

# TUTORIAL

## openEHR com PostMan

### INTRODUÇÃO

Este tutorial elaborado pela [Code4Health](#) foi adaptado para demonstrar a utilização de APIs openEHR para iniciar uma sessão EHR, popular e recuperar openEHR Compositions no ambiente EhrScape.

O ambiente EhrScape foi construído sobre as especificações do [OpenEHR](#), fornecendo uma estrutura de serviços para permitir que criemos aplicativos de saúde inovadores através de um ambiente web amigável.

No [site](#) do EhrScape estão disponíveis documentos com detalhes sobre como criar aplicativos utilizando as APIs do EhrScape.

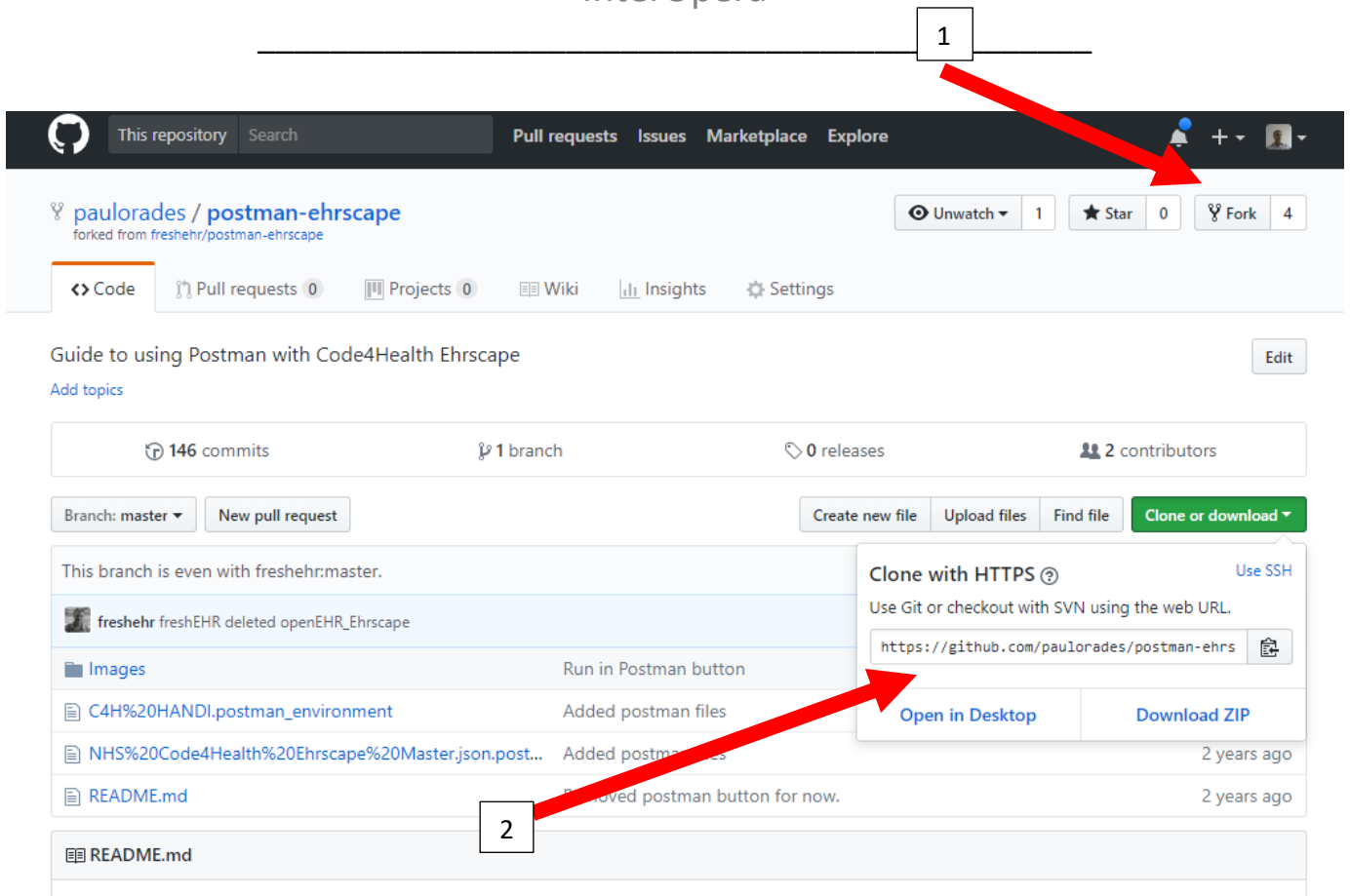
Manter-se atualizado sobre as especificações openEHR e a documentação do EhrScape é fundamental, estão sujeitas a alterações, sendo as APIs versionadas e podem ficar obsoletas.

### COLEÇÃO DE APIs

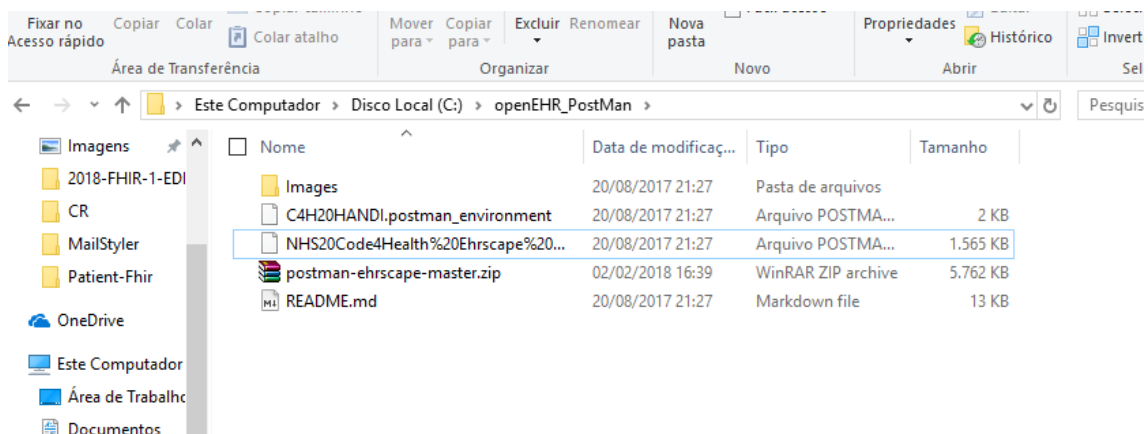
Um conjunto de arquivos contendo as APIs são necessários para a configuração de um ambiente de trabalho em seu cliente HTTP. Neste tutorial vamos utilizar o cliente HTTP [POSTMAN](#).

Este conjunto de arquivos está disponível junto do projeto original deste tutorial no [GITHUB](#). Você precisará usar a opção FORK para ter acesso ao conteúdo.

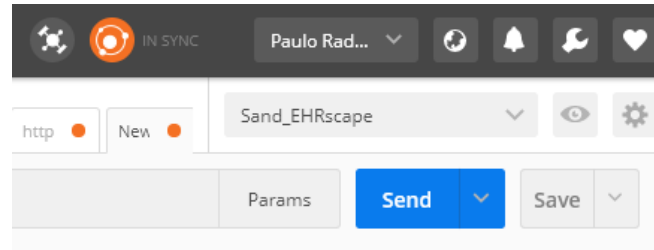
# InterOpera



1. FORK
2. Clone o repositório ou faça o download em um diretório.
3. Descompacte o arquivo
4. Certifique-se de que os arquivos `NHS_Code4Health_Ehrscape.json.postman` e `<your_environment_name.postman_environment` tenham sido corretamente extraídos.



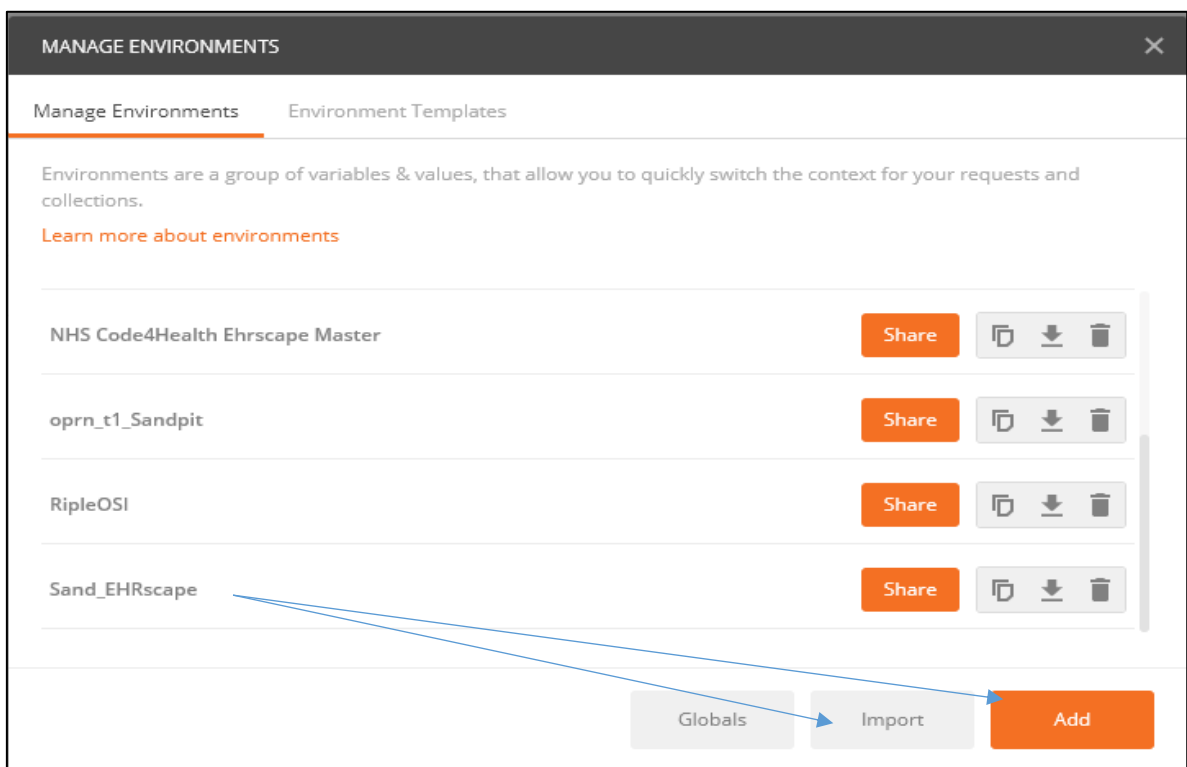
## CRIAR AMBIENTE E IMPORTAR OS ARQUIVOS



1 - Crie o ambiente de trabalho. Nós estamos usando **Sand\_EhrScape**.

Snap de Tela POSTMAN - Ambiente utilizado para a Collection do tutorial

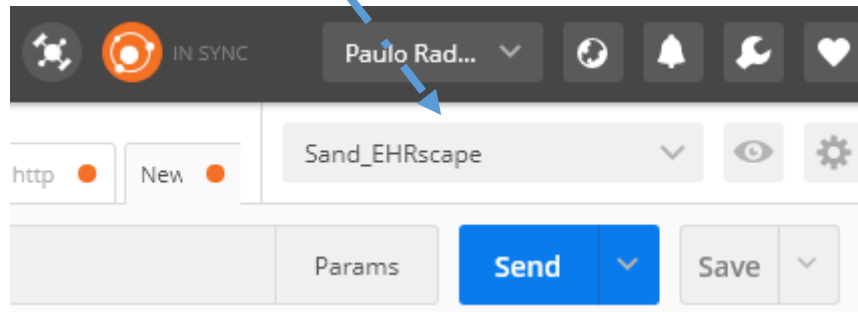
Abra o Postman e acesse o gerenciador de ambientes. Nele você pode criar um novo ambiente, clonar existentes e gerenciá-los. Após criado o ambiente realize a importação de ambos arquivos. Use a opção **Import** disponível.



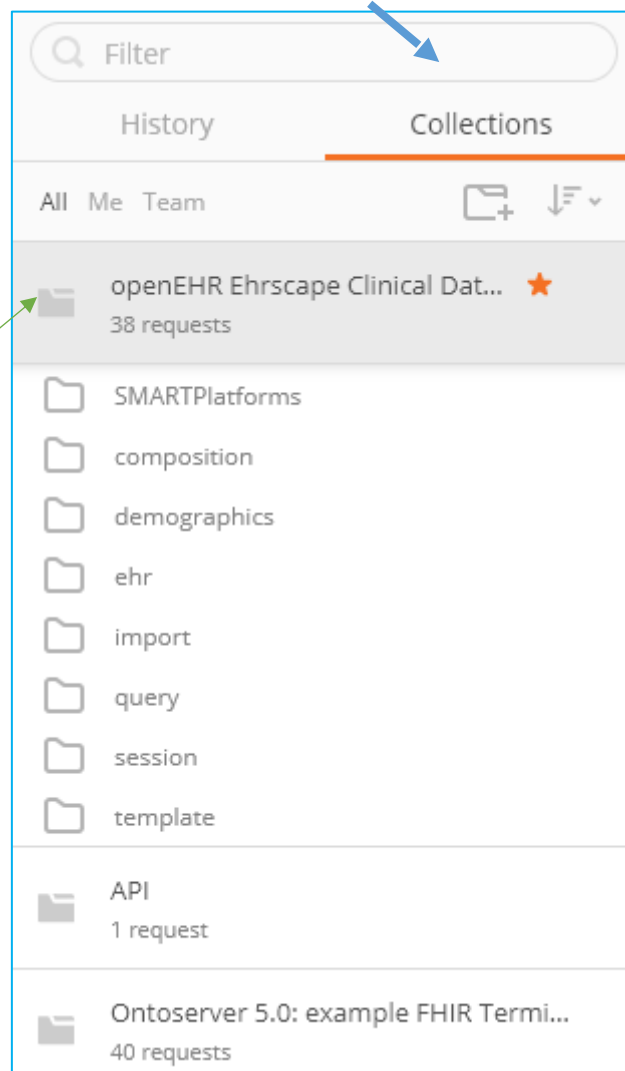
2

1

No canto superior direito da janela do POSTMAN, certifique-se de estar utilizando o ambiente criado.

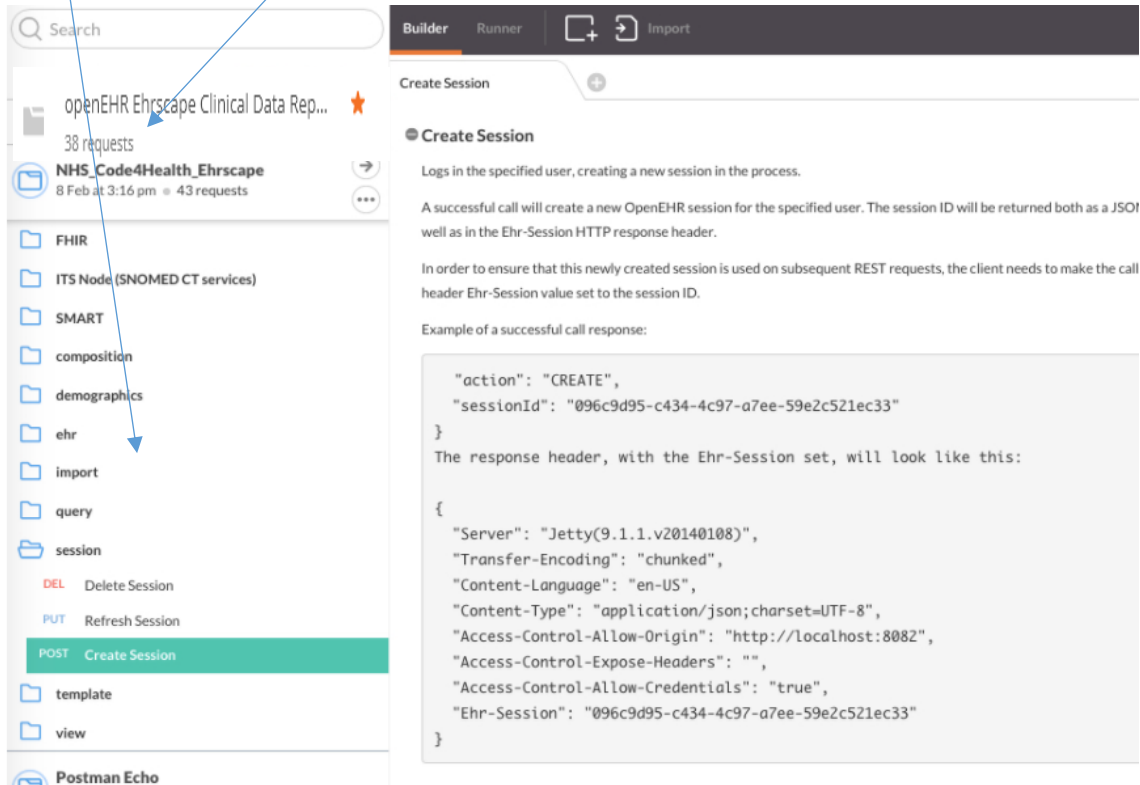


No frame esquerdo, temos acesso as Collections, como visto abaixo. Note que sua tela poderá ser diferente da apresentada. Esta é uma área onde temos acesso a todas as diferentes Collections disponíveis para uso e eu tenho outras configuradas.



## EXPLORANDO A COLLECTION

Expanda a collection **openEHR EHRscape Clinical Data Repository** e clique nas pastas para expandir seu conteúdo. Selecione uma das APIs e detalhes serão exibidos no lado direito.



Ao clicarmos no ícone **x** no canto superior direito acessaremos o conjunto de variáveis pré-definidas usadas no ambiente selecionado (neste tutorial Sand\_EHRscape). Estes conjuntos pré-definidos são usados pelas chamadas das APIs.

## TRABALHANDO COM O EhrScape

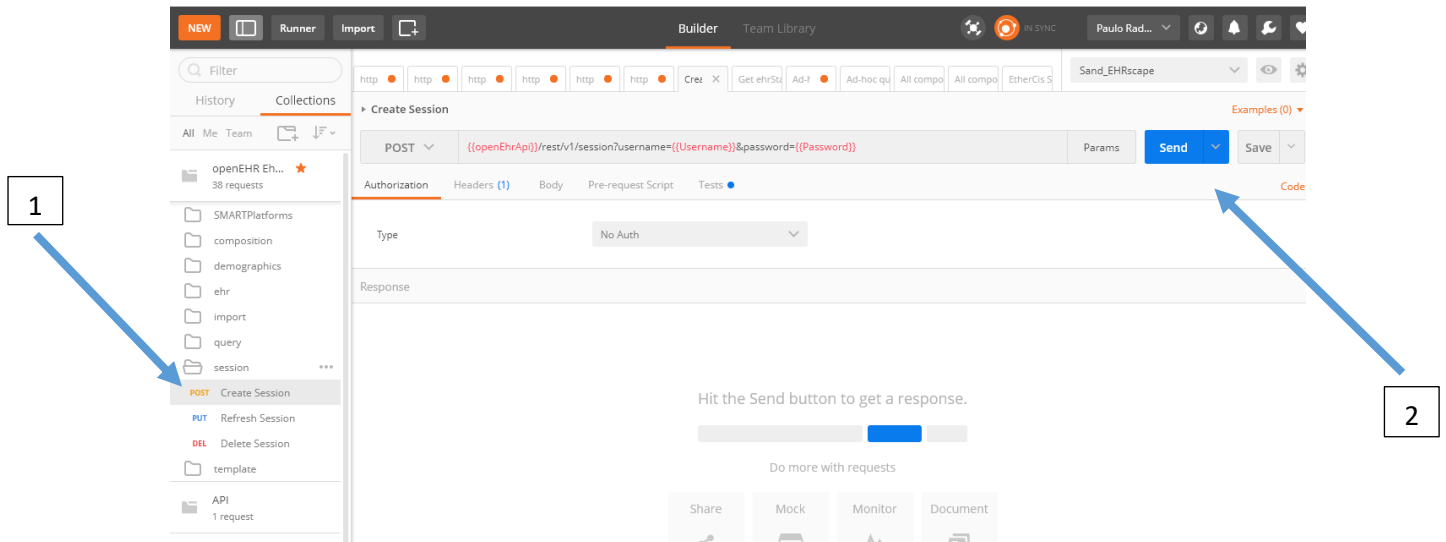
Com o ambiente criado, estamos prontos para iniciarmos com o Ehrscape.

- Criando uma Sessão

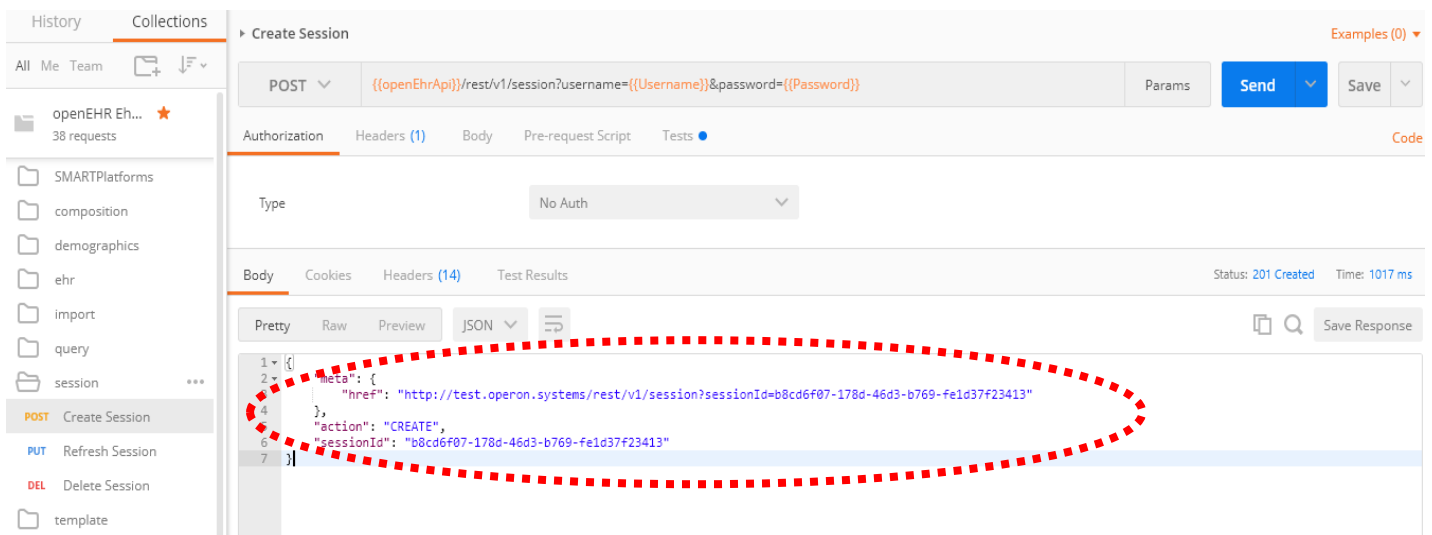
# InterOpera

O primeiro passo é criarmos uma sessão openEHR e obtermos um token do tipo `sessionId`. Isto nos permitirá que as chamadas API subsequentes sejam feitas sem a necessidade de fazermos login a cada interação.

Na pasta `session` selecione `Create Session` e clique no botão `Send`. As credenciais de acesso necessárias são geradas automaticamente no arquivo Pre-sets como explicado acima.



A captura de tela abaixo mostra o ID retornado para a sessão.

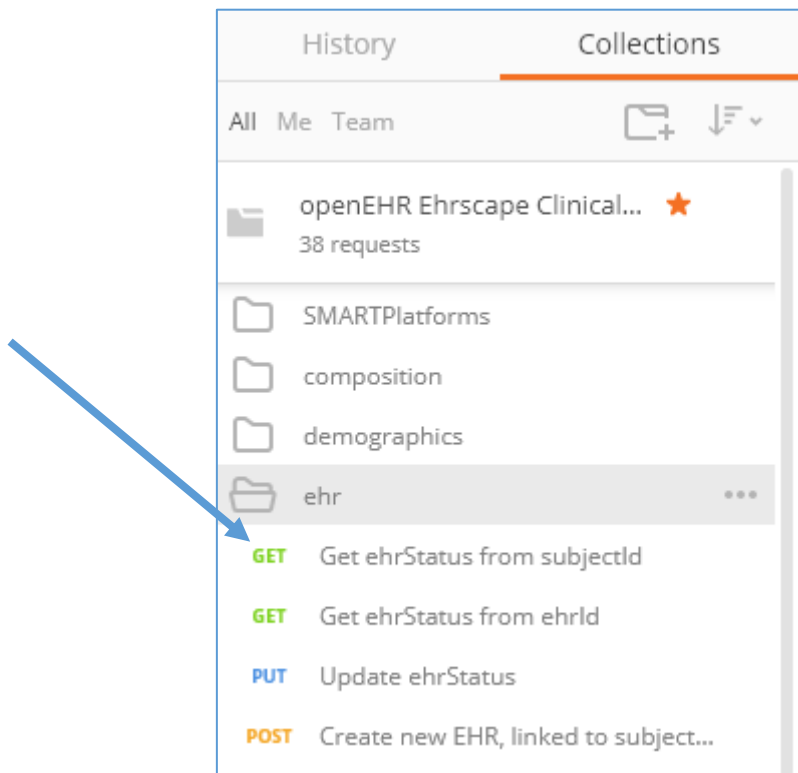


## IDENTIFICADOR EHR DO PACIENTE

O próximo passo é obter o identificador EHR interno do paciente através de um identificador externo, como por exemplo, o CNS – Cartão Nacional de Saúde. Neste tutorial vamos usar a identificação do [NHS - National Health Service](#).

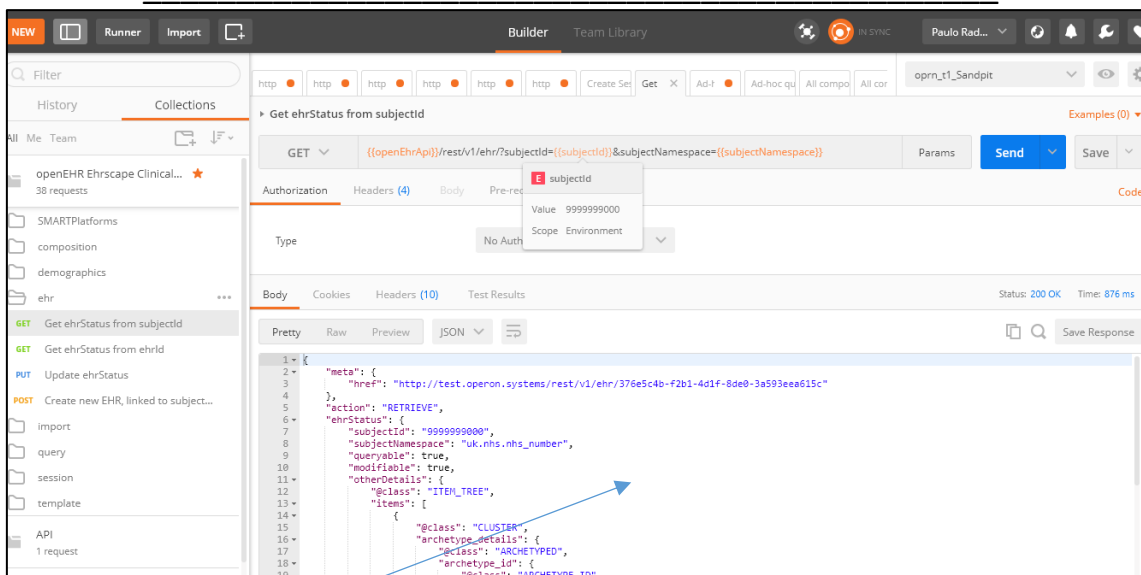
O **ehrId** é uma identificação única que, por motivos de segurança, não pode ser associada ao paciente, se, por exemplo, seus registros de saúde forem vazados.

Abra a pasta **ehr**, selecione a opção **Get ehrStatus from subjectId** e clique no botão **Send**. O número de identificação NHS do paciente é extraído do arquivo Pre-sets e, portanto, preenchido automaticamente.

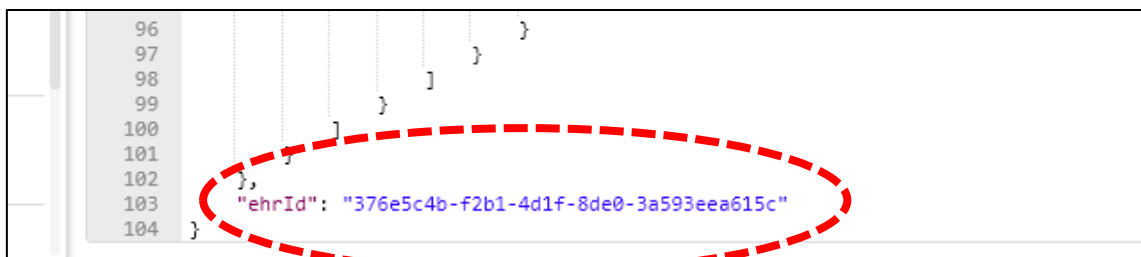


Clique no **Scroll to responses** para exibir os detalhes da resposta. O snippet JSON abaixo mostra o **ehrId** para nosso paciente de testes.

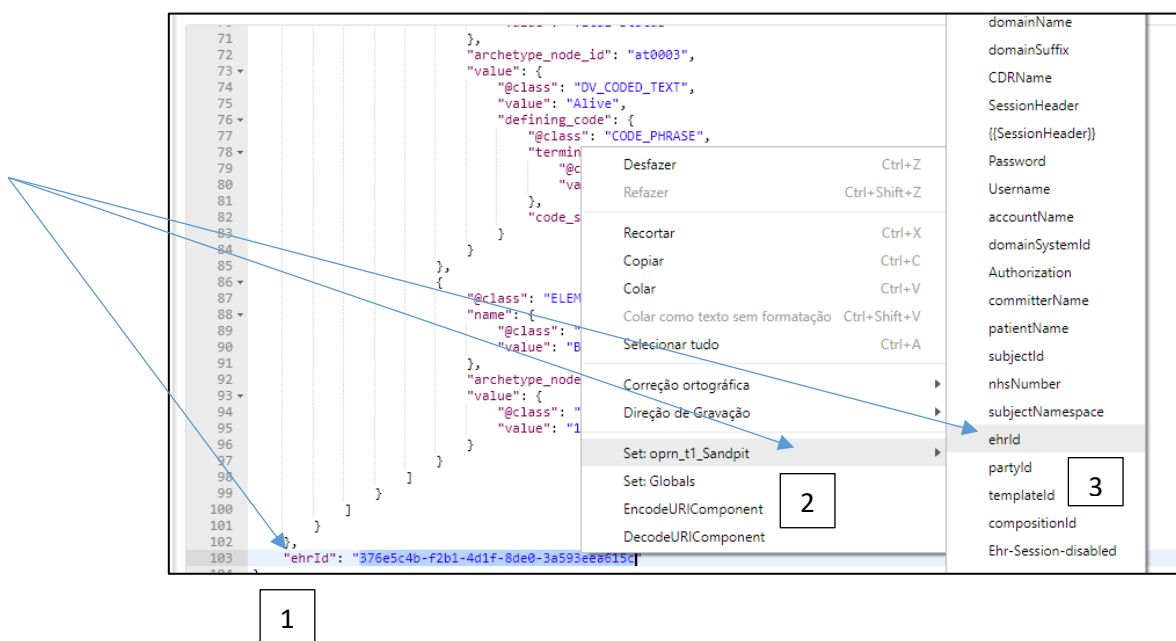
# InterOpera



Scrool Responses



Para armazenarmos o **ehrid** retornado como pré-definido no ambiente de trabalho, selecione e marque a string de caracteres retornado, clicando com o botão direito do mouse, e defina o ambiente (em nosso tutorial: Sand\_EHRscape) atribuindo este valor para o atributo **ehrid**.



## RECUPERAR O ID DE UMA COMPOSITION

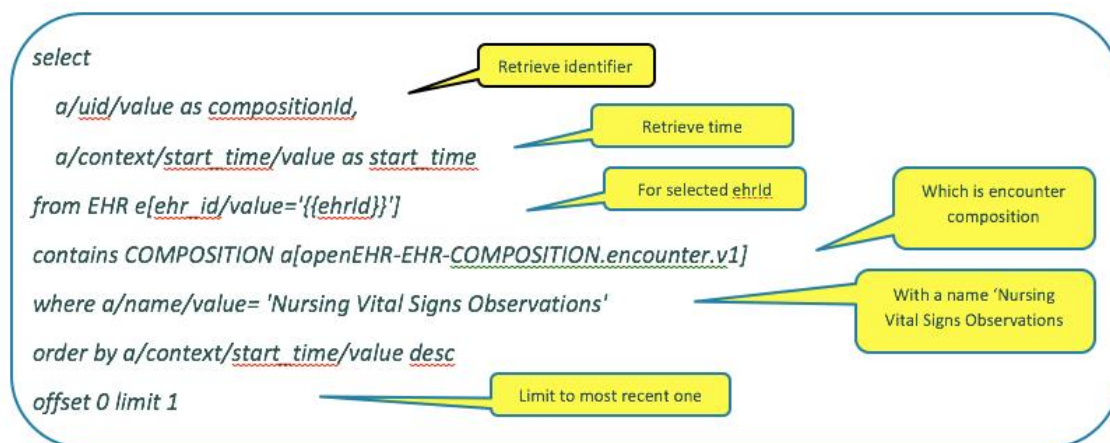
Agora que temos o identificador EHR do paciente, podemos usá-lo para localizar e recuperar informações clínicas.

O openEHR usa uma linguagem própria chamada de [AQL](#) (Arquetype Query Language) para recuperar a lista de identificadores e datas dos registros da Composition de Observações de Sinais Vitais.

As [Compositions](#) são registros a nível de documentos e que atuam como os contêineres para todos os dados dos pacientes, no openEHR.

O nome das Compositions seguem um padrão definido pela especificação do openEHR, sendo case-sensitive.

A interação que precisamos executar para obter o **id** da Composition de Sinais Vitais mais recente para o paciente selecionado é a seguinte:

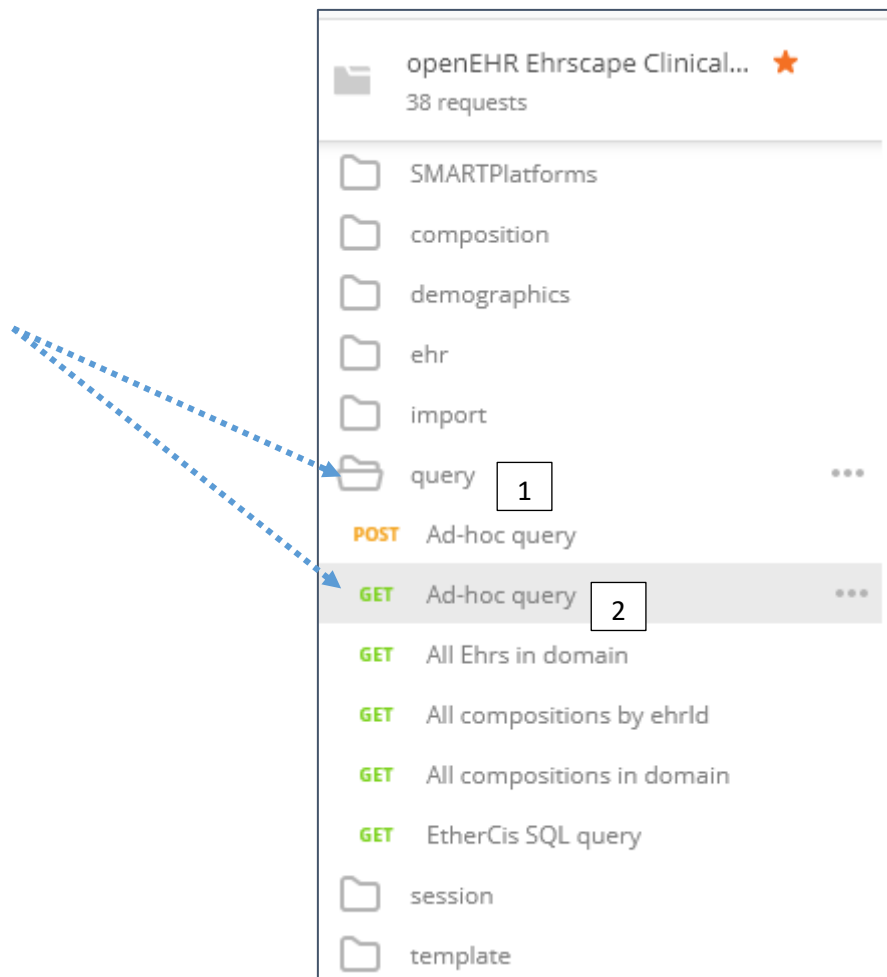


Fonte: <https://github.com/paulorades/postman-ehrscape>

Neste momento, não se preocupe com a sintaxe ou em como escrever consultas AQL, estes tópicos são abordados pelas especificações [openEHR](#) que fornecem todos os detalhes necessários.

# InterOpera

Abra a pasta **query** e selecione **Ad-hoc query**:

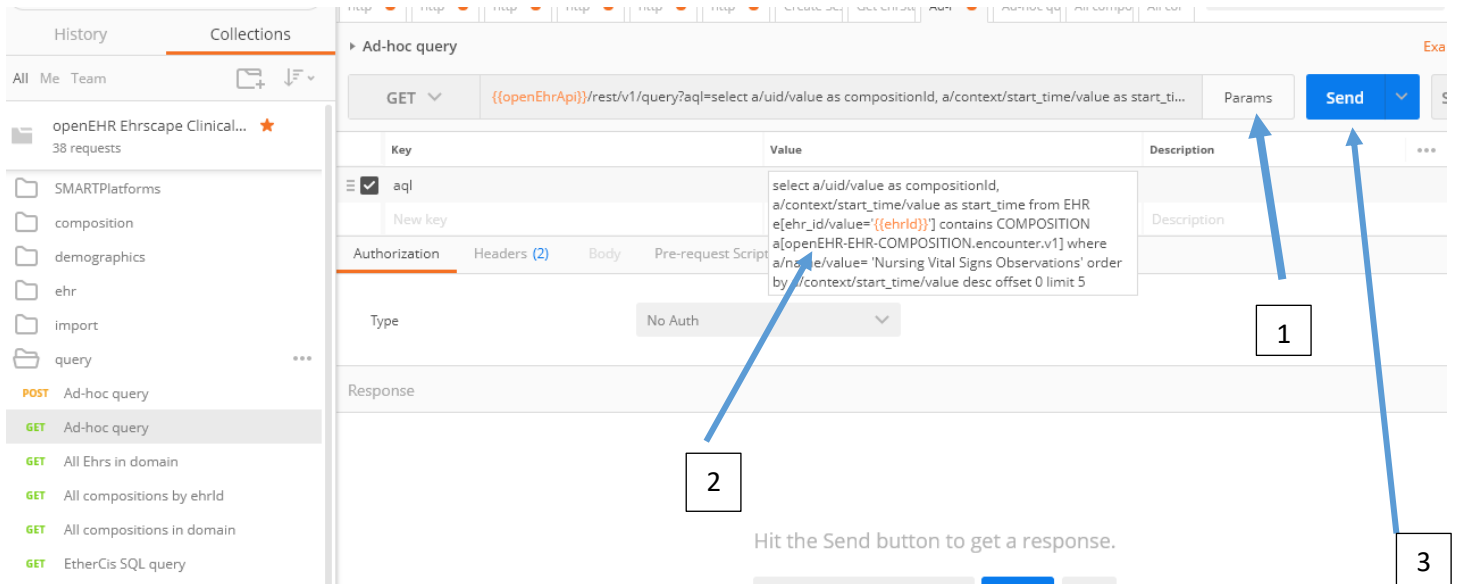


Esta é a sintaxe AQL para ser usada em nossa consulta. Vamos copiar e colar este código.

```
select
a/uid/value as compositionId,
a/context/start_time/value as start_time
from EHR e[ehr_id/value='{{ehrId}}']
contains COMPOSITION a[openEHR-EHR-COMPOSITION.encounter.v1]
where a/name/value= 'Nursing Vital Signs Observations'
order by a/context/start_time/value desc
offset 0 limit 1
```

## InterOpera

Na janela **Ad-hoc Query**, clique em **Param** e cole a sintaxe AQL copiada no campo **Value** e depois clique no botão **Send**.



Clique no **Scroll to response** para consultar os detalhes da resposta.

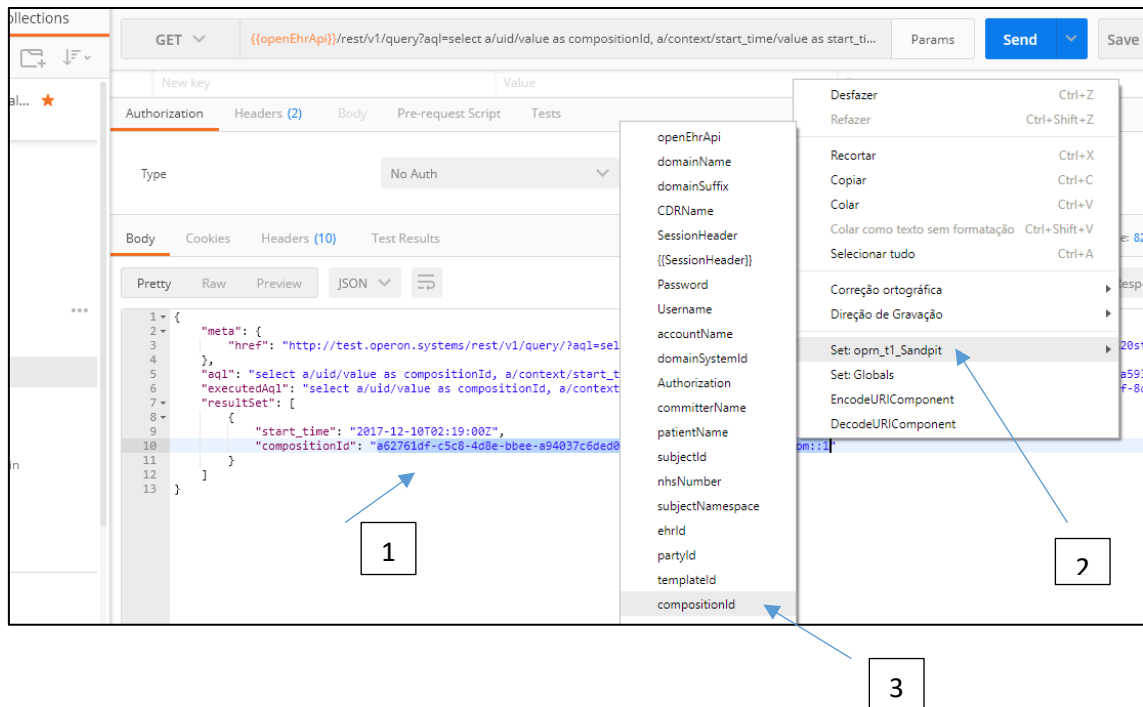


O elemento **compositionId** é o identificador único para a Composition e o elemento **start\_time** é o ponto-no-tempo em que o documento foi criado.

Vamos utilizar os resultados desta consulta para recuperar a composição completa, de modo que a ação final será armazenar o ID da composição como um valor pré-definido para uso posterior.

## InterOpera

Selecione a string e clique com o botão direito do mouse, definindo o ambiente (como feito anteriormente) e selecionando o atributo **compositionId** a partir desta lista.



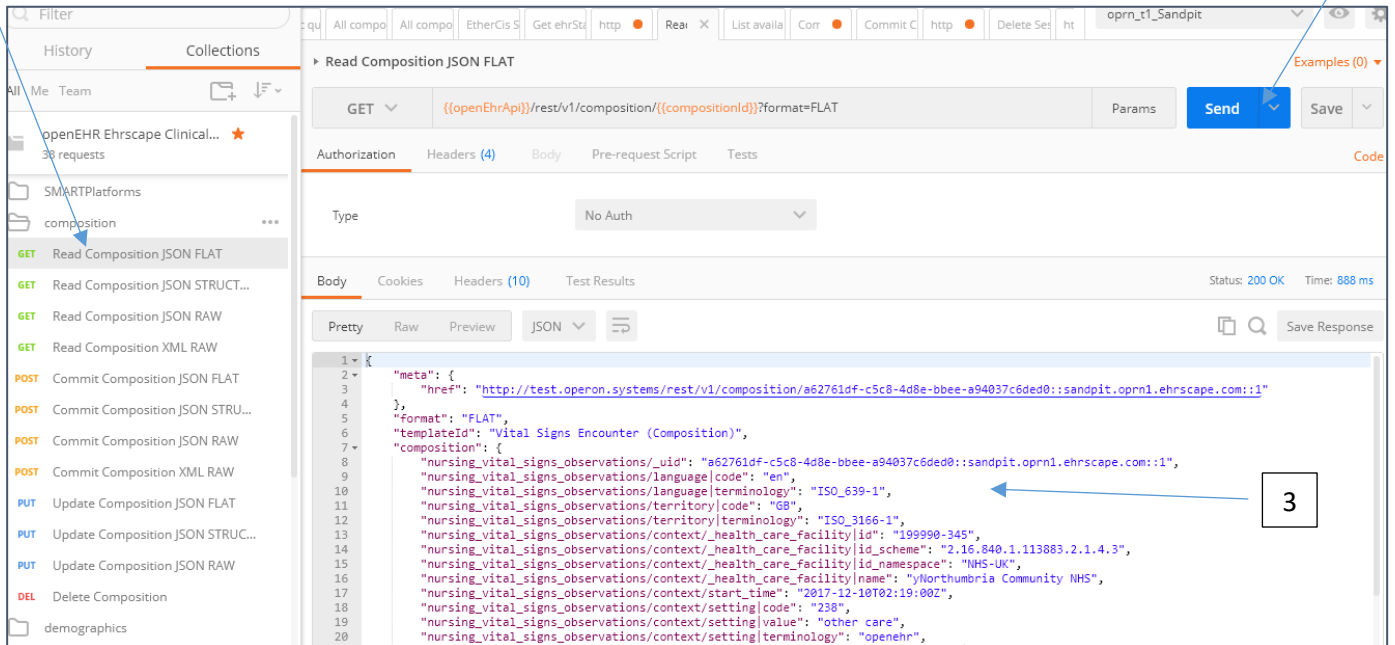
## COMPOSITIONS E COMPOSITIONID

O próximo passo é recuperarmos a Composition através de sua compositionID, recuperado e armazenado no passo anterior.

1

Navegue até a pasta **composition** e selecione **Read Composition JSON FLAT**, em seguida, clique no botão **Send**. O resultado será como o abaixo em FLAT JSON.

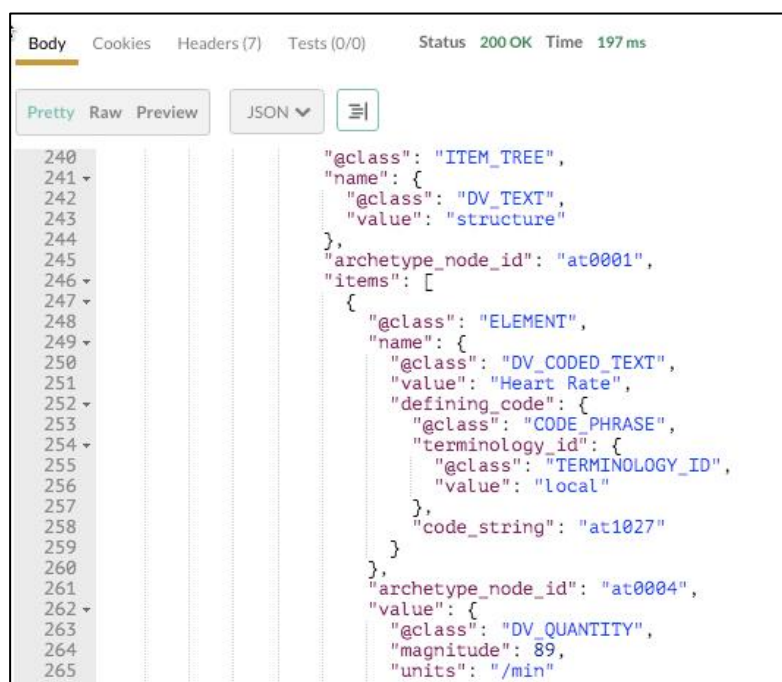
2



3

Outros formatos possíveis são JSON RAW, XML RAW ou JSON STRUCTURED.

Os snippets abaixo mostram trechos de códigos com os dados das medidas observadas para Pulso em alguns dos formatos possíveis.



# InterOpera

Body Cookies Headers (9) Tests (0/0) Status 200 OK Time 247 ms

Pretty Raw Preview XML

```
160 <value>2015-10-10T02:19:00.000Z</value>
161 </time>
162 <data xsi:type="ITEM_TREE" archetype_node_id="at0001">
163 <name>
164 <value>structure</value>
165 </name>
166 <items xsi:type="ELEMENT" archetype_node_id="at0004">
167 <name xsi:type="DV_CODED_TEXT">
168 <value>Heart Rate</value>
169 <defining_code>
170 <terminology_id>
171 <value>local</value>
172 </terminology_id>
173 <code_string>at1027</code_string>
174 </defining_code>
175 </name>
176 <value xsi:type="DV_QUANTITY">
177 <magnitude>89.0</magnitude>
178 <units>/min</units>
179 <precision>0</precision>
180 </value>
181 </items>
```

Body Cookies Headers (7) Tests (0/0) Status 200 OK Time 183 ms

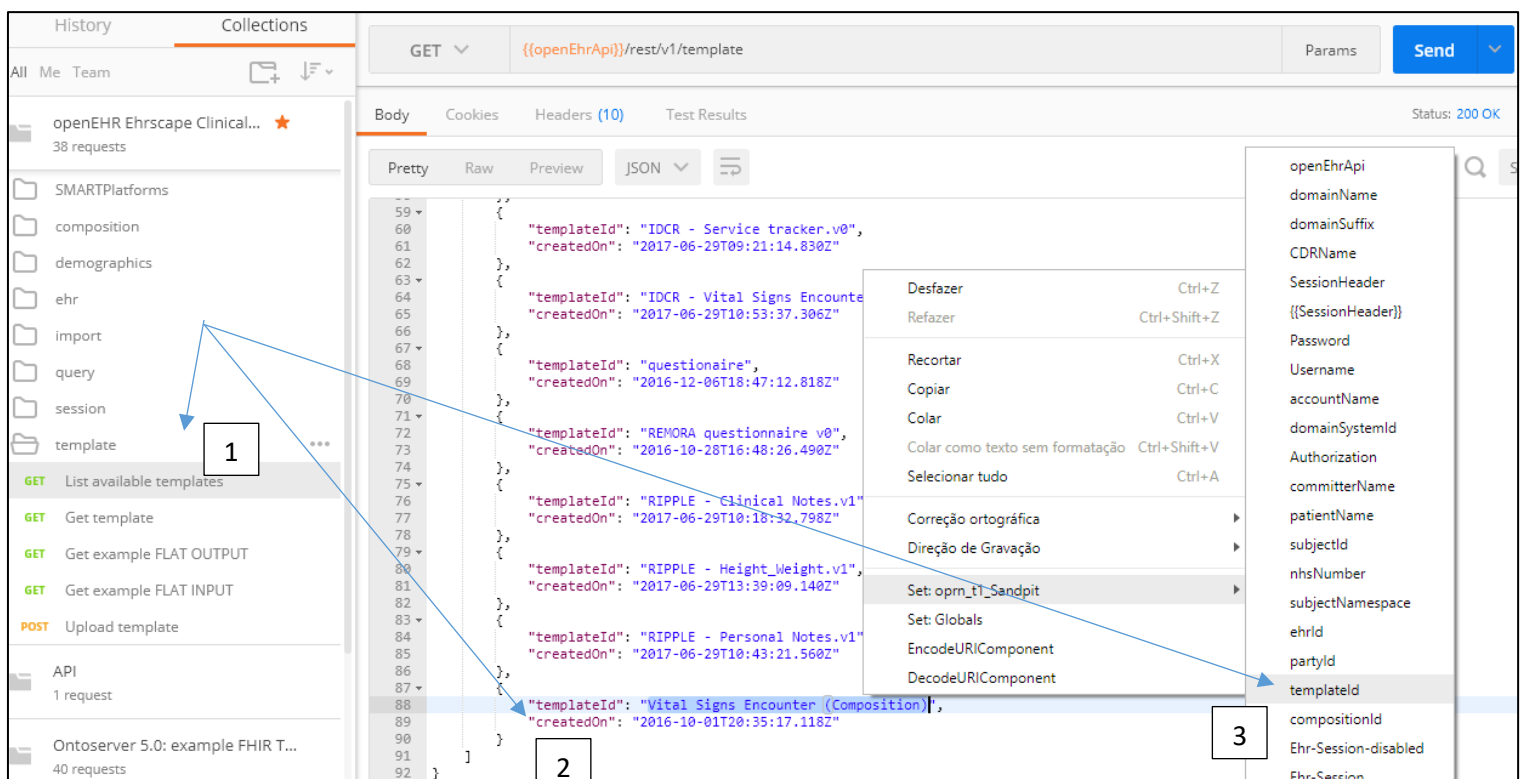
Pretty Raw Preview JSON

```
39 "2015-10-10T02:19:00.000Z"
40 ]
41 }
42 ]
43 }
44 ],
45 "pulse": [
46 {
47 "any_event": [
48 {
49 "heart_rate": [
50 {
51 "magnitude": 89,
52 "unit": "/min"
53 }
54 ],
55 "time": [
56 "2015-10-10T02:19:00.000Z"
57 ]
58 }
59 ]
60 }
```

## PERSISTINDO UMA COMPOSITION

O próximo passo é persistir uma nova composição. Os dados na composição são validados contra um modelo, sendo a primeira ação, definir o template para que a composição possa ser associada, validada e persistida.

Na pasta **Template**, selecione **List available templates**, em seguida, clique no botão **Send**. Selecione **Vital Signs Encounter template** a partir da lista de templates disponíveis, clique com o botão direito do mouse, defina o ambiente (como já feito anteriormente) e selecione **templateId** a partir da lista de atributos.



Uma vez que o templateId esteja configurado em nosso ambiente de trabalho, podemos fazer o commit (gravar) a Composition. Utilize o modelo abaixo como exemplo de uma Composition de sinais vitais:

```
{
  "ctx/language": "en",
  "ctx/territory": "GB",
  "ctx/composer_name": "Hazel Smith",
  "ctx/time": "2015-12-10T02:19:00.000Z",
  "ctx/health_care_facility|id": "999999-345",
  "ctx/health_care_facility|name": "Northumbria Community NHS",
  "ctx/id_namespace": "NHS-UK",
```

## InterOpera

```
"ctx/id_scheme": "2.16.840.1.113883.2.1.4.3",

"nursing_vital_signs_observations/vital_signs:0/respirations:0/any_event:0/rate|magnitude": 22,

"nursing_vital_signs_observations/vital_signs:0/respirations:0/any_event:0/rate|unit": "/min",

"nursing_vital_signs_observations/vital_signs:0/pulse:0/any_event:0/heart_rate|magnitude": 101,

"nursing_vital_signs_observations/vital_signs:0/pulse:0/any_event:0/heart_rate|unit": "/min",

"nursing_vital_signs_observations/vital_signs:0/body_temperature:0/any_event:0/temperature|magnitude": 36.6,

"nursing_vital_signs_observations/vital_signs:0/body_temperature:0/any_event:0/temperature|unit": "°C",

"nursing_vital_signs_observations/vital_signs:0/blood_pressure:0/any_event:0/systolic|magnitude": 100,

"nursing_vital_signs_observations/vital_signs:0/blood_pressure:0/any_event:0/systolic|unit": "mm[Hg]",

"nursing_vital_signs_observations/vital_signs:0/blood_pressure:0/any_event:0/diastolic|magnitude": 60,

"nursing_vital_signs_observations/vital_signs:0/blood_pressure:0/any_event:0/diastolic|unit": "mm[Hg]",

"nursing_vital_signs_observations/vital_signs:0/indirect_oximetry:0/any_event:0/spo2|numerator": 94,

"nursing_vital_signs_observations/vital_signs:0/indirect_oximetry:0/any_event:0/spo2|denominator": 100,

"nursing_vital_signs_observations/vital_signs:0/national_early_warning_score_rcp_uk:0/total_score": 3
}
```

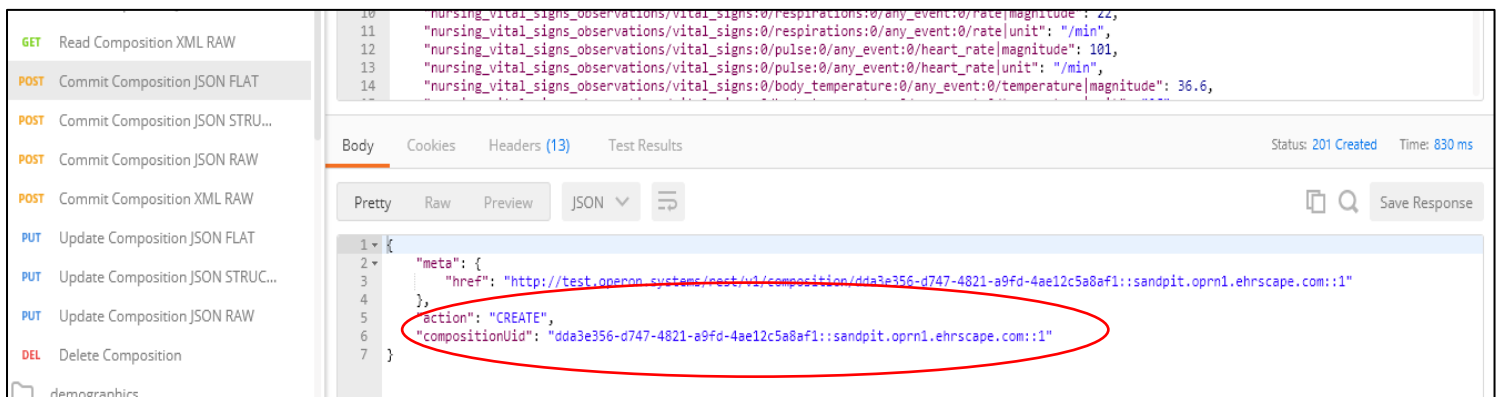
Como dito anteriormente, não se preocupe em como escrever a sintaxe para criar uma Composition. Estas definições são cobertas pela especificação do openEHR. Selecione a sintaxe acima exatamente como fornecida e copie para a área de transferência.

## Interopera

Na pasta **Composition** selecione **Commit Composition JSON FLAT** e cole o texto copiado. Para isto, clique em **Body** na parte superior da janela e clique no botão **Send**.



E como saída temos o compositionID gerado para esta Composition.



---

## CONSULTAS COM AQL

O próximo passo é executar uma consulta para as Compositions de observações de sinais vitais mais recentes do paciente e retornar este conjunto de dados.

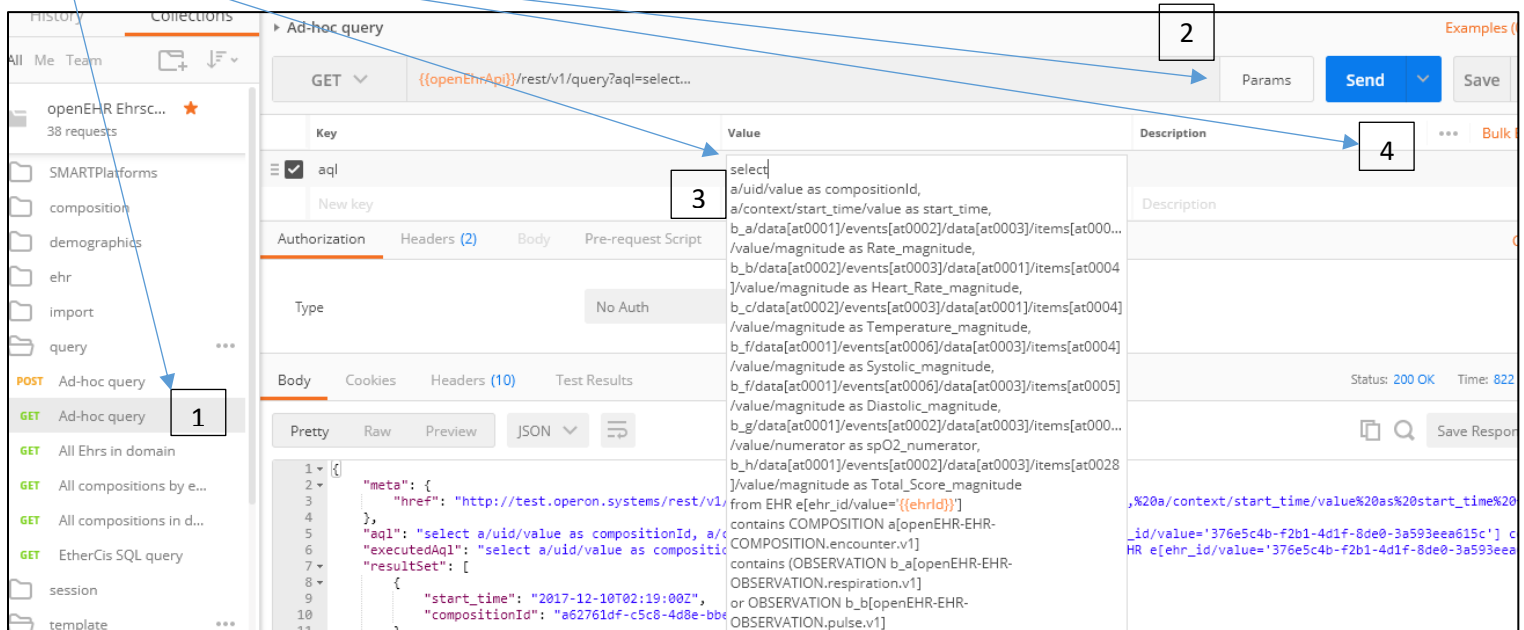
Navegue até a pasta `query` e selecione `Ad-hoc Query`.

Abaixo está a sintaxe a ser utilizada para recuperar as últimas 5 Compositions de Sinais Vitais do paciente. A documentação sobre como criar consultas AQL é abordada pela especificação do openEHR. Por enquanto, simplesmente copie e cole a sintaxe abaixo:

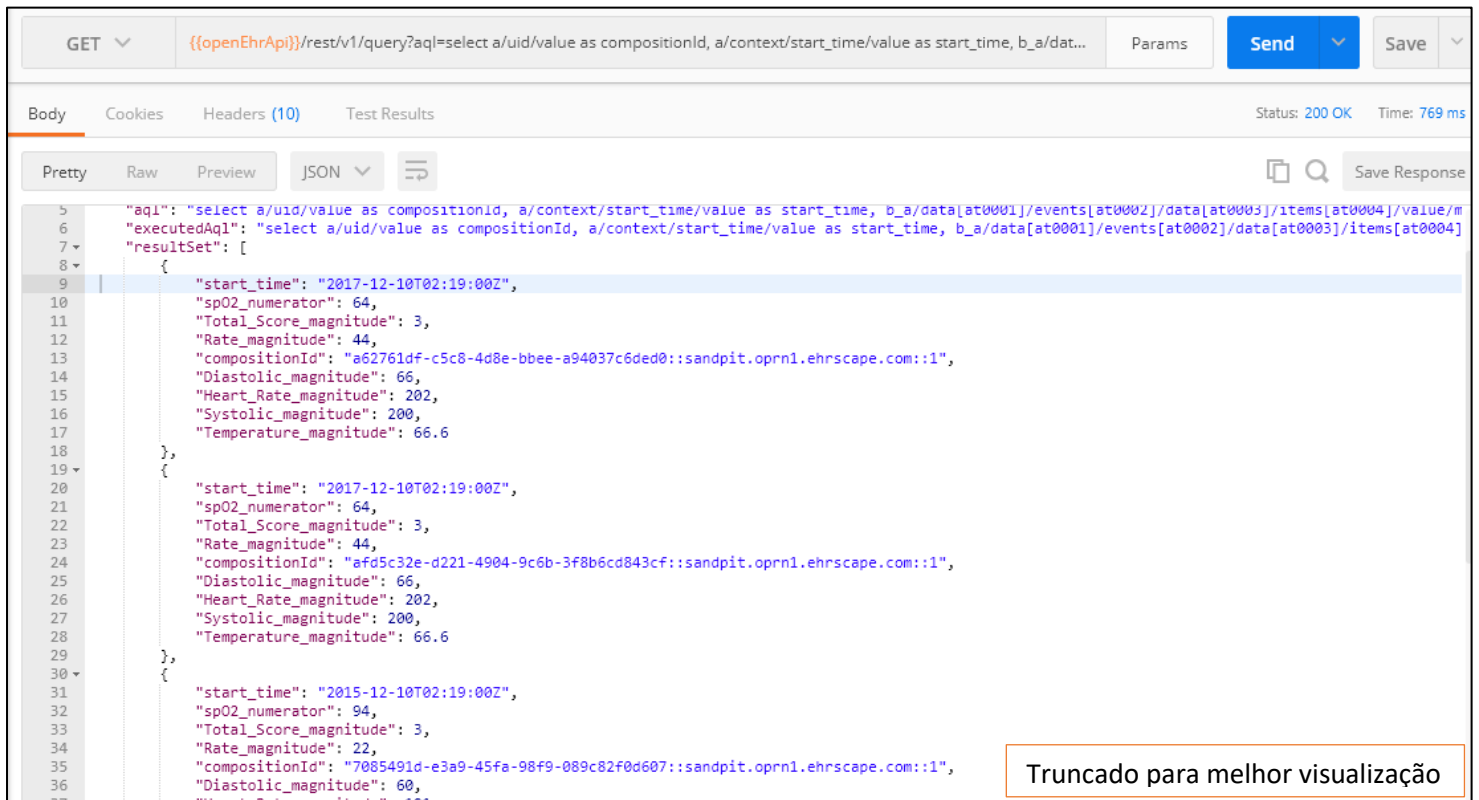
```
select a/uid/value as compositionId, a/context/start_time/value as
start_time,
b_a/data[at0001]/events[at0002]/data[at0003]/items[at0004]/value/magni
tude as Rate_magnitude,
b_b/data[at0002]/events[at0003]/data[at0001]/items[at0004]/value/magni
tude as Heart_Rate_magnitude,
b_c/data[at0002]/events[at0003]/data[at0001]/items[at0004]/value/magni
tude as Temperature_magnitude,
b_f/data[at0001]/events[at0006]/data[at0003]/items[at0004]/value/magni
tude as Systolic_magnitude,
b_f/data[at0001]/events[at0006]/data[at0003]/items[at0005]/value/magni
tude as Diastolic_magnitude,
b_g/data[at0001]/events[at0002]/data[at0003]/items[at0006]/value/numer
ator as spO2_numerator,
b_h/data[at0001]/events[at0002]/data[at0003]/items[at0028]/value/magni
tude as Total_Score_magnitude from EHR e[ehr_id/value='{{ehrId}}']
contains COMPOSITION a[openEHR-EHR-COMPOSITION.encounter.v1] contains
(OBSERVATION b_a[openEHR-EHR-OBSERVATION.respiration.v1] or
OBSERVATION b_b[openEHR-EHR-OBSERVATION.pulse.v1] or OBSERVATION
b_c[openEHR-EHR-OBSERVATION.body_temperature.v1] or OBSERVATION
b_f[openEHR-EHR-OBSERVATION.blood_pressure.v1] or OBSERVATION
b_g[openEHR-EHR-OBSERVATION.indirect_oximetry.v1] or OBSERVATION
b_h[openEHR-EHR-OBSERVATION.news_rcp_uk.v1]) where a/name/value=
'Nursing Vital Signs Observations' order by a/context/start_time/value
desc offset 0 limit 5
```

## InterOpera

Na janela **Ad-hoc query**, clique em **Params**, insira o texto copiado no campo **Value** e clique no botão **Send**



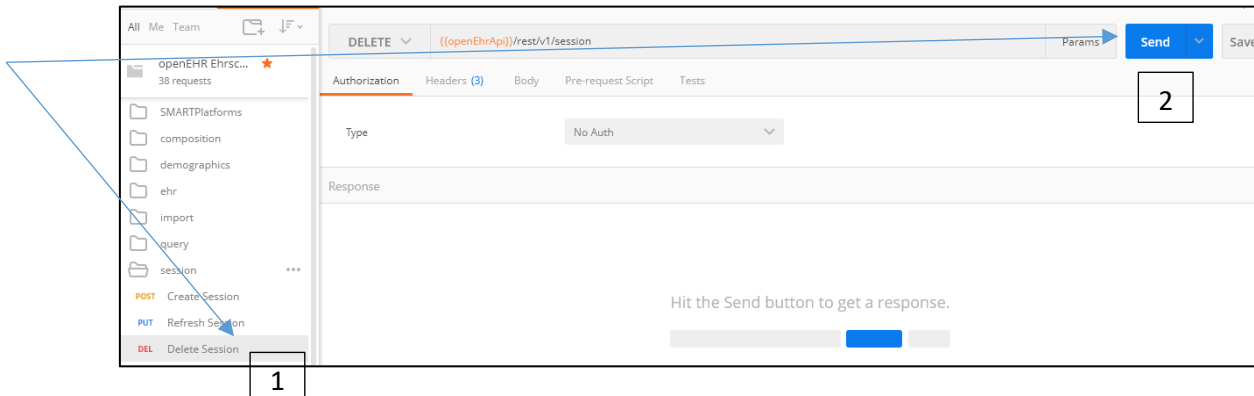
O resultado são as últimas 5 Compositions para observações de Sinais Vitais.



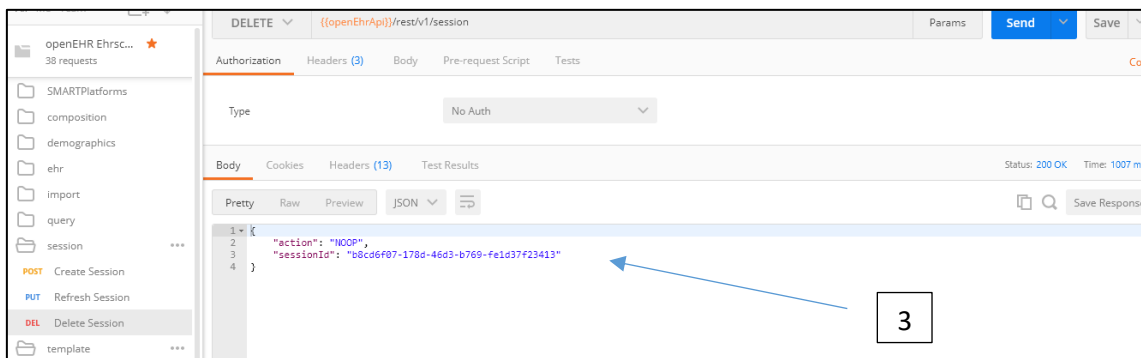
## ENCERRANDO A SESSÃO

O passo final é finalizar a sessão openEHR.

Para fazer isso, navegue até a pasta **session**, selecione **Delete Session** e Clique no botão **Send**.



O resultado mostrará uma sessão nula indicando que não há mais nenhuma sessão aberta.



Você pode acessar este tutorial em inglês: [Cod4Health](https://www.cod4health.com/interopera-tutorial/interopera-10-encerrando-a-sessao/).