

**PRODUTOS E
SERVIÇOS DE
TECNOLOGIA E
MARKET DATA**

**CALCULADORA
DE RENDA FIXA
PRODUTOS DE
FRONT E MIDDLE
- BALCÃO**

ÍNDICE

1	INTRODUÇÃO.....	3
2	CERTIFICADO DE SEGURANÇA SSL.....	3
3	AUTENTICAÇÃO	4
4	WEB METHOD: LOGIN.....	5
5	WEB METHOD: LISTBONDCODES	6
6	WEB METHOD: LISTBONDCODESCSV.....	6
7	WEB METHOD: LISTBONDCODESCRA	6
8	WEB METHOD: LISTTITULOSPUBLICOS	7
9	WEB METHOD: LISTTITULOSPUBLICOSCSV	8
10	WEB METHOD: GETBONDDetails.....	8
11	WEB METHOD: CALCPU	8
12	WEB METHOD: CALCPUCSV.....	10
13	WEB METHOD: CALCYIELD	11
14	WEB METHOD: CALCYIELDCSV	13
15	TIPOS ESPECÍFICOS DE DADOS DA API	13
16	PARÂMETRO OPCIONAL DE IDENTIFICAÇÃO DE CHAMADAS	16
17	EXTRATO DE CONSUMO	16
18	CÁLCULO DE DI.....	18

1 INTRODUÇÃO

Este documento descreve os métodos disponíveis no Web Service para acessar e utilizar a CALC, a calculadora de renda fixa da B3.

O site de **Teste** da calculadora é: <https://apihom.calculadorarendafixa.com.br/>

O site de **Produção** do Web Service é: <https://api.calculadorarendafixa.com.br/>

A comunicação consiste no envio (request) e recebimento (response) de dados no formato JSON (JavaScript Object Notation), que segue o padrão de comunicação RESTful Web Services. Esse padrão permite a utilização de tipos específicos de recursos que não precisam ser definidos. Porém, para manter a compatibilidade com o padrão SOAP Web Service, serão utilizados os tipos definidos a seguir.

2 CERTIFICADO DE SEGURANÇA SSL

O endereço de acesso tem como prefixo “https”. Isso significa que a conexão com a API do Web Service será feita através do protocolo SSL criptografado.

Para uma conexão correta deve-se utilizar um componente apropriado para esse protocolo.

Em alguns casos, é necessário buscar um arquivo CA (PEM) em uma lista de certificados autorizados. Ao fazer isso, o componente será informado se a certificadora é válida. Um exemplo de arquivo que pode ser utilizado está disponível neste link: <http://curl.haxx.se/docs/caextract.html>.

3 AUTENTICAÇÃO

Para acessar os métodos do Webservice com segurança, é necessário realizar uma autenticação por meio do Web Method “login”.

Para efetuar o login é necessário informar:

2 - Código token: 64 caracteres alfanuméricos.

Esses dados serão fixos no caso de sistemas e clientes da B3.

Caso ocorra algum erro na autenticação (Status: 403 – Forbidden), a resposta indicará o motivo da falha.

Caso a autenticação seja bem-sucedida (Status: 200 – Ok), a resposta do Web Method “login” informará o código de autorização ou o código da sessão de utilização do Webservice, denominado “authorization”. Esse código de autorização possui de 32 a 96 caracteres alfanuméricos e deve ser enviado em todas as demais chamadas feitas ao Web Service, utilizando um cabeçalho HTTP “Authorization”, conforme o exemplo:

```
$ curl -H "Authorization:  
5db1bfe6ef7d54b31c42ef5a493590878f276f785e2d442297514f" -X GET  
<metodo>
```

Se código de autorização não for enviado ou estiver inválido, a resposta (Status: 401 – Unauthorized) informará o motivo pelo qual o código não foi aceito.

Um caso comum é a expiração do tempo da sessão, o que ocorre quando o Webservice não é chamado durante 3 (três) horas. Nesse caso, é necessário realizar a autenticação novamente.

Para o site de **Produção**, o *token* pode ser obtido na aba “Meus dados”

4 WEB METHOD: LOGIN

O que faz: autentica o acesso ao Web Service.

HTTP Method: POST

Argumentos de Chamada: Via método POST (no corpo da requisição HTTP), enviar os dados de login (*key* e *token*).

Os dados podem ser enviados nos seguintes formatos:

Urlencoded:

Necessário cabeçalho HTTP: Content-Type: application/x-www-form-urlencoded

Ex.: token=meuToken

Json:

Necessário cabeçalho HTTP: Content-Type: application/json

Ex.: {"token": "MeuToken"}

Valor de resposta é um objeto JSON, por exemplo:

```
{"login":"meuLogin","Authorization":"depadmbaf3addd89012e524c38d8adfd2119185c7b9600584c6c165d3c24c0e9cbeb4b87d250a6d5fa1d5dcde2894","perfil":"BASICO"}
```

É Necessário enviar o campo "Authorization" para a chamada de todos os outros métodos do WebService.

Forma de chamada: POST <webServiceUrl>/login

Exemplos utilizando o cUrl:

```
$ curl -H "Content-type: application/x-www-form-urlencoded" -X POST -o
```

```
token=MeuToken' <WebserviceUrl>/login
```

```
$ curl -H "Content-type: application/json" -X POST -d
```

`{'token':'MeuToken'} <WebserviceUrl>/login`

5 WEB METHOD: LISTBONDCODES

O que faz: lista todos os códigos de Debêntures disponíveis na calculadora.

HTTP Method: GET

Argumentos de Chamada: Nenhum

Valor de Resposta: (JSON – Array of Strings) – Array contendo os códigos das debêntures.

Forma de chamada: `<WebserviceUrl>/listBondCodes`

6 WEB METHOD: LISTBONDCODESCSV

O que faz: lista todos os códigos de Debêntures disponíveis na calculadora.

HTTP Method: GET

Argumentos de Chamada: Nenhum

Valor de Resposta: (CSV) – Códigos de debêntures separados por ‘;’ (ponto e vírgula).

Forma de chamada: `<WebserviceUrl>/listBondCodesCSV`

7 WEB METHOD: LISTBONDCODESCRA

O que faz: lista todos os códigos de CRA disponíveis na calculadora.

HTTP Method: GET

Argumentos de Chamada: Nenhum

Valor de Resposta: (JSON – Array of Strings) – Array contendo os códigos dos CRA's.

Forma de chamada: *<WebserviceUrl>/listBondCodesCRA*

8 WEB METHOD: LISTBONDDESCRI

O que faz: lista todos os códigos de CRI disponíveis na calculadora.

HTTP Method: GET

Argumentos de Chamada: Nenhum

Valor de Resposta: (JSON – Array of Strings) – Array contendo os códigos dos CRA's.

Forma de chamada: *<WebserviceUrl>/listBondCodesCRI*

9 WEB METHOD: LISTTITULOSPUBLICOS

O que faz: lista todos os Títulos Públicos disponíveis na calculadora.

HTTP Method: GET

Argumentos de Chamada: Nenhum

Valor de Resposta: (JSON – Array of TituloPublico) – Tipo específico descrito a seguir.

Forma de chamada: *<WebserviceUrl>/listTitulosPublicos*

10 WEB METHOD: LISTTITULOSPUBLICOSCSV

O que faz: lista todos os Títulos Públicos disponíveis na calculadora.

HTTP Method: GET

Argumentos de Chamada: Nenhum

Valor de Resposta: (CSV) – Lista de Títulos Públicos (um por linha) com campos separados por ';' (ponto e vírgula).

Forma de chamada: <WebserviceUrl>/listTitulosPublicosCSV

11 WEB METHOD: GETBONDDetails

O que faz: informa os dados das Debêntures, CRA e CRI, como data de vencimento, taxa base, etc.

HTTP Method: GET

Argumentos de Chamada: Código da Debênture/CRA (bondCode – string)

Valor de Resposta: (JSON – Bond) – Tipo específico descrito a seguir.

Forma de chamada: <WebserviceUrl>/getBondDetails/<bondCode>

Ex.: <WebserviceUrl>/getBondDetails/CSMG26

Ex.: <WebserviceUrl>/getBondDetails/CRA019000XD

Obs.: Método somente para Debêntures, CRA's e CRI's

12 WEB METHOD: CALCPU

O que faz: calcula o Preço Unitário (PU) da Debênture, do CRA, do CRI ou do Título Público com base na taxa informada.

HTTP Method: GET

Argumentos de Chamada:

CRA:

Código do CRA (bondCode –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Taxa de Retorno (yield – double)

CRI:

Código do CRI (bondCode –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Taxa de Retorno (yield – double)

Debêntures:

Código da Debênture (bondCode –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Taxa de Retorno (yield – double)

Títulos Públicos:

Código SELIC do Título (selicid –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Taxa de Retorno (yield – double)

Valor de Resposta: (JSON – CalculationResult) – Tipo específico descrito a seguir.

Forma de chamada:

`<WebserviceUrl>/calcPU/<bondCode>/<calculationDate>/<yeild>`

Ex.: `<WebserviceUrl>/calcPU/CSMG26/2014-05-07/6.22`

Ex.: `<WebserviceUrl>/calcPU/CRA019000XD/2020-07-07/6.22`

Obs.: O *Web Method* detecta automaticamente se o cálculo pedido é de uma Debênture, de um CRA ou de um Título Público.

13 WEB METHOD: CALCPUCSV

O que faz: calcula o Preço Unitário (PU) da Debênture, do CRA ou do Título Público com base na taxa informada.

HTTP Method: GET

Argumentos de Chamada:

CRA:

Código do CRA (bondCode –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Taxa de Retorno (yield – double)

CRI:

Código do CRI (bondCode –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Taxa de Retorno (yield – double)

Debêntures:

Código da Debênture (bondCode –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Taxa de Retorno (yield – double)

Títulos Públicos:

Código SELIC do Título (selicid –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Taxa de Retorno (yield – double)

Valor de Resposta: (CSV) – method, Yield, VNA, PUPar, PU e duration separados por ';' (ponto e vírgula).

Forma de chamada:

`<WebserviceUrl>/calcPUCSV/<bondCode>/<calculationDate>/<yeild>`

Ex.: `<WebserviceUrl>/calcPUCSV/CSMG26/2014-05-07/6.22`

Obs.: O *Web Method* detecta automaticamente se o cálculo pedido é de uma Debênture, de um CRA ou de um Título Público.

14 WEB METHOD: CALCYIELD

O que faz: Calcula a taxa de uma Debênture ou de um Título Público com base no PU informado.

HTTP Method: GET

Argumentos de Chamada

Debêntures:

Código da Debênture (bondCode –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Preço Unitário (PU – double)

CRA:

Código do CRA (bondCode –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Preço Unitário (PU – double)

CRI:

Código do CRI (bondCode –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Preço Unitário (PU – double)

Títulos Públicos:

Código SELIC do Título (selicid –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Preço Unitário (PU – double)

Valor de Resposta: (JSON - CalculationResult) – Tipo específico descrito a seguir.

Forma de chamada:

`<WebserviceUrl>/calcYield/<bondCode>/<calculationDate>/<PU>`

Ex.: [<WebserviceUrl>/calcYield/CSMG26/2014-05-07/961256.968405](#)

Obs.: O *Web Method* detecta automaticamente se o cálculo pedido é de uma Debênture ou de um Título Público.

15 WEB METHOD: CALCYIELDCSV

O que faz: Calcula a taxa de uma Debênture ou de um Título Público com base no PU informado.

HTTP Method: GET

Argumentos de Chamada:

Debêntures:

Código da Debênture (bondCode –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Preço Unitário (PU – double)

Títulos Públicos:

Código SELIC do Título (selicid –string)

Data de Cálculo (calculationDate –string no formato “YYYY-MM-DD”)

Preço Unitário (PU – double)

Valor de Resposta: (CSV) – method, Yield, VNA, PUPar, PU e duration separados por ';' (ponto e vírgula).

Forma de chamada:

<WebserviceUrl>/calcYieldCSV/<bondCode>/<calculationDate>/<PU>

Ex.: <WebserviceUrl>/calcYieldCSV/CSMG26/2014-05-07/961256.968405

Obs.: O *Web Method* detecta automaticamente se o cálculo pedido é de uma Debênture ou de um Título Público.

16 TIPOS ESPECÍFICOS DE DADOS DA API

Os tipos específicos de dados transmitidos pelo WebService são:

Bond:

codbond (string) – Código da Debênture e/ou do CRA

issuer (string) – Nome do Emissor do Papel

method (string) – Método de Cálculo ou calculadora específica para este papel

vne (double) – Valor Nominal de Emissão

yield (double) – Taxa Base (cupom) da Debênture e/ou do CRA

expiredate (string no formato “YYYY-MM-DD”) – Data de Vencimento

issuedate (string no formato “YYYY-MM-DD”) – Data de Emissão

startingdate (string no formato “YYYY-MM-DD”) – Data de Início de Cálculo

events (Array de Event) – Eventos do Papel

TituloPublico:

selicid (string) – Código SELIC do Título

issuer (string) – Fixo: “Banco Central do Brasil”

method (string) – Método de Cálculo ou calculadora específica para este papel
(LTN, LFT,

NTN-B, etc.)

expiredate (string no formato “YYYY-MM-DD”) – Data de Vencimento

CalculationResult:

codbond (string) – Código da Debênture e/ou do CRA

method (string) – Método de Cálculo ou calculadora específica para este papel

date (string no formato “YYYY-MM-DD”) – Data da Operação

VNA (double) – Valor Nominal Atualizado

interest (double) – Valor de Juros Valorizado até a data de cálculo

PUPar (double) – PUPar da Debênture e/ou do CRA para a data de cálculo

PU (double) – Preço Unitário da Operação

yield (double) – Taxa de Retorno da Operação

duration (double) – Duração do Papel para data e taxa especificadas

cashFlowList (Array de CashFlow) – Fluxos Futuros de Caixa da Operação

note (string) – Alguma informação adicional sobre a Debênture e/ou do CRA, como pagamento de prêmio

hasNote (boolean) – indica se há ou não alguma informação adicional na debênture ou no CRA.

CashFlow:

date (string no formato “YYYY-MM-DD”) – Data do Fluxo de Caixa

eventType (string) – Caractere que indica o tipo de evento que gerou o Fluxo de Caixa, sendo:

“J” – Pagamento de Juros, “A” – Amortização e “V” – Vencimento

workingDays (inteiro) – Quantidade de dias úteis entre a data do fluxo e a data de cálculo

yield (double) – Taxa do Fluxo (utilizada em casos de futuro de DI)

finalValue (double) – Valor Futuro do Fluxo de Caixa

presentValue (double) – Valor do Fluxo de Caixa trazido a valor presente

Event:

date (string no formato “YYYY-MM-DD”) – Data do Evento

eventType (string) – Caractere que indica o tipo de evento que gerou o Fluxo de Caixa, sendo:

“J” – Pagamento de Juros, “A” – Amortização e “V” – Vencimento

yield (double) – Taxa do Fluxo

17 PARÂMETRO OPCIONAL DE IDENTIFICAÇÃO DE CHAMADAS

Para que sua chamada de cálculo seja identificada é possível adicionar um parâmetro opcional “Client-ID” ao header de cada requisição.

Exemplo:

```
▼ Request Headers view source
Accept: application/json
Accept-Encoding: gzip, deflate, br
Accept-Language: pt-BR,pt;q=0.9,en-US;q=0.8,en;q=0.7
Authorization: batchb3803abdbb220d2a7c656e386062e428da8d31221335932dd28c95c56973a5d0c2cf3e6ea77095745867de18
Cache-Control: no-cache
Client-ID: identificacao-client-1234
```

18 EXTRATO DE CONSUMO

Chamada para obter o extrato de cálculos realizados pelo participante entre um período de data.

HTTP Method: GET

Ex.: <WebserviceUrl>/consumo/{dataInicial}/{dataFinal}

```
curl -X GET -H 'Authorization: conegtester79b32b914bdb19a86af73d2626d1f7f2f44fdfef77c50642e00dc868265803a01de304f7fe0a847688' -H 'Content-Type: application/json' -H 'Accept: application/json' -v -i 'https://api.hom.calculadorarendafixa.com.br/consumo/2020-07-01/2020-07-30'
```

Exemplo de resposta:

```
{
  "quantidadeCalculos": 21,
  "quantidadeCalculosDependentes": 59,
  "detalhe": [
    {
      "quantidadeCalculos": 12,
      "login": "assinante3",
      "nome": " ASSINANTE 3 ",
      "codAtivo": "DI"
    },
    {
      "quantidadeCalculos": 6,
      "login": "assinante3",
      "nome": " ASSINANTE 3 ",
      "codAtivo": "Debenture"
    },
    {
      "quantidadeCalculos": 3,
      "login": "assinante3",
      "nome": " ASSINANTE 3 ",
      "codAtivo": "Titulo Publico"
    }
  ],
  "detalheDependentes": [
    {
      "quantidadeCalculos": 25,
      "login": "assinante1",
      "nome": "ASSINANTE 1",
      "codAtivo": "DI"
    },
    {
      "quantidadeCalculos": 19,
      "login": "assinante1",
      "nome": "ASSINANTE 1",
      "codAtivo": "Debenture"
    },
    {
```

```
"quantidadeCalculos": 15,  
"login": "assinante1",  
"nome": "ASSINANTE 1",  
"codAtivo": "Titulo Publico"  
}  
]  
}
```

19 CÁLCULO DE DI

Exemplo: [<WebserviceUrl /di/calculo?dataInicio=2018-09-19&dataFim=2018-09-20&percentual=100&valor=1.2](https://api.calculadorarendafixa.com.br/di/calculo?dataInicio=2018-09-19&dataFim=2018-09-20&percentual=100&valor=1.2)

HTTP Method: GET

```
curl -X GET -v -i 'https://api.calculadorarendafixa.com.br/di/calculo?  
dataInicio=2018-09-19&dataFim=2018-09-20&percentual=100&valor=1.2'
```

Exemplo de resposta:

```
{  
  "dataInicial": "19/09/2018",  
  "dataFinal": "20/09/2018",  
  "percentual": "100.00",  
  "fator": "1.00024583",  
  "taxa": "0.02",  
  "valorBase": "1.20",  
  "valorCalculado": "1.20"  
}
```



[B3.COM.BR](https://www.b3.com.br)