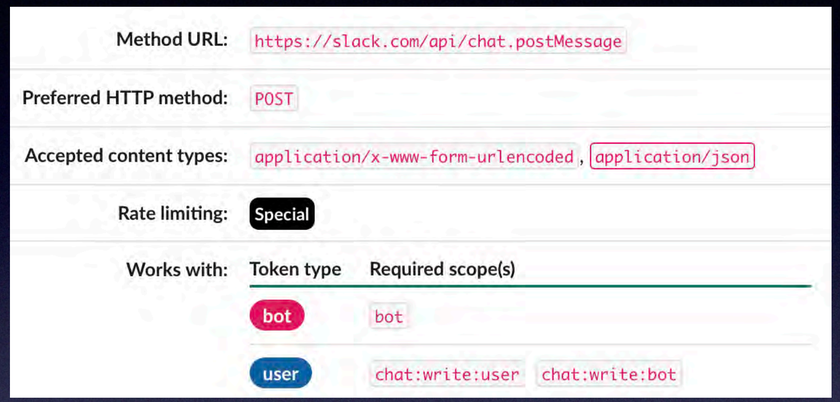


precursor.ca/slides

- Web API
- Events API
- Conversations API
- Real Time Messaging API
- Methods
- Object Types
- Event Types
- SCIM API
- Audit Logs API

`https://api.slack.com`

precursor.ca/slides

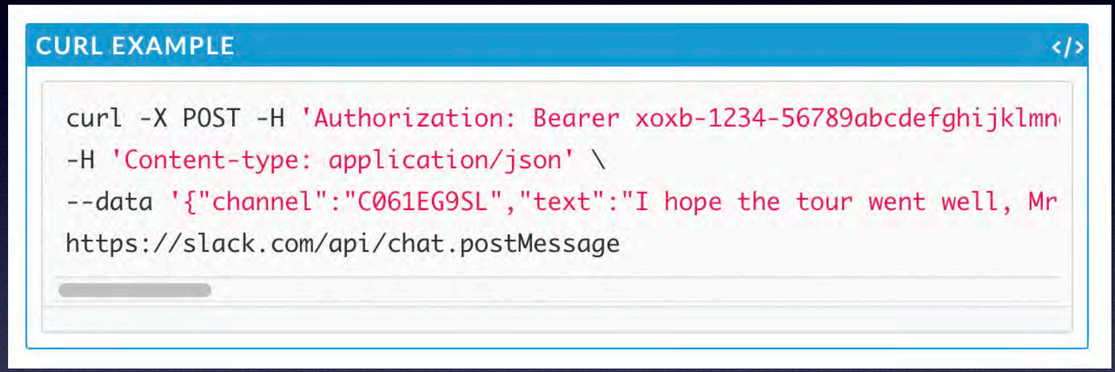


https://api.slack.com/methods/chat.postMessage

precursor.ca/slides

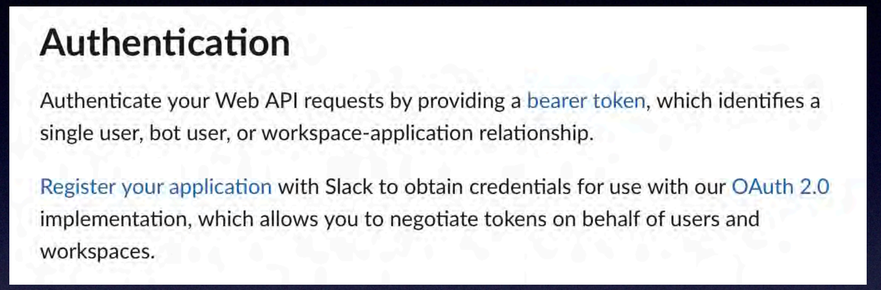
Argument	Example	Required	Description
`token`	`xxxx-xxxxxxxxx-xxxx`	Required	Authentication token bearing required scopes.
`channel`	`C1234567890`	Required	Channel, private group, or IM channel to send message to. Can be an encoded ID, or a name. See below for more details.
`text`	`Hello world`	Required	Text of the message to send. See below for an explanation of formatting. This field is usually required, unless you're providing only `attachments` instead. Provide no more than 40,000 characters or risk truncation.

precursor.ca/slides



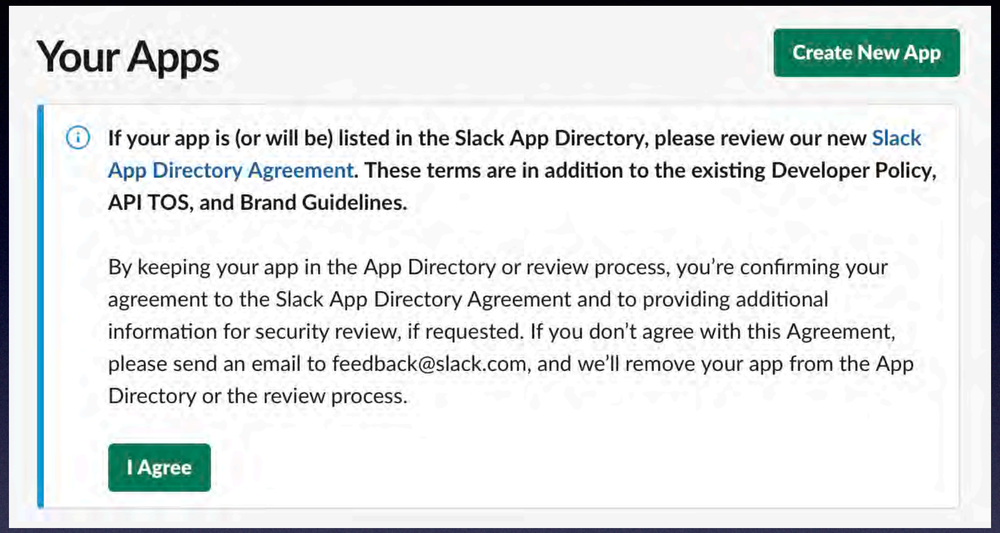
https://api.slack.com/web#posting_json

precursor.ca/slides



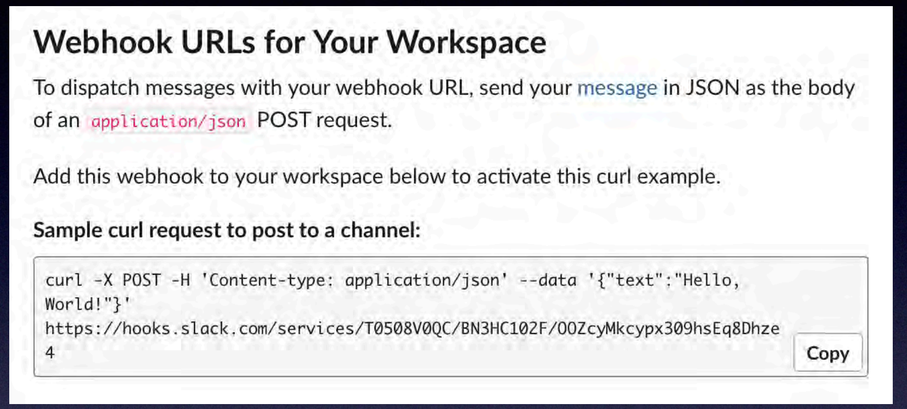
`https://api.slack.com/web#posting_json`

precursor.ca/slides



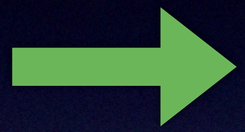
`https://api.slack.com/apps`

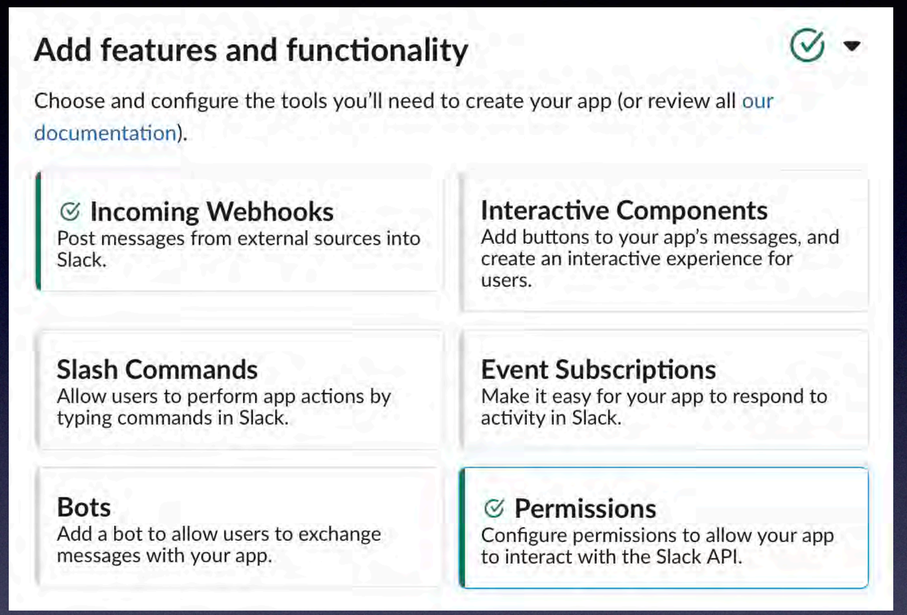
precursor.ca/slides



https://api.slack.com/incoming-webhooks

precursor.ca/slides





`https://api.slack.com/methods/chat.postMessage`

precursor.ca/slides

App Credentials

These credentials allow your app to access the Slack API. They are secret. Please don't share your app credentials with anyone, include them in public code repositories, or store them in insecure ways.

App ID

[redacted]

Date of App Creation

August 25, 2019

Client ID

50[redacted].7[redacted]

Client Secret

•••••••••• Show Regenerate

You'll need to send this secret along with your client ID when making your oauth.access request.

Signing Secret

•••••••••• Show Regenerate

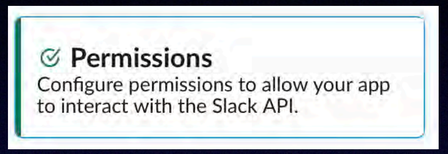
Slack signs the requests we send you using this secret. Confirm that each request comes from Slack by verifying its unique signature.

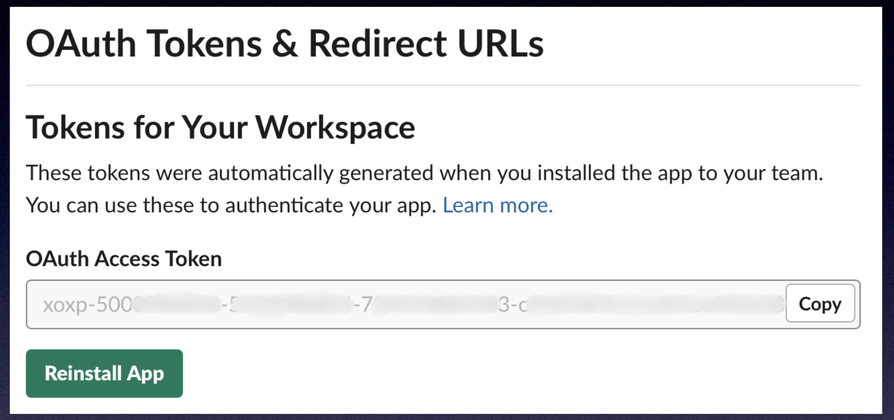
Verification Token

LU[redacted] Regenerate

This deprecated Verification Token can still be used to verify that requests come from Slack, but we strongly recommend using the above, more secure, signing secret instead.

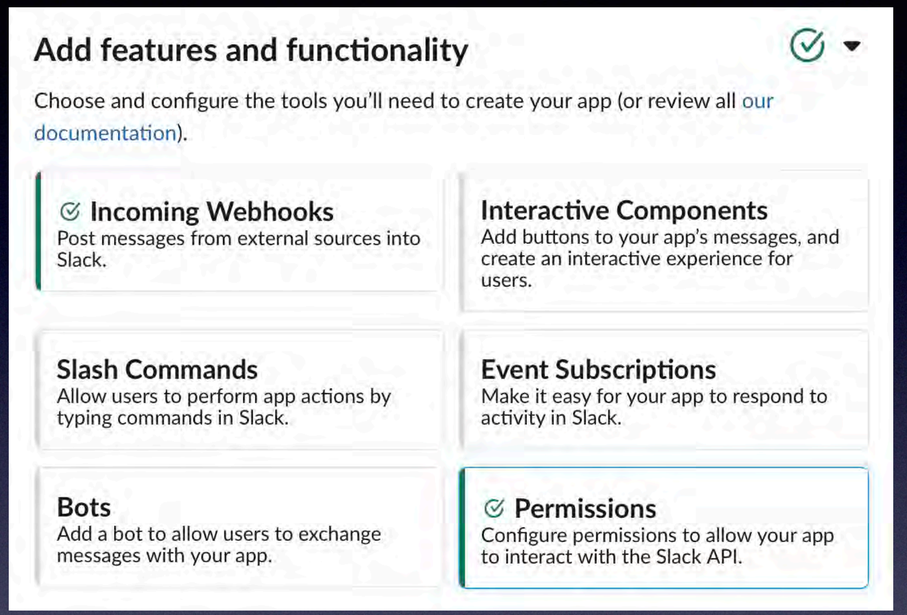
precursor.ca/slides





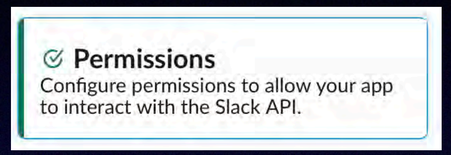
https://api.slack.com/docs/oauth

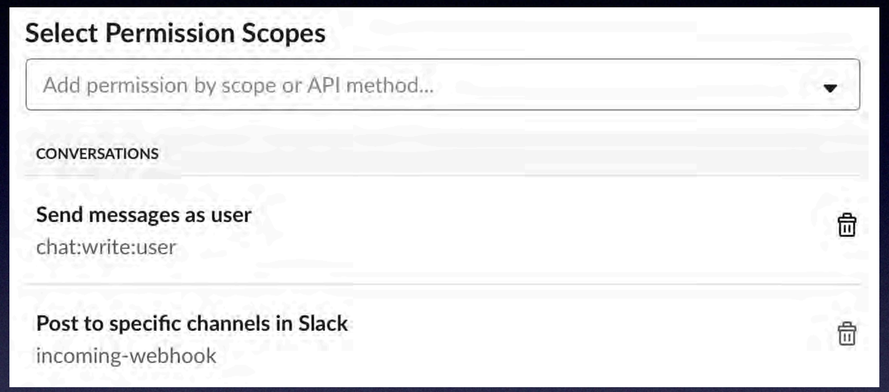
precursor.ca/slides



`https://api.slack.com/methods/chat.postMessage`

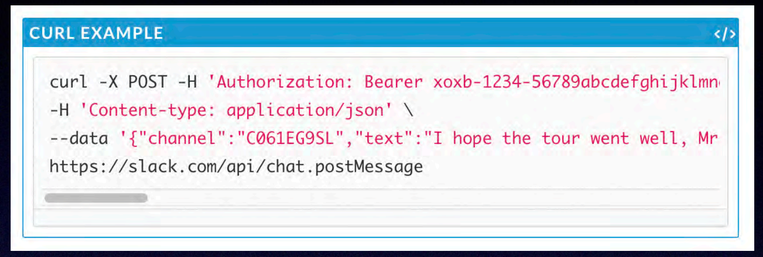
precursor.ca/slides

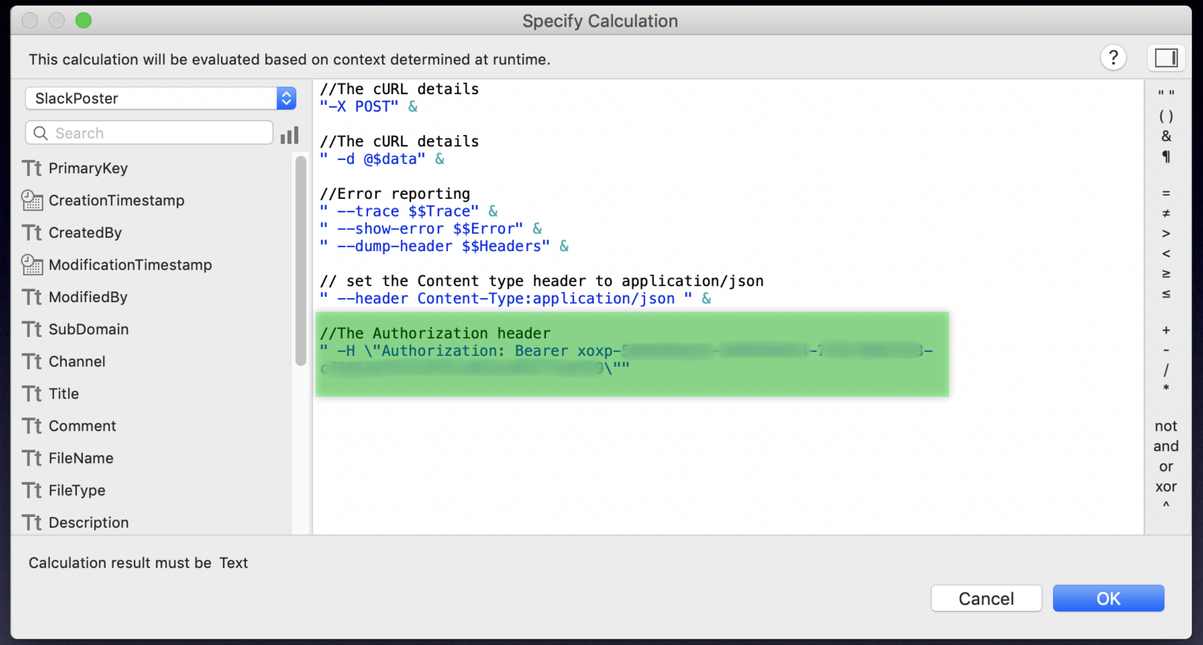




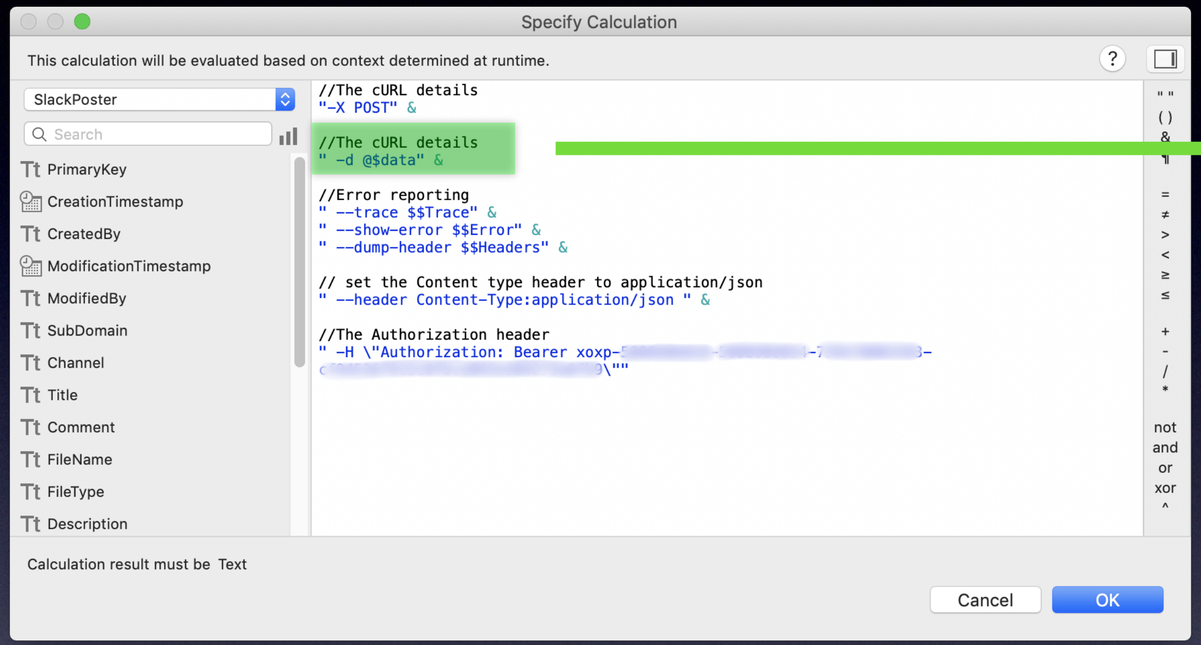
https://api.slack.com/docs/oauth-scopes

precursor.ca/slides

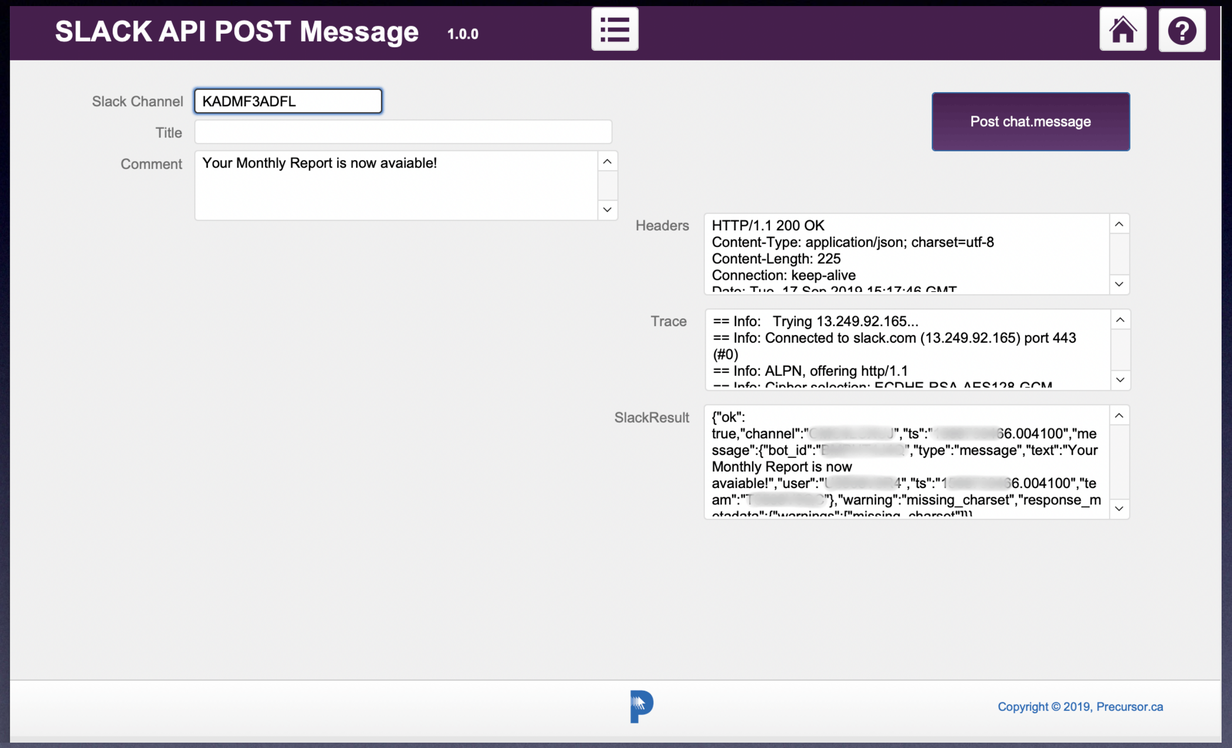




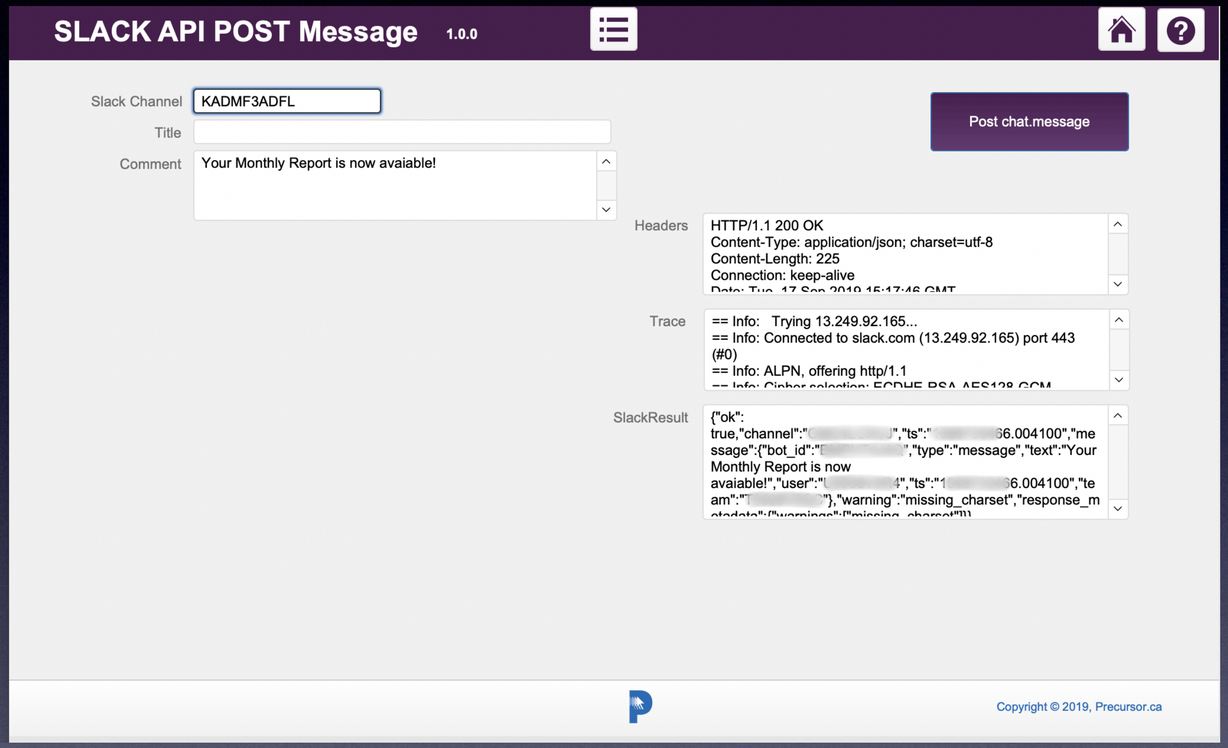
precursor.ca/slides

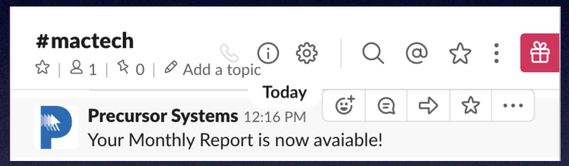


precursor.ca/slides

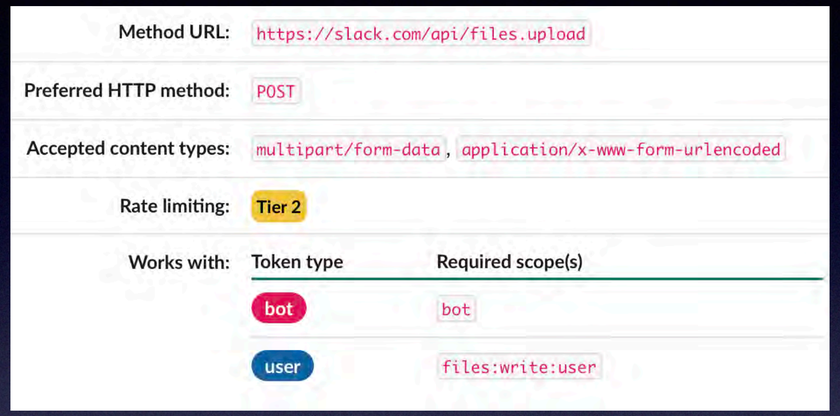


precursor.ca/slides



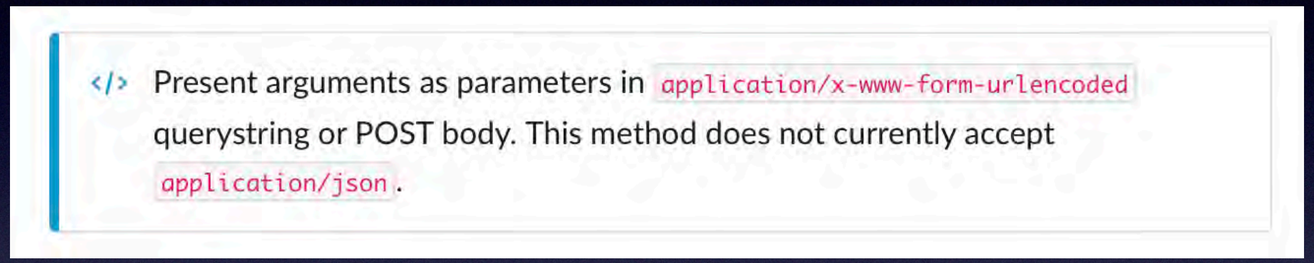


precursor.ca/slides



https://api.slack.com/methods/files.upload

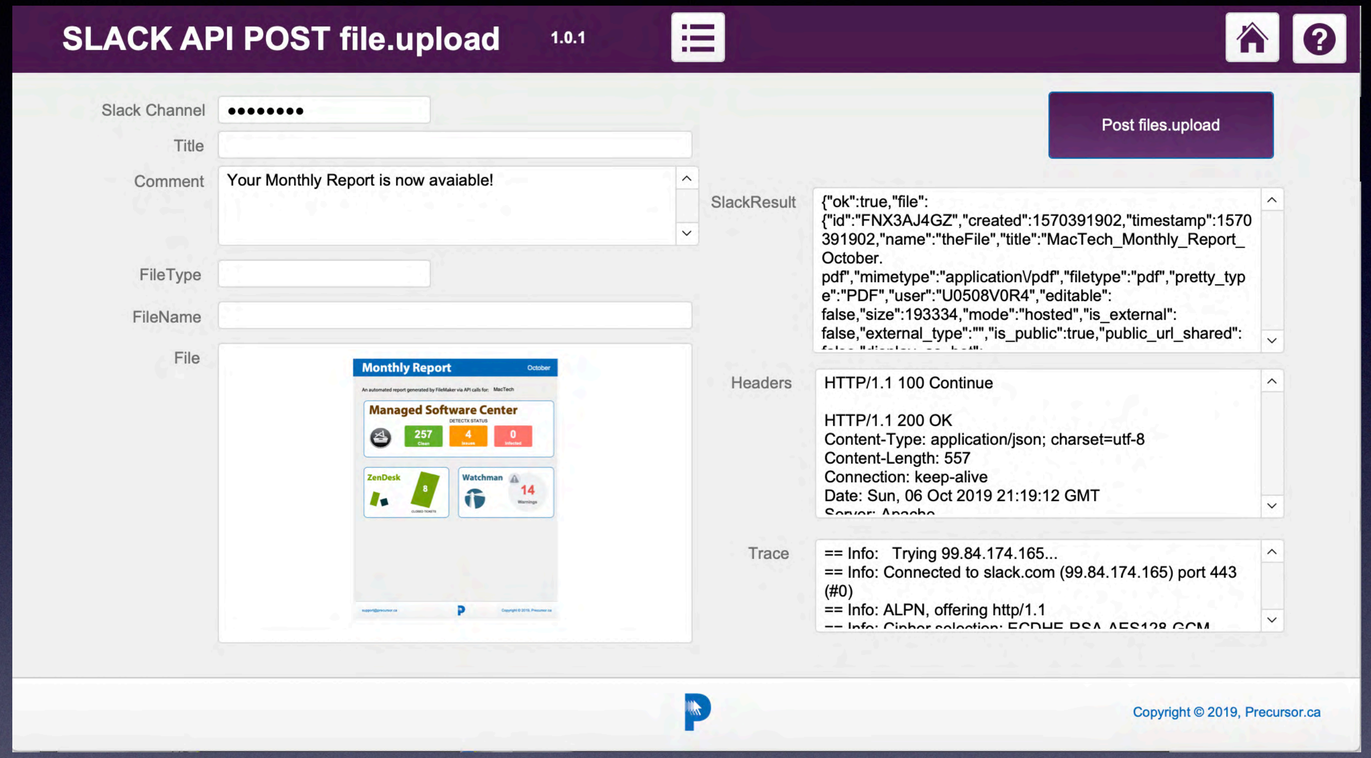
precursor.ca/slides



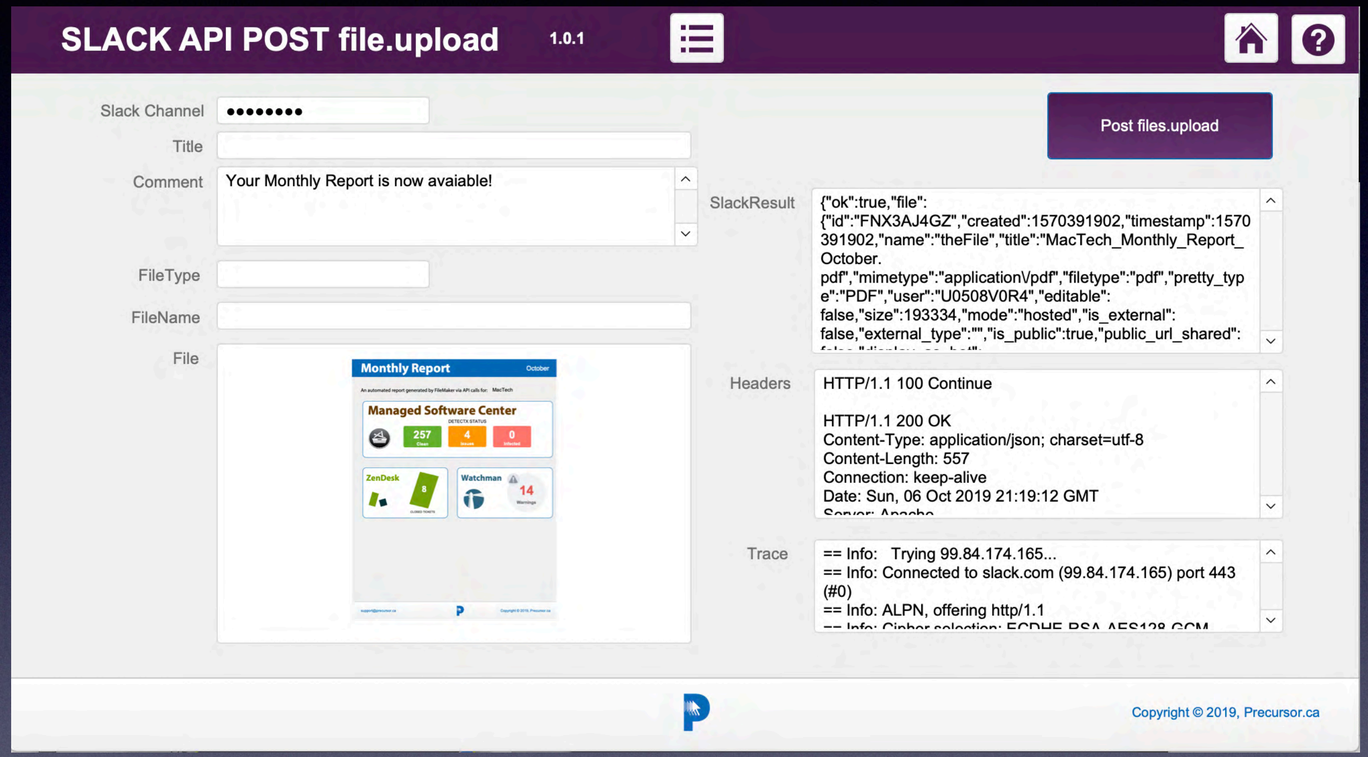
precursor.ca/slides

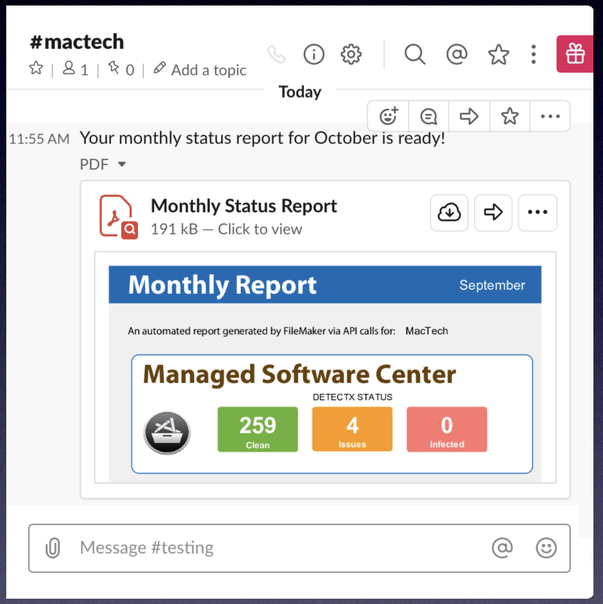
```
curl -F "file=@$FilePath" -F "initial_comment=Your monthly status report is ready!"
-F "title=Monthly Status Report" -F "filename=Monthly_Report.pdf" -F channels=CMC4LCXUJ
-H "Authorization: Bearer xoxp-1234567890-[redacted]-453627128-[redacted]788372829f9"
https://slack.com/api/files.upload
```

precursor.ca/slides

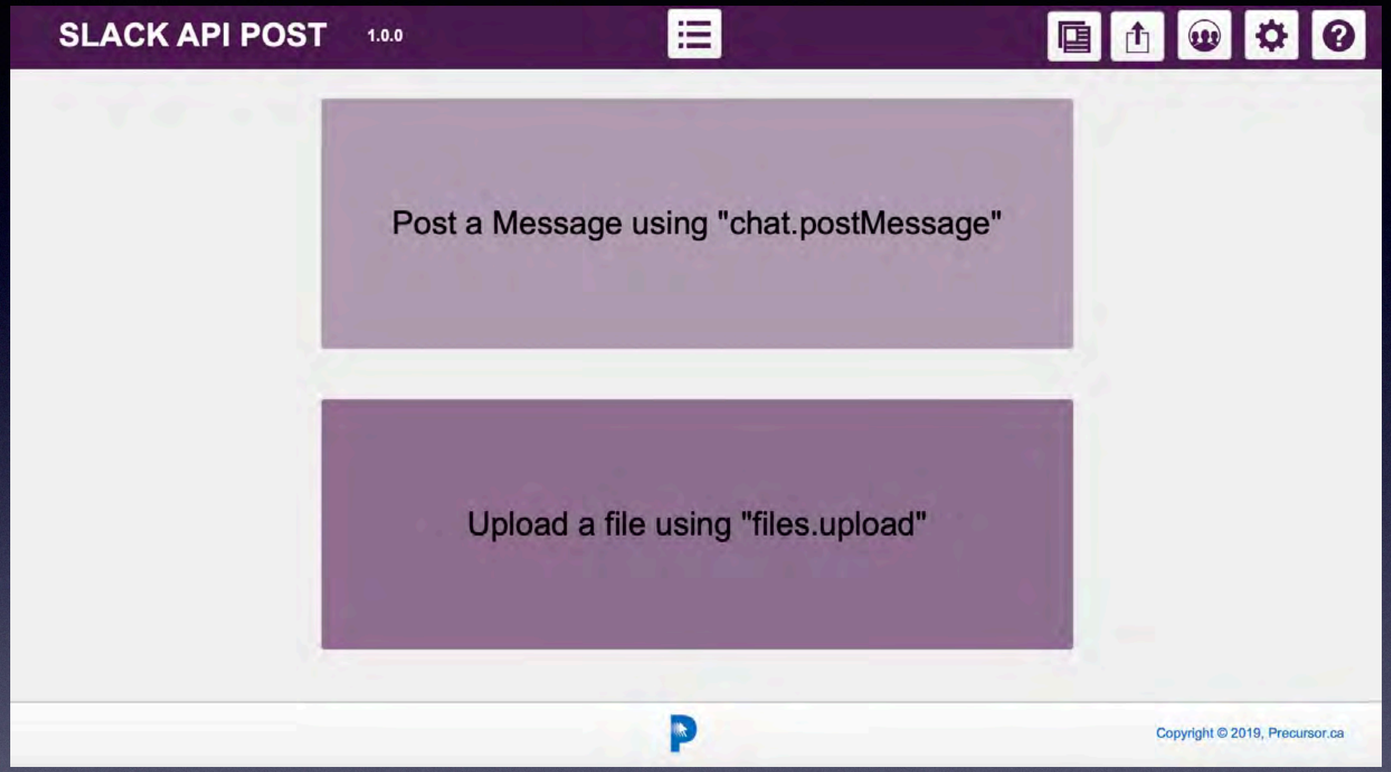


precursor.ca/slides





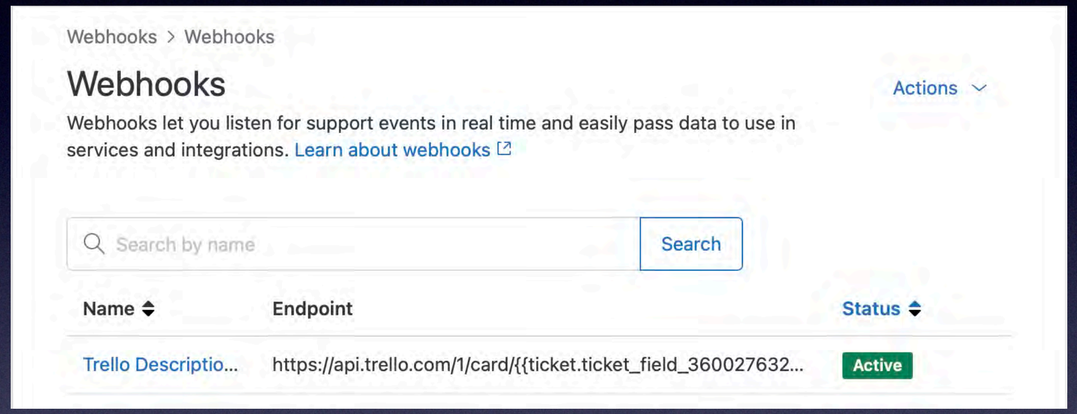
precursor.ca/slides



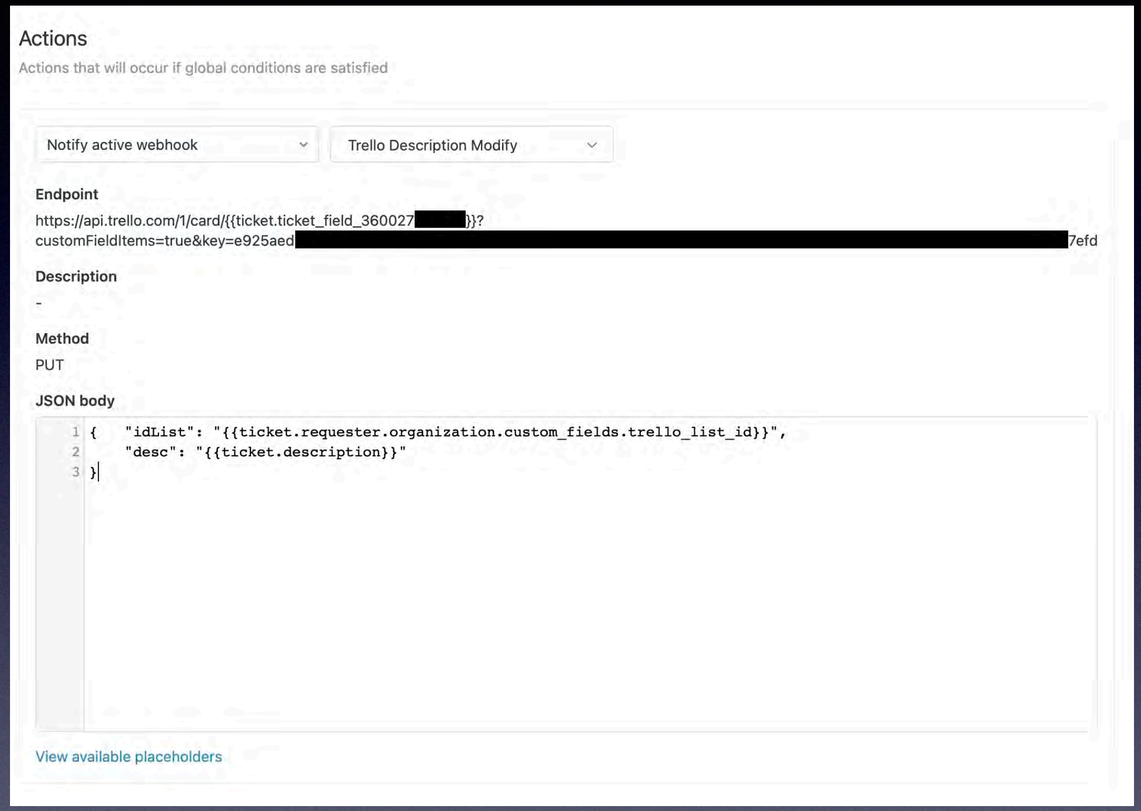
precursor.ca/slides

Triggers & Webhooks

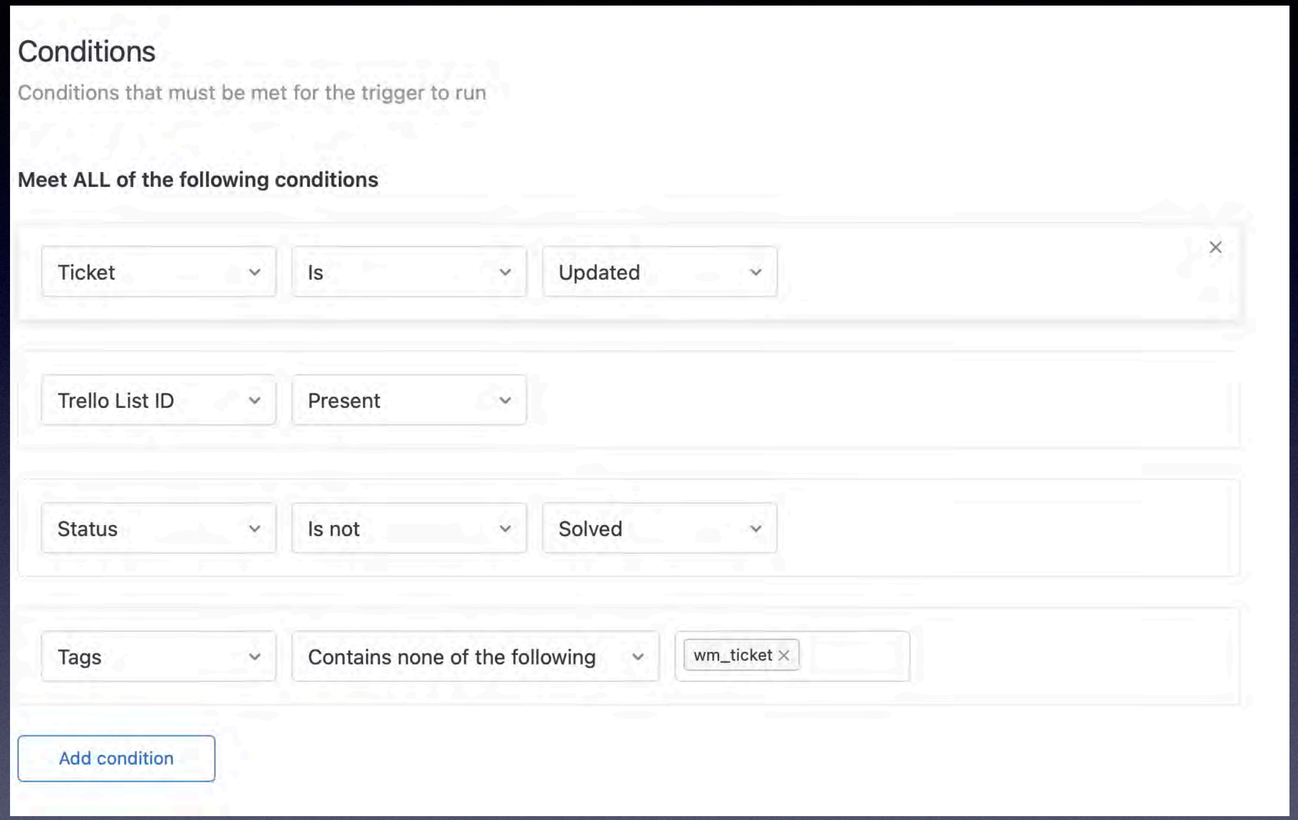
precursor.ca/slides



precursor.ca/slides



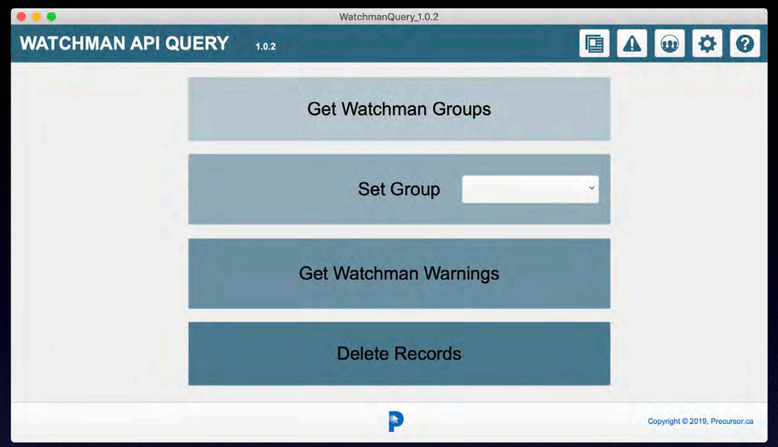
precursor.ca/slides

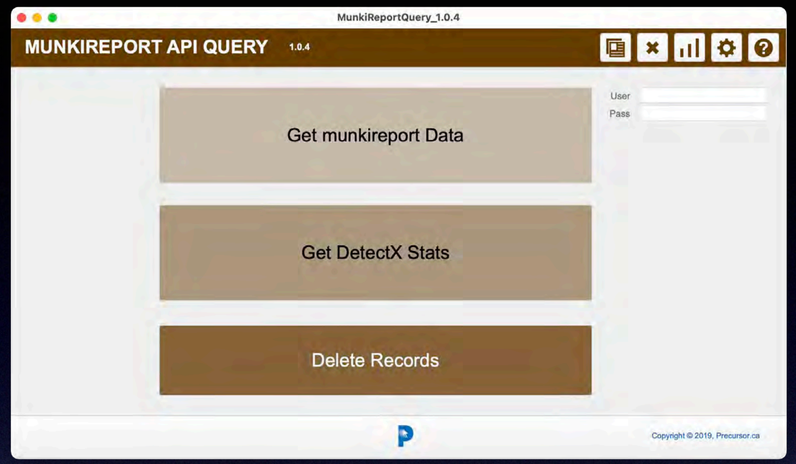


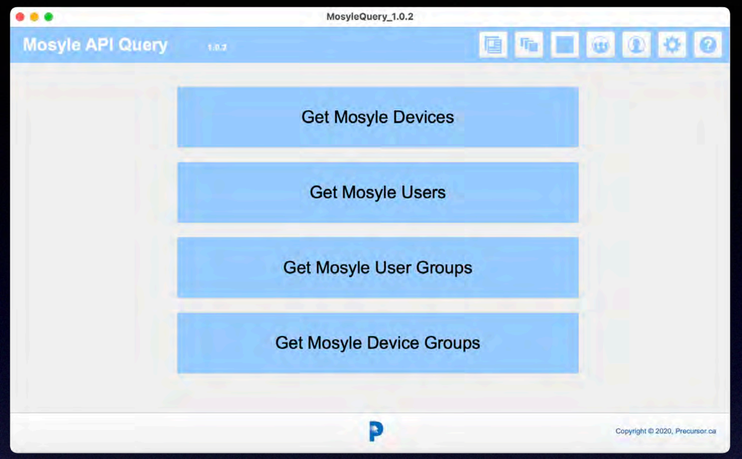
precursor.ca/slides

Templates

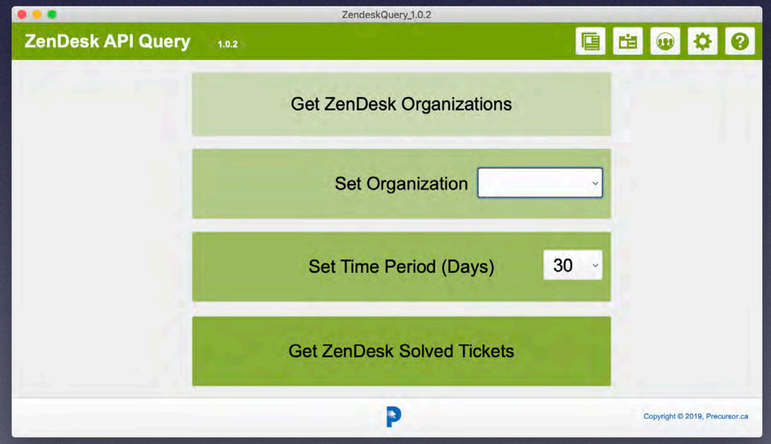
precursor.ca/slides

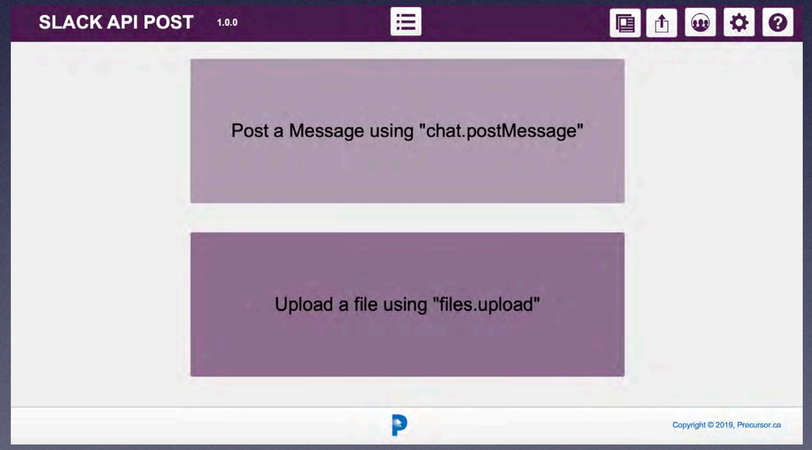


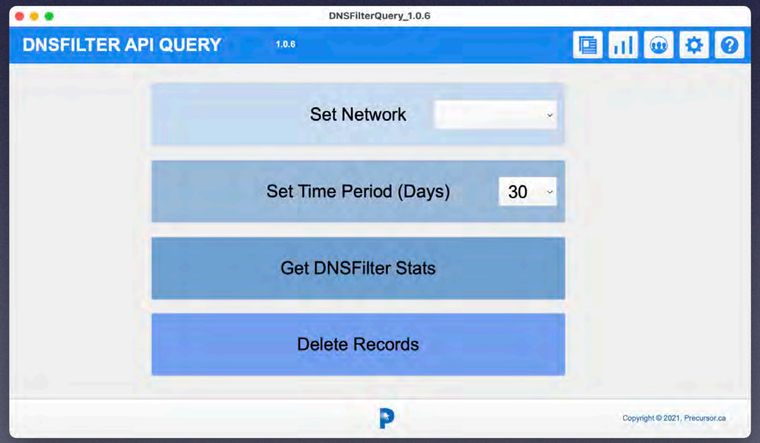




https://precursor.ca/APItemplates







precursor.ca/slides

Claris FileMaker

45 day demo
https://www.claris.com/trial/

ACN FileMaker NFR
https://consultants.apple.com

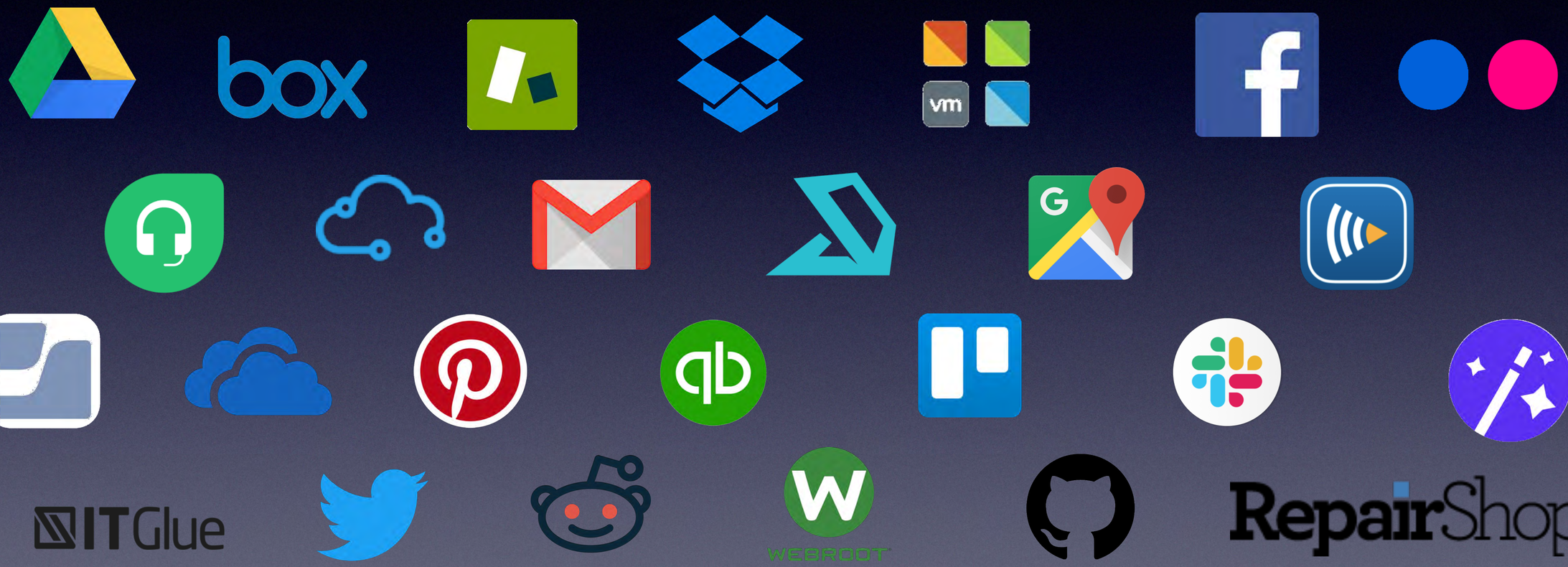
Claris Engage
https://community.claris.com/en/s/engage

FileMaker Partners (free to partners)
https://www.claris.com/partners/become-a-partner.html

Purchase at Retail
https://store.claris.com

precursor.ca/slides

REST APIs



precursor.ca/slides

Alex Narvey

https://Precursor.ca

@precursorca

precursor.ca/APItemplates

precursor.ca/slides