



THE LOANPRO API

INTRODUCTION

loanprosoftware.simnang.com

API RESOURCES

1. Plunker - plnkr.co/users/simnangsupport
2. Articles - help-loanpro.simnang.com/article-categories/api
3. Simnang Forums - help-loanpro.simnang.com/community-forums
4. SDKs - github.com/autopalsoftware
5. If you are in a live workshop event, please raise your hand if you have a question!

PLUNKER – API SAMPLES

API samples are stored on Plunker

Examples showing API use are found on [Plunker](#)

- Examples use AngularJS
- Pre-set with demo credentials

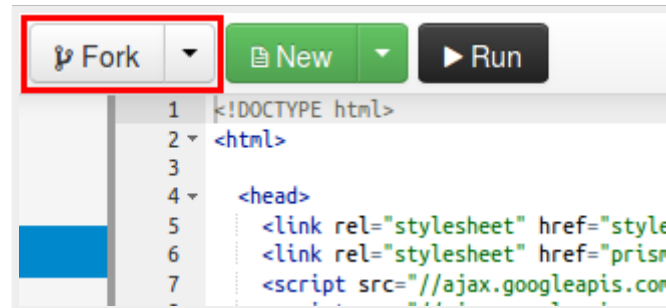
PLUNKER – INTRODUCTION

Plunker is used as a learning environment for the LoanPro API.

- Plunker lets you code in the browser
- Plunker lets you fork samples and download code
- Plunker lets multiple people work on the same project

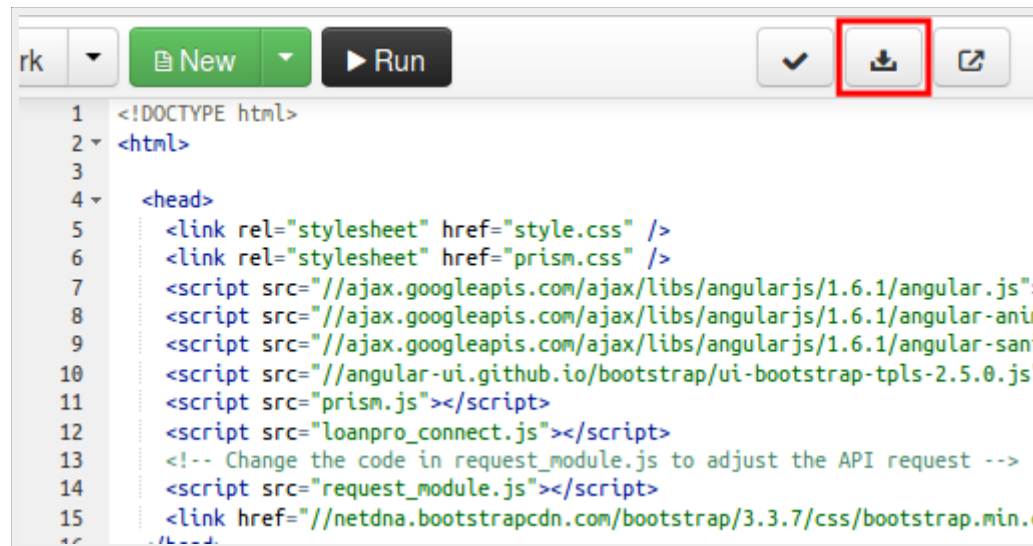
PLUNKER – FORKING A PLUNK

Plunker lets you fork a plunk to use in your own code



PLUNKER – DOWNLOAD YOUR PROJECT

Download a zip with the entire plunk by clicking the Download icon



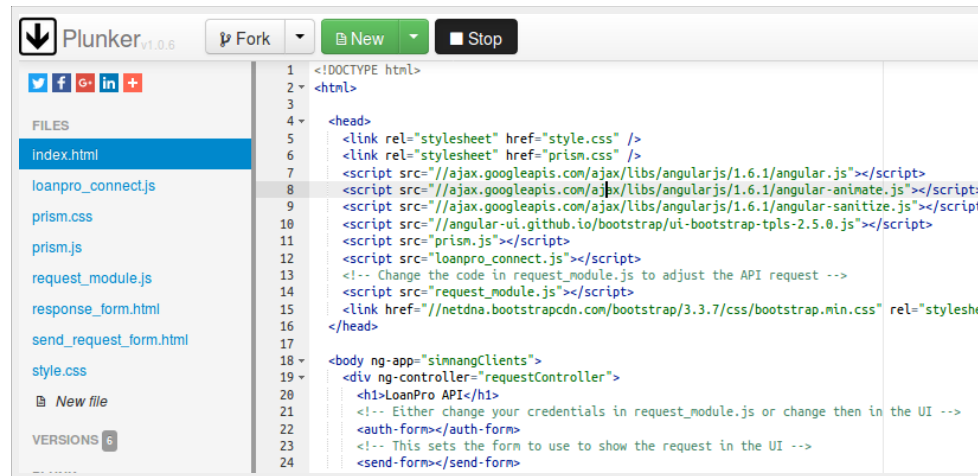
```
rk [New] [Run] [Check] [Download] [Share]
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <link rel="stylesheet" href="style.css" />
6   <link rel="stylesheet" href="prism.css" />
7   <script src="//ajax.googleapis.com/ajax/libs/angularjs/1.6.1/angular.js">
8   <script src="//ajax.googleapis.com/ajax/libs/angularjs/1.6.1/angular-anim
9   <script src="//ajax.googleapis.com/ajax/libs/angularjs/1.6.1/angular-sani
10  <script src="//angular-ui.github.io/bootstrap/ui-bootstrap-tpls-2.5.0.js"
11  <script src="prism.js"></script>
12  <script src="loanpro_connect.js"></script>
13  <!-- Change the code in request_module.js to adjust the API request -->
14  <script src="request_module.js"></script>
15  <link href="//netdna.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.c
16  </head>
```

CAUTIONARY NOTE

The provided API samples store API credentials in the browser

Use the API samples to explore the API and not as your integration

API SAMPLE STRUCTURE



The image shows a screenshot of the Plunker editor interface. The top bar includes the Plunker logo, version 1.0.6, and buttons for 'Fork', 'New', and 'Stop'. On the left, there is a 'FILES' sidebar with a list of files: index.html (selected), loanpro_connect.js, prism.css, prism.js, request_module.js, response_form.html, send_request_form.html, and style.css. Below the files list are 'New file' and 'VERSIONS 6' options. The main editor area displays HTML code for an AngularJS application. The code includes a head section with links to style.css and prism.css, and script tags for AngularJS, AngularJS animation, AngularJS sanitization, AngularJS UI Bootstrap, Prism.js, and loanpro_connect.js. The body section features an AngularJS application with a controller named 'requestController' and a view containing an 'h1' element for 'LoanPro API', an 'auth-form' element, and a 'send-form' element. Comments in the code provide instructions on how to adjust the API request in request_module.js and how to change credentials or the UI.

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <link rel="stylesheet" href="style.css" />
6   <link rel="stylesheet" href="prism.css" />
7   <script src="//ajax.googleapis.com/ajax/libs/angularjs/1.6.1/angular.js"></script>
8   <script src="//ajax.googleapis.com/ajax/libs/angularjs/1.6.1/angular-animate.js"></script>
9   <script src="//ajax.googleapis.com/ajax/libs/angularjs/1.6.1/angular-sanitize.js"></script>
10  <script src="//angular-ui.github.io/bootstrap/ui-bootstrap-tpls-2.5.0.js"></script>
11  <script src="prism.js"></script>
12  <script src="loanpro_connect.js"></script>
13  <!-- Change the code in request_module.js to adjust the API request -->
14  <script src="request_module.js"></script>
15  <link href="//netdna.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" rel="stylesheet" />
16 </head>
17
18 <body ng-app="sinnangClients">
19   <div ng-controller="requestController">
20     <h1>LoanPro API</h1>
21     <!-- Either change your credentials in request_module.js or change them in the UI -->
22     <auth-form</auth-form>
23     <!-- This sets the form to use to show the request in the UI -->
24     <send-form</send-form>
```


INDEX.HTML

- The entry point of the web app
- Includes all requisite libraries (ie. Angular, Prizm, Bootstrap, etc)

LOANPRO_CONNECT.JS

- Backend for communicating to the API via Angular
- Defines directives used to simplify code (ie. "auth-form", "send-form", "response-form")
- Good to use in your app if you are determined to publish your API token and have it stolen 😬

PRIZM.CSS & PRIZM.JS

- Used for highlighting JSON response from the server
- Not needed in 99.999% of integrations

REQUEST_MODULE.JS

THIS IS THE MOST IMPORTANT FILE!

- Defines the request sent to the server
 - uri - which endpoint to call
 - method - what HTTP method to use
 - data - the JSON data to send for POST, PUT, DELETE
 - query - JSON object representing the URL query
 - title - irrelevant, used to make samples look pretty

REQUEST_MODULE.JS (CONTINUED)

- Also where the sample API key and token are stored

DO NOT STORE API CREDENTIALS HERE AND PUBLISH!

Ok to store on local, but **never** put on the web - including Plunker

REQUEST_MODULE.JS (CONTINUED)

```
1
2 app.controller('requestController', ['$scope', '$http', '$sce', '$controller', function($scope, $http, $sce, $controller) {
3   $controller('loanproClient', {$scope: $scope, $http: $http, $sce: $sce});
4
5   // Change the following to match your credentials
6   $scope.api_auth.tenant_id = 5200243;
7   $scope.api_auth.api_token = "cc0199330033f963007b07e75f1b7fb6b7025887";
8
9   $scope.queueAPICall({
10    uri: "odata.svc/Customers",
11    method: "PUT",
12    data: {
13      name: "Bob",
14      employer: {
15        name: "Simnang",
16        job: "Tech Support"
17      }
18    },
19    query: {$expand: "Employer,MailAddress,Phones"},
20    title: "Customers"
21  });
22
23  });
```

Sample Tokens

API Endpoint

HTTP Method

JSON data being sent

URL Query

Used to make sample pretty

RESPONSE_FORM.HTML

Displays the results from the server

- Raw tab - shows unformatted JSON response
- Formatted tab - shows formatted JSON response
- Info tab - gives error information

SEND_REQUEST_FORM.HTML

Displays the raw HTTP request that is sent

Has the "Send" button

STYLE.CSS

Holds CSS code to make the samples look good

PURPOSE OF SAMPLES

- Samples serve the purpose of demonstrating how payloads are formed
- Allow you to experiment with the API in your browser
- Samples are not ment to be used in real-world applications
- Samples are not secure

REST OF PRESENTATIONS

- The rest of the presentations use the API samples
- Each presentation starts with a link to the associated sample
- Samples aren't required for presentations, but do help with understanding

QUESTIONS?

Feel free to ask in our [Forums](#)

Or, email us at support@simnang.com