

Login

Last updated by | Tiago Souza < tiago.souza@simply.com.br > | Mon, 19 Sep 2022 11:22:52 GMT

Efetuando Login

Para acesso ao Atomics, é necessário efetuar o login informando o usuário e senha.

POST no endpoint:

```
https://atomics-api.simply.com.br/Token
```

Para logar, efetue uma requisição do tipo `POST`. Nesta requisição, deve ser enviado o `Content-Type` com o valor `application/x-www-form-urlencoded` e o `Accept` com o valor `application/json` no header:

```
{
  "Content-Type": "application/x-www-form-urlencoded"
  "Accept": "application/json"
}
```

No `body` informe o usuário e a senha, e na chave `grant_type` informe o valor `password`:

```
{
  "username": "informe_seu_usuario",
  "password": "informe_sua_senha",
  "grant_type": "password"
}
```

Exemplo de retorno de sucesso

```
{
  "access_token": "token_de_acesso",
  "token_type": "bearer",
  "expires_in": 86399,
  "userName": "nome_usuario",
  ".issued": "Tue, 17 Apr 2018 18:48:10 GMT",
  ".expires": "Wed, 18 Apr 2018 18:48:10 GMT"
}
```

Quando o login for efetuado com sucesso, você receberá o `access_token` que será utilizado para todas as outras requisições no Atomics. Também será retornado a chave `.expires`, que contém a data de expiração do seu `access-token`, sendo necessário logar novamente.

Exemplo de retorno de erro

```
{
  "error": "invalid_grant",
  "error_description": "O usuario/senha informado não é válido."
}
```

Exemplo de Código (C#)

```
public static async Task LoginAtomics(string usuario, string senha)
{
    try
    {
        //recupera token de acesso
        using (var client = new HttpClient
            (new HttpClientHandler {AutomaticDecompression = DecompressionMethods.GZip |
                DecompressionMethods.Deflate}))
        {
            client.BaseAddress = new Uri (UrlBase);
            var data = new Dictionary ();

            //adiciona os campos de usuário
            data.Add("username", usuario);
            data.Add("password", senha);
            data.Add("grant_type", "password");

            FormUrlEncodedContent form = new FormUrlEncodedContent (data);
            client.DefaultRequestHeaders.Add("Accept", "application/json");
            HttpResponseMessage resposta = await client.PostAsync("token", form);

            resposta.EnsureSuccessStatusCode();
            string result2 = await resposta.Content.ReadAsStringAsync();

            JObject objeto = JObject.Parse(result2);
            BearerToken = objeto["access_token"].ToString();
        }
    }
    catch (Exception e)
    {
        //Sua excessão
    }
}
```

Exemplo de Código (PHP)

```

$curl = curl_init();

curl_setopt_array($curl, array(
    CURLOPT_URL => "URL_BASE",
    CURLOPT_RETURNTRANSFER => true,
    CURLOPT_ENCODING => "",
    CURLOPT_MAXREDIRS => 10,
    CURLOPT_TIMEOUT => 0,
    CURLOPT_FOLLOWLOCATION => true,
    CURLOPT_SSL_VERIFYPEER => false,
    CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
    CURLOPT_CUSTOMREQUEST => "POST",
    CURLOPT_POSTFIELDS => "username=XXXXXXXX&password=XXXXXXXX&grant_type=password",
    CURLOPT_HTTPHEADER => array(
        "Accept: application/json",
        "Content-Type: application/x-www-form-urlencoded"
    ),
));

$response = curl_exec($curl);
curl_close($curl);
$token_simple = json_decode($response);
$token_simple__ = "";

if (isset($token_simple->access_token)) {
    $token_simple__ = $token_simple->access_token;
}

else if (isset($token_simple["access_token"])) {
    $token_simple__ = $token_simple["access_token"];
}

```

Exemplo de Código (Node)

```

var request = require('request');
var options = {
  'method': 'POST',
  'url': 'URL_BASE',
  'headers': {
    'Accept': 'application/json',
    'Content-Type': 'application/x-www-form-urlencoded'
  },
  form: {
    'username': 'xxxxxxx',
    'password': 'xxxxxxx',
    'grant_type': 'password'
  }
};

request(options, function (error, response) {
  if (error) throw new Error(error);
  var tokenJSON = JSON.parse(response.body);
});

```