



The Loan Payment Schedule Generator for Salesforce®

Web Services Developer Guide

Updated for AMCalc ver. 3.4



Copyright 3 Creeks Technologies LLC © 2020 - 2024

Table of Contents

Introduction	3
AMCalc Web Services Requirements	3
Using the AMCalc Web Services	4
AMCalc Web Services Response Headers	5
Web Services Query Parameter Reference	6
Request Response Formats	6
Request a list of supported loan types	7
Describe the calculator inputs for a given loan type	8
Request a list of current rates entries	12
Generate a loan payment schedule and return the results	14
Handling Schedule Generation Error Responses	21



Introduction

The AMCalc installation package includes a REST web service API that allows AMCalc subscribers to develop interfaces to the same loan schedule generation engine that AMCalc calculators use within Salesforce. The benefit of this web services is that it allows AMCalc subscribers to standardize their loan schedule generation across their organization. This includes using AMCalc for systems and websites outside of Salesforce.

REST (Representational State Transfer) web services have been a standard and widely used protocol across the Internet for years. It is a request and response service where the requestor, the system external to Salesforce, sends a request for information to a specific URL, and the responding system, AMCalc within Salesforce, sends the requested information back in a response.

Using the AMCalc web service, subscribers can do the following:

1. Get a list of the loan types that the AMCalc REST services supports
2. Get a description of the calculator inputs that are required or optional in order to generate a loan schedule for a given loan type
3. Get a list of the current interest and index rate entries in AMCalc
4. Generate an loan schedule for a given set of calculator inputs
5. Save an loan schedule to a parent Salesforce object

AMCalc Web Services Requirements

Other than your subscription, there are no additional AMCalc licensing requirements or fees imposed by 3 Creeks Technologies that are required for you to use the AMCalc web services. However, there may be additional licensing or fees required by Salesforce in order for your company to make calls to any Salesforce APIs. Refer to your Salesforce licensing agreement or contact your Salesforce rep to understand if additional licensing or fees are required.

Additionally, the following should be considered before you undertake a project involving the web service.

1. Making calls to the AMCalc web service counts against Salesforce API call limits

Salesforce places limits on the number of API calls that can be made to your Salesforce org. These limits are cumulative across all API calls that are made to your Salesforce org within a 24 hour period. This means that API calls made to AMCalc and to the standard Salesforce web services API, or any other Salesforce app that you installed off of the AppExchange, within a 24 hour period are added together to determine if you have reached your API limit. To understand more about API call limits in Salesforce please refer to the Salesforce documentation.

2. Using the AMCalc web service requires development knowledge

Using any web service requires development skills. Salesforce system administrators that are experts at configuring Salesforce through administration screens will not have the skills required



to implement a web service interface unless they are also familiar with programming languages, internet protocols, and database management.

3. *You will also need to be familiar with Salesforce development tools and practices*

Successfully using the AMCalc web services will also require you to be familiar with Salesforce's different technologies. These include such things as being able to authenticating an API user to Salesforce using one of their authentication methods such as OAuth, and being familiar with Salesforce database structures.

Using the AMCalc Web Services

To receive and use any response from the AMCalc Web Service, your requesting system must do the following in the order specified:

1. Authenticate the Salesforce user that will be making the request.

The authenticated user must have access privileges that allow them to make API calls and database privileges to query, create, and update data in the AMCalc database tables. See the AMCalc Administration guide for the permission sets that come with the AMCalc package. Your AMCalc web service user will need to have these permission sets assigned to their profile. Depending on other web services call that you will be making, there may be addition user privileges that your user will need. See the Salesforce documentation for more information.

2. Make a call to the AMCalc web services URL with your request.

The AMCalc REST web service is accessible through the following relative URL:

```
/services/apexrest/amortcalc/schedule_req
```

For example, to make an AMCalc web service request, append the above relative URL to the Salesforce instance part of your URL:

```
https://<Salesforce instance>.salesforce.com/services/apexrest/amortcalc/schedule_req
```

For Salesforce instances using My Domain, the full URL will have the form:

```
https://<My Domain name>.my.salesforce.com /services/apexrest/amortcalc/schedule_req
```

The request (list loan types, generate schedule, etc) that you are making to the AMCalc web service would need to be added to the end of the above URL by using specific query parameters. See the Web Services Query Parameter References section below for detailed information.

3. Decode and parse the request response



All AMCalc web services responses come in the form of URL encoded JSON strings. The URL encoding scheme used for all AMCalc responses is UTF-8. Use whatever URL decoding methods available in the programming language that you are using to decode the response. *You must first URL decode the response before parsing the JSON string.* See the Web Services Query Parameter References section below for details about the structure of the JSON string for each type of request.

AMCalc Web Services Response Headers

All AMCalc responses will have a header that provides information about the response. Use the response header to help understand the response that you received and to authenticate that the response is coming from the expected source.

Below is a sample of a response header from a request to get a list of loan types.

```
HTTP/1.1 200 OK
Date: Wed, 30 Sep 2020 17:25:50 GMT
Strict-Transport-Security: max-age=31536002; includeSubDomains
Public-Key-Pins-Report-Only: pin-sha256="9n0izTnSRF+W4W4JTq51avSXkWhQB8duS2bxVLfzXsY="; pin-sha256="5kJvNEMw0KjrCAu7eXY5HZdvyCS13BbA0VJG1RSP91w="; pin-sha256="njN4rRG+22dNXAi+yb8e3UMypgzPUPHlv4+foULwl1g="; max-age=86400; includeSubDomains; report-uri="https://a.forcesslreports.com/hpkp-report/00D5B0000003K3Dm";
Expect-CT: max-age=86400, report-uri="https://a.forcesslreports.com/Expect-CT-report/00D5B0000003K3Dm"
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
X-Robots-Tag: none
X-B3-TraceId: 1f28a201f00601b7
X-B3-SpanId: 1f28a201f00601b7
X-B3-Sampled: 0
Cache-Control: no-cache,must-revalidate,max-age=0,no-store,private
Set-Cookie: BrowserId=_Bt_lwNBEeuzMpFjuboxug; domain=.salesforce.com; path=/; expires=Thu, 30-Sep-2021 17:25:50 GMT; Max-Age=31536000
package-namespace: amortcalc
params-processed: listtypes
response-encoding: UTF-8
Content-Type: application/x-www-form-urlencoded
Transfer-Encoding: chunked
```



Much of this header is standard response header information but the AMCalc web service also inserts additional information into the header that can be used by your code.

Header Key	Description	Purpose
package-namespace	This is the unique name space that the AMCalc code runs in within Salesforce.	All AMCalc responses will have a package-namespace equal to <i>amortcalc</i> . Use this value to help ensure that the response you received came from AMCalc.
params-processed	The query parameters that the response is a result of. In the above example, the response body of this request will contain a list of the AMCalc loan types.	Use the value of this header key to further ensure that the response received is what was expected based on the request.
response-encoding	The URL encoding scheme that is used to encode the response body.	This key value can be used to set the correct encoding method in your code.

Web Services Query Parameter Reference

Request Response Formats

All AMCalc web service responses will come in the following form:

```
{"result": [success or error], "response": [AMCalc information or error message]}
```

Response formats of successful requests will vary depending on the request. See below for specifics.

If there is an error response, be aware that the error message may not always be understandable to an end user. For this reason, you may want to translate error messages in your code to something generic for your users.

Developer's Tip:

As you read through the different request types, you may notice that one request builds on another. For example, loan type names returned by the *listtypes* request are used in the request to describe calculator inputs. Then, the calculator input keys returned by *descinputs*, are re-used to specify the calculator input values in a request to generate a schedule.



The following web services requests are supported by AMCalc.

Request a list of supported loan types

Query Parameter: listtypes

Send only the query parameter key. There is no key value required for this request.

Request format:

```
.../services/apexrest/amortcalc/schedule_req?listtypes
```

Response Description:

A successful request (result = success), will contain an encoded JSON array with elements containing the loan types that AMCalc can generate a loan payment schedule for.

Example Response (URL decoded and formatted):

```
{
  "result": "success",
  "response": [
    "Fixed Payment",
    "Straight Line",
    "Fixed Principal Reduction",
    "Interest Only"
  ]
}
```



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

Describe the calculator inputs for a given loan type

Query Parameter: descinputs=[loan type]

Accepted query parameter values are any one of the loan types returned by the *listtypes* request. Only one loan type can be sent per request

Request format:

.../services/apexrest/amortcalc/schedule_req?descinputs=Fixed Payment

Response Description:

A successful request (result = success), contains a complete description of all inputs that are required or optional in order to generate a loan schedule for the loan type specified.

The input descriptions returned include the data type, possible values for data types that are lists, any required formatting, and whether the input is required for the loan type requested.

The table below provides a short description of all of the possible keys in the “response” portion of the returned JSON string. Note that depending on the loan type, some of the keys below may or may not be received in the returned JSON string. This is because AMCalc will only send the keys that are relevant to the loan type requested.

For more complete descriptions of AMCalc calculator inputs and example loan payment schedules for each type of loan, see the AMCalc User Guide.

Key	Description
loan_amount	The total dollar amount of the loan
start_date	The start date of first period of the loan
first_payment	The end date of the first period of the loan. Note: the <i>first_payment</i> key is optional. You can either send a specific first payment date, or allow the AMCalc engine calculate the <i>first_payment</i> date based on the other loan parameters that are sent. If you would like AMCalc to calculate the <i>first_payment</i> value, either send an empty <i>first_payment</i> value (ie. “”) or do not include the <i>first_payment</i> key in the request.
last_payment	The last payment day on the loan. Note: the <i>last_payment</i> key is optional. You can either send a specific last payment date, or allow the AMCalc engine



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

	<i>calculate the last _payment date based on the other loan parameters that are sent.</i> <i>If you would like AMCalc to calculate the last _payment value, either send an empty last _payment value (ie. "") or do not include the last _payment key in the request.</i>
frequency	Frequency of payments
time_unit	The time unit to use for the values that are shown for the term_periods and amort_periods keys
term_periods	The length of the loan. Must be less than or equal to amort_periods
amort_periods	The number of periods that the loan should be amortized over.
interest_rate	The interest rate to use in the schedule
month_convention	The day convention to use for the month portion of a period's interest rate calculation
year_convention	The day convention to use for the year portion of a period's interest rate calculation
eom_payment	Whether to force payment days to the last day of a month
payment_day	Force the payment day to a particular day of the month
interest_only_periods	For Interest Only loan types, the number of payment periods where only interest payments are made, ie. no portion of the payment is applied to principal.
interest_only_included	For Interest Only loan types, indicates whether the interest only periods are included within the loan of the loan or outside of it.
payment_amount	For Fixed Principal Reduction loan types, the payment amount to apply to the principal each payment period.

Example Response (URL decoded and formatted):

```
{
  "result": "success",
  "response": {
    "loan_amount": {
      "type": "decimal",
      "required": true,
      "format": "scale 2"
    },
    "start_date": {
      "type": "date",
      "required": true,
      "format": "yyyy-MM-dd"
    },
    "first_payment": {
```



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

```

    "type":"date",
    "required":true,
    "format":"yyyy-MM-dd"
  },
  "last_payment":{
    "type":"date",
    "required":true,
    "format":"yyyy-MM-dd"
  },
  "frequency":{
    "type":"list",
    "required":true,
    "values":[
      "Monthly",
      "Quarterly",
      "Semiannual",
      "Annual"
    ]
  },
  "time_unit":{
    "type":"list",
    "required":false,
    "values":[
      "year",
      "month"
    ]
  },
  "term_periods":{
    "type":"decimal",
    "required":true,
    "format":"scale 0"
  },
  "interest_rate":{
    "type":"decimal",
    "required":true,
    "format":"scale 6"
  },
  "month_convention":{
    "type":"list",
    "required":true,
    "values":[
      "30",
      "Actual"
    ]
  },
  "year_convention":{
    "type":"list",

```



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

```
"required":true,
"values":[
  "360",
  "365",
  "Actual"
]
},
"payment_day":{
  "type":"list",
  "required":false,
  "values":[
    "1",
    "2",
    "3",
    "4",
    "5",
    "6",
    "7",
    "8",
    "9",
    "10",
    "11",
    "12",
    "13",
    "14",
    "15",
    "16",
    "17",
    "18",
    "19",
    "20",
    "21",
    "23",
    "24",
    "25",
    "26",
    "27",
    "28",
    "29",
    "30",
    "31"
  ]
},
"eom_payment":{
  "type":"boolean",
  "required":false
},
```



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

```

    "amort_periods":{
      "type":"decimal",
      "required":true,
      "format":"scale 0"
    }
  }
}

```

Request a list of current rates entries

Query Parameter: listrates

Send only the query parameter key. There is no key value required for this request.

Request format:

.../services/apexrest/amortcalc/schedule_req?listrates

Response Description:

A successful request (result = success), returns a JSON array of arrays listing the current and active rate entries in the system. Each inner array is a three element array where the first element is the rate percentage, the second element is the rate entry name, and the third is the type of interest rate entry (Interest rate or Index).

When returning array of arrays, AMCalc always places the column heading labels in the first inner array that the outer array contains. See the example below.

Example Response (URL decoded and formatted):

```

{
  "result":"success",
  "response":[
    [
      "interest_rate",
      "entry_name",
      "type"
    ],
    [
      "2.670000",
      "Commercial 10 year rate",
      "Interest Rate"
    ],
    [
      "2.602300",

```



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

```
"Commercial 15 year rate"  
],  
[  
  "2.780000",  
  "Retail 10 year rate",  
  "Interest Rate"  
],  
[  
  "1.3000",  
  "Secured Overnight Financing Rate",  
  "Index"  
]  
]  
}
```



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

Generate a loan payment schedule and return the results

Query Parameter: `getschedule=[JSON formatted schedule input]`

A properly formatted query parameter key value for this request is a JSON formatted string containing the loan type of the loan schedule along with its calculator inputs pointing to their values. Use the calculator input labels (the JSON keys) returned by the *descinputs* request to ensure that you are referencing each calculator input correctly. See the example below for more information.

Developer's Tip

There are multiple variations of this call depending on what type of schedule information you would like returned and whether you want to save the schedule to a parent object record or not. All variations of this call start with the *getschedule* query parameter and a JSON string that specifies the loan type and the calculator inputs.

You will note that the below JSON strings are missing one additional closing bracket and, instead, end with ellipses. This is to illustrate that this is the first part of the JSON string for the *getschedule* call. The remaining sections describe the variations of the *getschedule* request.

Request format for loan type and calculator inputs only:

```
.../services/apexrest/amortcalc/schedule_req?getschedule= {  
    [loan type]:{  
        "loan_amount":[dollar amount],  
        "start_date":[yyyy-MM-dd],  
        "first_payment":[yyyy-MM-dd],  
        "last_payment":[yyyy-MM-dd],  
        "frequency":[payment frequency label],  
        "term_periods":[number periods],  
        "interest_rate":[interest rate],  
        "month_convention":[month convention label],  
        "year_convention":[year convention label],  
        "amort_periods":[number periods],  
        "time_unit":[time unit label]  
    } ...
```

Request Example:

```
.../services/apexrest/amortcalc/schedule_req?getschedule= {"Fixed Payment":  
    {"loan_amount":10000.00,  
    "start_date":"2019-12-12",  
    "first_payment":"2020-01-12",  
    "last_payment":"2029-12-12",  
    "frequency":"Monthly",
```



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

```
"term_periods":10,  
"interest_rate":2.780000,  
"month_convention":"30",  
"year_convention":"360",  
"amort_periods":10,  
"time_unit":"year"} ...
```

The following three sections specify the valid add-on key value pairs that can be appended to the above loan type and calculator inputs to make a complete *getschedule* request

Add-On #1

Generate a schedule, return only the calculated fixed payment, do not save the schedule

```
... "schedule_request":"payment"}
```

Request Example:

```
.../services/apexrest/amortcalc/schedule_req?getschedule= {"Fixed Payment":  
  {"loan_amount":10000.00,  
   "start_date":"2019-12-12",  
   "first_payment":"2020-01-12",  
   "last_payment":"2029-12-12",  
   "frequency":"Monthly",  
   "term_periods":10,  
   "interest_rate":2.780000,  
   "month_convention":"30",  
   "year_convention":"360",  
   "amort_periods":10,  
   "time_unit":"year"},  
  "schedule_request":"payment"}
```

Important: Although you can send a request with `schedule_request = payment` for any loan type, only Fixed Payment loan types have the same payments for the life of the loan. If you send a request with `schedule_request = payment` for a loan type that does not have fixed payments, the entire loan schedule will be sent back in the response.

You can easily determine in your code if you received a full schedule or not by examining the header returned in the first inner array and examining the number of outer array elements returned. Responses with only fixed payments will always have three outer array elements; one for the header and two for the payment amounts, one showing the fixed payment and one showing the last payment (last payments are typically different amounts due to rounding).

Response Description:



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

For a payment only request for a Fixed Payment loan type, a successful request (result = success), returns a JSON array of arrays listing the fixed payment amount and the loan's last payment amount. The last payment is included because usually the last payment varies from the fixed payment by anywhere from a few cents to a dollar or two due to rounding. You can also generate loan schedules with balloon payments for the last payment using Fixed Payment loan types. See the AMCalc User guide for details on Fixed Payment loans types.

When returning array of arrays, AMCalc always places the column heading labels in the first inner array that the outer array contains. See the example below.

Example Response (URL decoded and formatted):

```
{
  "result":"success",
  "response":[
    [
      "payment_description",
      "payment_amount"
    ],
    [
      "fixed_payment",
      1033.29
    ],
    [
      "last_payment",
      1033.02
    ]
  ]
}
```

Add-On #2

Generate a schedule, return the full schedule, do not save the schedule

```
... "schedule_request":"full"}
```

Request Example:

```
.../services/apexrest/amortcalc/schedule_req?getschedule= {"Fixed Payment":
{"loan_amount":10000.00,
"start_date":"2019-12-12",
"first_payment":"2020-01-12",
"last_payment":"2029-12-12",
"frequency":"Monthly",
"term_periods":10,
"interest_rate":2.780000,
"month_convention":"30",
"year_convention":"360",
```



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

```
"amort_periods":10,  
"time_unit":"year"},  
"schedule_request":"full"}
```

Response Description:

A successful request (result = success), returns a JSON array of arrays listing the generated schedule. Each element of the inner array represents a line in the generated schedule with element number 1 being the first period of the loan and the last element being the last loan payment.

When returning array of arrays, AMCalc always places the column heading labels in the first inner array that the outer array contains. See the example below.

Example Response (URL decoded and formatted):

Note that for readability, all schedule rows between 2 and 119 have been removed.

```
{  
  "result":"success",  
  "response":[  
    [  
      "period",  
      "period_start",  
      "payment_date",  
      "loan_balance",  
      "principal_paid",  
      "interest_paid",  
      "payment_amount"  
    ],  
    [  
      1,  
      "12/12/2019",  
      "1/12/2020",  
      100000.00,  
      661.38,  
      375.00,  
      1036.38  
    ],  
    [  
      2,  
      "1/12/2020",  
      "2/12/2020",  
      99338.62,  
      663.86,  
      372.52,  
      1036.38  
    ],  
    ...  
  ]  
}
```



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

```

119,
"10/12/2029",
"11/12/2029",
2061.76,
1028.65,
7.73,
1036.38
],
[
120,
"11/12/2029",
"12/12/2029",
1033.11,
1033.11,
3.87,
1036.98
]
]
}

```

Add-On #3

Generate a schedule, return only the calculated fixed payment, save the schedule and associate it with a parent Salesforce sObject

```
... "schedule_insert":{"parent_id":[parent id],"name": [schedule name],"status": [schedule status],"convert_schedule_to":[account, contact, or opportunity] }
```

Key/value pair descriptions:

- **schedule_insert** – Indicates that you are requesting that the generated loan schedule should be saved to a parent record
- **parent_id** – Key that points to the Salesforce ID of the parent record to associate the loan schedule. *Note: There must already be an existing parent-child relationship between the sObject and AMCalc.*
- **name** – This key points to the name to save the loan schedule with.
- **status** – Points to the Status value to assign to the saved loan schedule.
- **convert_schedule_to** – This key indicates the name of the new parent Salesforce sObject to move the loan schedule to if, in the future, the lead record that the loan record is associated to is converted to a Contact. The value of this key can only be one of the following Salesforce sObjects:
 - account
 - contact
 - opportunity

*Note: **convert_schedule_to** is an optional query parameter key and is only relevant if*



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

*the **parent_id** query parameter references a Lead ID. To be successfully moved on the lead conversion, the Salesforce object specified as the value of this key must already have a parent relationship with AMCalc loan schedules.*

Request Example:

```
.../services/apexrest/amortcalc/schedule_req?getschedule= {"Fixed Payment":  
    {"loan_amount":10000.00,  
    "start_date":"2019-12-12",  
    "first_payment":"2020-01-12",  
    "last_payment":"2029-12-12",  
    "frequency":"Monthly",  
    "term_periods":10,  
    "interest_rate":2.780000,  
    "month_convention":"30",  
    "year_convention":"360",  
    "amort_periods":10,  
    "time_unit":"year"},  
    "schedule_request":"payment",  
    "schedule_insert":  
    {"parent_id":"a0654000004KHwSAAW",  
    "name":"REST WS Generated",  
    "status":"Draft"}}
```

Response Description:

You are allowed to add the *schedule_insert* portion of the JSON string to either a *schedule_insert = payment* or *schedule_insert = full request*. In either case, the complete and full loan schedule is always saved when creating the AMCalc record. This is the case even when only the fixed payment is returned by the web service.

A successful request (result = success), returns a JSON array of arrays listing the requested loan schedule information followed by JSON key value pairs indicating if the schedule insert was successful and, if so, the newly created loan schedule's record Id.

Note: A getschedule request with a schedule_insert request typically returns two responses. The first response indicates if the schedule was generated successfully with the inputs that were sent. The second response indicates if the loan schedule insert request was successful or not. If there is an error on the schedule_insert request, the "id" key will not be returned and, instead, the key "error_message" will be sent with information specific to the insert request error. See below for an example of a response with two success responses and a second example with a successful schedule generation but an insert request error.

When returning array of arrays, AMCalc always places the column heading labels in the first inner array that the outer array contains. See the example below.



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

Example 1: Successful schedule generation and successful schedule insert response (URL decoded and formatted):

```
{
  "result": "success",
  "response": [
    [
      "payment_description",
      "payment_amount"
    ],
    [
      "fixed_payment",
      1036.38
    ],
    [
      "last_payment",
      1036.98
    ]
  ],
  "schedule_insert": "success",
  "id": "a005B000005O7zPQAS"
}
```

Example 2: Successful schedule generation but errored schedule insert response (URL decoded and formatted):

```
{
  "result": "success",
  "response": [
    [
      "payment_description",
      "payment_amount"
    ],
    [
      "fixed_payment",
      1036.38
    ],
    [
      "last_payment",
      1036.98
    ]
  ],
  "schedule_insert": "error",
  "error_message": "The value for the parent_id key, 0035B00000jgPJwQAM, is for a sObject that does not have a lookup relationship with the amortcalc__LoanSchedule__c sObject."
}
```



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024

```
}
```

Handling Schedule Generation Error Responses

Before a loan schedule is generated, the AMCalc engine validates all of the inputs that it receives. In order to reduce the need for multiple web service calls, as much validation is done at one time as is possible. Be aware that because of this, error responses can sometimes contain multiple messages.

AMCalc web service error responses will have a HTTP response code of 400, and take the following form:

```
{ "result": "error",  
  "response": [ [ "error_number", "error_message",  
                  [ [error #], [error message],  
                    [ [error #], [error message],  
                    ...  
                  ] ]  
}
```

Here is an example of an AMCalc error response with two errors that has been URL decoded and formatted:

```
{  
  "result": "error",  
  "response": [  
    [  
      "error_number",  
      "error_message"  
    ],  
    [  
      1,  
      "The loan amortization must be greater than or equal to the loan term"  
    ],  
    [  
      2,  
      "The start date must occur before the first payment date"  
    ]  
  ]  
}
```



3 Creeks Technologies

3 Creeks Technologies LLC

Copyright 3 Creeks Technologies LLC © 2020 - 2024



Copyright 3 Creeks Technologies LLC © 2020 - 2024