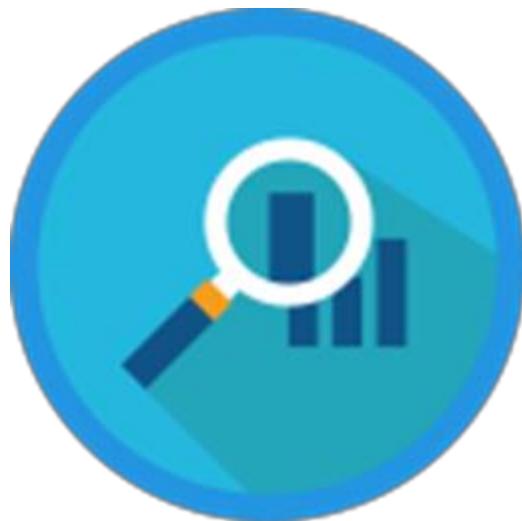


# **Query Tool Application Programming Interface (API)**





**BRICS**

---

**THIS PAGE HAS BEEN LEFT INTENTIONALLY BLANK**

---



## Table of Contents

<b>Query Tool Application Programming Interface</b> .....	3
<b>Objective</b> .....	3
Authentication.....	4
Authentication.....	4
Study API.....	5
Get All Study Information .....	5
Get all studies associated with a form structure.....	7
Form Structure API.....	8
Get form structures for a study .....	8
Data Elements API .....	10
Get data elements for a form structure .....	10
Data API.....	12
Get data from multiple form structures without doing joins .....	12
Get data with filter and joins.....	15



## QUERY TOOL APPLICATION PROGRAMMING INTERFACE



The Query Tool Application Programming Interface (API), a RESTful API that is connected to a micro service that allows users to make HTTPS requests using GET and POST. Users will be able to query data from different endpoints using Python, JavaScript, R and other tools that use Restful APIs.

### OBJECTIVE

This chapter provides information for users on how to:

- ❖ To log in to the Query Tool API
- ❖ Enter parameter information for each endpoint

For information about the Query Tool API Endpoints please access the API Documentation for the BRICS Instance:

BRICS Instance	Query API
NEI	<a href="https://brics.nei.nih.gov/gateway/query-api/swagger-ui/index.html">https://brics.nei.nih.gov/gateway/query-api/swagger-ui/index.html</a>
CdRns	<a href="https://cdrns.nih.gov/gateway/query-api/swagger-ui/index.html">https://cdrns.nih.gov/gateway/query-api/swagger-ui/index.html</a>
FITBIR	<a href="https://fitbir.nih.gov/gateway/query-api/swagger-ui/index.html">https://fitbir.nih.gov/gateway/query-api/swagger-ui/index.html</a>
PDBP	<a href="https://pdgp.ninds.nih.gov/gateway/query-api/swagger-ui/index.html">https://pdgp.ninds.nih.gov/gateway/query-api/swagger-ui/index.html</a>

Note the Query-API is designed similarly for all BRICs instances: Endpoint names will be identical except for the .gov name of the BRICs Instance (as shown above in bold)

For more information and examples on the endpoints, please refer to the Jupyter Notebook Version of this API Manual and the additional Demo Notebooks.



# AUTHENTICATION

## AUTHENTICATION

To access the Query Tool API the user needs to provide an authentication 'token' in the header of each API request (Figure 1). This token is found by logging on to the user's BRICs account (FITBIR etc) and going to the 'Account Management' tab (Figure 2). Here you will be able to cut and paste the token and set it as a variable in your program of choice (Jupyter Notebook, R, JavaScript, etc).

*\*Note: The authentication method for the Query-API changed to the above workflow after RAS (Researcher Auth Service) updates in Nov 2023.*

**Figure 1**

```
# Define API headers and query URL
queryurl ="https://fitbir.nih.gov/gateway/query-api/data/csv"

#Typically Included in Request Header
#Accept - For specifying the type of content that can be accepted in a response. (json etc) See cor
#Content-type - For indicating the type of content that's included in the request body. (json etc)
#Authorization - in the format of 'Bearer' + token

headers = {
    'accept': 'application/json', # note this cannot be 'application/csv' as it was in the old example
    'Content-type': 'application/json',
    'Authorization': 'Bearer ' + token }
```

**Figure 2**

The figure consists of two screenshots. The top screenshot shows the FITBIR navigation bar with 'Account Management' highlighted. An arrow points from this tab to the bottom screenshot. The bottom screenshot shows the 'API Tokens' page. On the left, there is a grid of icons for various modules: ProFoRMS, Subject Management, Data Dictionary, Data Repository, Query, Meta Study, Account Management, and Forum. The 'Account Management' module is circled in orange. On the right, the 'API Tokens' section is shown with a detailed description of what an API token is and how to use it. A 'Copy' button is highlighted with an orange circle at the bottom of the text area.

## STUDY API

The following endpoints will return the study profile information for a study. As mentioned in the Authentication section, a token is needed to retrieve data for all subsequent endpoints.

There are two endpoints that allow users to do the following:

1. Get study profile information for all studies.
2. Get study profile information for one study using a Study Prefix
3. Get studies that have submitted data to a form structure using the Form Structure Shortname.

Below are the endpoints for retrieving information about studies and examples.

### GET ALL STUDY INFORMATION

This service will return all the studies that have data in the instance. Optional it will return information for a study with the study Prefix ID (Study ID) provided. The Study Prefix ID can be retrieved using this endpoint.

*The following information is needed*

**Endpoint URL:** <https://fitbir.nih.gov/cit.nih.gov/gateway/query-api/study>

**Parameters:**

<b>Headers</b>	Response content type: application/json Content-Type: application/json Authorization: Bearer + <b>Token</b>
<b>Data</b>	Optional: prefixedId =
<b>GET Response</b>	<pre>response = requests.get("https://fitbir.nih.gov/gateway/ query-api/study", headers=headers)  response = requests.get("https://bricsnei- stage.cit.nih.gov/gateway/query- api/form/study?prefixedId=<b>STUDYID</b>", headers=headers)</pre>

**Figure 1: Example of Input**

```
#get study prefix from title
url = "https://" + BRICSInstance + "/gateway/query-api/study"

headers = {
    'accept': 'application/json',
    'Content-type': 'application/json',
    'Authorization': 'Bearer ' + token
}

query = requests.get(url, headers=headers)
query

<Response [200]>
```

**Output**

Output Format	JSON
Output Description	
Output Object	Required (Yes/No)
Study Title	Yes
Study ID	Yes
Abstract	Yes
Principal Investigator	Yes

**Figure 2: Example of Output**

```
output = query.json()
output
{
    'title': 'Sensory Integration Balance Deficits in Complex mTBI: Can Early Initiation of Rehabilitation with Wearable Sensor Technology Improve Outcomes?',
    'status': 'Public',
    'id': 'FITBIR-STUDY0000381',
    'pi': 'Laurie King',
    'fundingAmount': 4652120.0,
    'abstract': 'Traumatic brain injury (TBI) and post-traumatic headache (PTH) are common conditions that exert substantial impacts in the military and in the civilian population. TBI is a signature injury of U.S. Soldiers during modern warfare with 20% of Operation Enduring Freedom and Operation Iraqi Freedom Veterans having experienced TBI, 75% of which are mild TBI (mTBI). In addition, approximately 1.7 million American civilians seek medical attention each year for TBI and there are 1.6-3.8 million sports-related mTBIs annually. Headache is the most common symptom following mTBI, with estimates of headache prevalence following mTBI as high as 90%. PTH is often persistent (i.e., endures for >3 months following the injury), with about 66% of people with mTBI reporting continued headaches at 3 months post-injury. Although progress has been achieved in understanding PTH epidemiology, there are significant shortcomings in the description of PTH mechanisms, diagnosis, and treatment. A major challenge for the investigation, diagnosis, a
```

## GET ALL STUDIES ASSOCIATED WITH A FORM STRUCTURE

Returns all the studies that have data submitted to the form structure

*The following is needed*

**EndpointURL:** <https://fitbir.nih.gov/gateway/query-api/study/form?formName=>

### Parameters:

<b>Headers</b>	Response content type: application/json Content-Type: application/json Authorization: Bearer + Token
<b>Data</b>	Form Structure Short Name
<b>GET Response</b>	response = requests.get("https://fitbir.nih.gov/gateway/query-api/study/form?formName=Form Structure Shortname", headers=headers)

**Figure 1: Example of Input**

```
url = "https://" + BRICSInstance + "/gateway/query-api/study/form?formName="
header = {
    'accept': 'application/json',
    'Content-type': 'application/json',
    'Authorization': 'Bearer ' + token
}

formstructureshortname = input()
```

GCS

```
query = requests.get(url + formstructureshortname, headers=header)
query
```

<Response [200]>

### Output

<b>Output Format</b>	JSON
----------------------	------

### Output Description

<b>Output Object</b>	<b>Required (Yes/No)</b>
Study Title	Yes
Study ID	Yes
Abstract	Yes
Principle Investigator	Yes

**Figure 2: Example of output**

```
formstructureinformation = query.json()
formstructureinformation

[{'form': 'GCS',
 'studies': [{}{'abstract': 'Activity-related concussion is a growing epidemic particularly in c
ollegiate athletes. The cumulative effective of multiple concussions necessitates that return-t
o-activity testing post-concussion are both appropriate and sensitive in determining the readin
ess of the individual to return to the field. Postural stability has been shown to be a good in
dicator of recovery following a concussive episode. The Balance Error Scoring System (BESS) is
the most widely used clinical assessment of postural stability with post-concussion athletes, y
et it lacks adequate quantitative metrics needed to detect subtle changes in stability. The Vir
tual Environment Traumatic brain injury (TBI) Screen (VETS) device is a new assessment tool tha
t can potentially be used to measure postural stability in healthy and neurologically impaired
individuals. The VETS is a variation of our previously approved (closed) IRB approved protocol
(20839) for validation of our Therawii (Portable, Instrumented Postural Stability System). The
VETS uses a large wide screen TV to display a virtual reality scene to the subject. Two hundred
healthy adults will participate in this study. Subjects will perform standing postural tasks us
ing the VETS, clinical examination and neurocognitive tests. The purpose of this study is to va
lidate the use of the VETS as a measure of postural control for assessing an individual?s readi
ness to return to sport or activity.'},
 'status': 'Public',
 'id': 'FITBIR-STUDY0000237',
```

## FORM STRUCTURE API

The Form Structure API uses the Study Prefix ID and returns the form structures that have data submitted against in a JSON format. To learn more about retrieving the Study Prefix ID, please refer to the section on the Study API endpoints.

### GET FORM STRUCTURES FOR A STUDY

Returns all the form structures that have data submitted for the study.

*The following is needed*

**Endpoint URL:** <https://fitbir.nih.gov/gateway/query-api/form/study?prefixedId=>

#### Parameters:

<b>Headers</b>	Response content type: application/json Content-Type: application/json Authorization: Bearer + <b>Token</b>
<b>Data</b>	STUDYID
<b>GET Response</b>	response = requests.get("https://fitbir.nih.gov/gateway/query-api/form/study?prefixedId=STUDYID", headers=headers)



### Figure 1: Example of Input

```
url = "https://" + BRICSInstance + "/gateway/query-api/form/study?prefixedId="

headers = {
    'accept': 'application/json',
    'Content-type': 'application/json',
    'Authorization': 'Bearer ' + token
}
studyid = input("Enter Study PrefixID ")

Enter Study PrefixID FITBIR-STUDY0000267

query = requests.get(url + studyid, headers = headers)
query

<Response [200]>
```

### Output

Output Format	JSON
---------------	------

### Output Description

Output Object	Required (Yes/No)
Study Prefix ID	Yes
Form Structure Short Name	Yes
Form Structure Title	Yes

### Figure 2: Example of Output

```
studyformstructuredata = query.json()
studyformstructuredata

[{'studyId': 'FITBIR-STUDY0000267',
 'forms': [{'id': 2314,
            'shortName': 'AIS',
            'title': 'Abbreviated Injury Scale (AIS)',
            'version': '1.0'},
           {'id': 2727,
            'shortName': 'AIS_Appdx_TRACKTBI',
            'title': 'Abbreviated Injury Scale (AIS) Appendix for TRACK-TBI',
            'version': '1.0'},
           {'id': 2575,
            'shortName': 'AUDITC',
            'title': 'Alcohol Use Disorders Identification Test - Consumption Questions (AUDIT-C)',
            'version': '1.2'},
           {'id': 2782,
            'shortName': 'AUDITC_Appdx_TRACKTBI',
            'title': 'Alcohol Use Disorders Identification Test - Consumption Questions (AUDIT-C) Appendix for TRACK-TBI',
            'version': '1.0'},
           {'id': 2641,
            'shortName': 'Adverse_Events_FITBIR',
            'title': 'Adverse Events for FITBIR'}]
```



## DATA ELEMENTS API

The data element API uses the form structure short name and returns the data elements within that form structure. The output will be in a JSON format and provide information about the data element such as the position in the form structure, the title and variable name.

The information about the data element is useful when filtering data in the Data API.

Below is information and examples for retrieving the data elements for that form structure.

### GET DATA ELEMENTS FOR A FORM STRUCTURE

Return all data elements associated with the form structure.

*The following is needed*

**Endpoint URL:** <https://fitbir.nih.gov/gateway/query-api/dataElement/form/>

#### Parameters:

<b>Headers</b>	Content-Type: application/json Authorization: Bearer + Token
<b>Data</b>	Form Structure Short Name
<b>GET Response</b>	response = requests.get("https://fitbir.nih.gov/gateway/query-api/dataElement/form/Form Structure Shortname", headers=headers)

**Figure 1: Example of Input**

```
url = "https://" + BRICSInstance + "/gateway/query-api/dataElement/form/"

headers = {
#     'accept': 'application/json',
    'Content-type': 'application/json',
    'Authorization': 'Bearer ' + token
}

# print("Input Form Structure Short Name")
formstructureshortname = input()
```

GCS

```
| dataelementapiquery = requests.get(url + formstructureshortname,headers = headers)
| dataelementapiquery
<Response [200]>
```

## Output

Output Format	JSON
---------------	------

### Output Description

Output Object	Required (Yes/No)
Data Element Name	Yes
Form Structure Group Name	Yes
Data Element Title	Yes
Data Element Short Description	Yes
Data Element Data Type	Yes

**Figure 2: Example of Output**

```
dataelementapiinformation = dataelementapiquery.json()
dataelementapiinformation

[{'name': 'Main',
 'position': 0,
 'threshold': 1,
 'dataElements': [{'id': 69410,
   'name': 'GUID',
   'title': 'GUID',
   'description': 'Global Unique ID (GUID) which uniquely identifies a subject',
   'dataType': 'GUID',
   'inputRestriction': 'Free-Form Entry',
   'requiredType': 'Required'},
  {'id': 69738,
   'name': 'SubjectIDNum',
   'title': 'Subject identifier number',
   'description': 'An identification number assigned to the participant/subject within a given protocol or a study.',
   'dataType': 'Alphanumeric',
   'inputRestriction': 'Free-Form Entry',
   'requiredType': 'Optional'},
  {'id': 70501,
   'name': 'AgeYrs',
   'title': 'Age in years'}]]
```



## DATA API

The Data API allows users to retrieve data for one or more studies and form structures. Similar to the Query Tool, users are able to (1) Download data for form structures within a study, (2) Join form structures for one or more studies, (3) Filter on data elements with the advance Boolean Search.

Below are the endpoints and examples for retrieving data from studies and form structures.

---

### GET DATA FROM MULTIPLE FORM STRUCTURES WITHOUT DOING JOINS

Returns data for multiple form structures without doing joins. The output is one .csv table per form/study input specified.

---

*The following is needed*

---

**Endpoint URL:** <https://fitbir.nih.gov/gateway/query-api/data/bulk/form/study>

**Parameters:**

<b>Headers</b>	Content-Type: application/json Authorization: Bearer + <b>Token</b>
<b>Data</b>	Form Structure Short Name Optional: Study Prefix IDs
<b>GET Response</b>	response = requests.get("https://fitbir.nih.gov/gateway/query-api/data/bulk/form/study, headers=headers,json=data)

**Figure 1: Example of Input**

```
# set input parameters:  
# Input any form structures associated with your BRICS instance of interest (use short name)  
form1 = 'GCS'  
form2 = 'ImagingDiffusion'  
  
# Input any study study prefix ID  
study1 = ['FITBIR-STUDY000409']  
study2 = ['FITBIR-STUDY000267']  
  
multipleformsheader = {  
    'Content-type': 'application/json',  
    'Authorization': 'Bearer ' + token  
}  
  
multipleformsurl = "https://" + BRICSInstance + "/gateway/query-api/data/bulk/form/study"
```



```
multipleformsfilter = {
    "flattened": "false",
    "formStudies": [
        {
            "form": form1,
            "studies": study1
        },
        {
            "form": form1,
            "studies": study2
        },
        {
            "form": form2,
            "studies": study1
        },
        {
            "form": form2,
            "studies": study2
        }
    ],
    "outputFormat": "csv"
}
```

```
multipleformsquery = requests.post(multipleformsurl,headers = multipleformsheader,json = multipleformsfilter)
multipleformsquery
<Response [200]>
```

## Output

### Output Format

### Output Description

Output Object	Required (Yes/No)
CSV File of Data	Yes

**Figure 2: Example of Output**

query_result_GCS_2024-04-10T09-30-46451...	4/10/2024 9:31 AM	Microsoft Excel Com...	436 KB
query_result_GCS_2024-04-10T09-31-00178...	4/10/2024 9:31 AM	Microsoft Excel Com...	18,771 KB
query_result_ImagingDiffusion_2024-04-10T...	4/10/2024 9:31 AM	Microsoft Excel Com...	151 KB
query_result_ImagingDiffusion_2024-04-10T...	4/10/2024 9:31 AM	Microsoft Excel Com...	3,439 KB



Where:

query\_result\_GCS\_2024-04-10T09-30-464513621180269995292

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
Study ID	Dataset	GCS.Main	GCS.Main	GCS.Main	GCS.Main	GCS.Main	GCS.Main	GCS.Main	GCS.Main	GCS.Main	GCS.Main	GCS.Form	GCS.Form	GCS.Form	GCS.Form	GCS.Glasg	GCS.Glasg
409	FITBIR-DA	TBIHF83GXD0			2			2	12.86	Case	CAMRI						
409	FITBIR-DA	TBITT216PCF			9			6	1.69	Case	CA						
409	FITBIR-DA	TBIGF501TT3			15			2	12.94	Case	CAMRI						
409	FITBIR-DA	TBINF55ODRZ			8			2	1.69	Case	CAMRI						
409	FITBIR-DA	TBIGF501TT3			15			2	12.69	Case	CAMRI						
409	FITBIR-DA	TBINC471RT2			12			2	1.7	Case	CAMRI						
409	FITBIR-DA	TBIRD354FCX			15			2	1.7	Case	CAMRI						
409	FITBIR-DA	TBIGF501TT3			15			2	12.77	Case	CAMRI						
409	FITBIR-DA	TBITN052XX8			11			2	1.67	Case	CAMRI						
409	FITBIR-DA	TBITP964XWF			0			2	1.67	Case	CAMRI						
409	FITBIR-DA	TBITR17HDX			13			2	1.67	Case	CAMRI						
409	FITBIR-DA	TBIGF501TT3			15			2	12.52	Case	CAMRI						
409	FITBIR-DA	TBIGF501TT3			15			2	12.27	Case	CAMRI						
409	FITBIR-DA	TBIGF501TT3			15			2	12.35	Case	CAMRI						

query result GCS 2024-04-10T09-30-464513621180269995292

query\_result\_ImagingDiffusion\_2024-04-10T09-31-03786940054294527485

query result ImagingDiffusion 2024-04-10T09-31-0515687447877164834836

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
Study ID	Dataset	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi	ImagingDi
267	FITBIR-DA	TBIZY358UAZ			43	2015-01-2	VCU	14	Case	2 week vis	2015-01-2 /DTI_Site1DICOM			MRI		/DTI_Site1e96	
267	FITBIR-DA	TBIBA600EKZ			22	2015-07-1	VCU	180	Case	6 month v	2015-07-1 /DTI_Site1DICOM			MRI		/DTI_Site18b7	

Note the blank rows for the ImagingDiffusion results correspond to the fact that multiple ImagingDiffusion.Diffusion Derived Data ImgFiles (FA etc) can exist for a single scan ImagingDiffusion.Image Information ImgFile



## GET DATA WITH FILTER AND JOINS

Returns data with filters and joins data into a single output for up to five form structures. Data can be returned in two formats: text (csv) and JSON. Note that if the study is not specified the output will include/search any study WITH AT LEAST ONE of the queried forms. (In the .csv output, the form data element columns corresponding to any additional forms queried for, but not used by the given study, will be left as NaN) If you want to find/compile data for only those studies sharing a common set of specified forms please see BRICSAPIDemo\_FindStudiesUsingTwoOrMoreCommonForms

*The following is needed*

**Endpoint URL:** (1) <https://fitbir.nih.gov/gateway/query-api/data/csv>

(2) <https://fitbir.nih.gov/gateway/query-api/data/json>

**Parameters:**

<b>Headers</b>	Response content type: application/zip or application/json Content-Type: application/json Authorization: Bearer + <b>Token</b>
<b>Data</b>	Filter on data elements, form structure and studyID
<b>GET Response</b>	response = requests.get("https://fitbir.nih.gov/gateway/query-api/data/bulk/form/study, headers=headers,json=data)

**Define Filter (input to both CSV and Json)**

```
filterForQuery = {
    "formStudy": [
        {
            "form": 'GCS',
            "studies": ['FITBIR-STUDY0000267']
        },
        {
            "form": "DemogrFITBIR",
            "studies": ['FITBIR-STUDY0000267']
        },
    ],
    "filter": [
        {
            "dataElement": "AgeYrs",
            "form": "GCS",
            "repeatableGroup": "Main",
            "operator": "AND",
            "rangeStart": "20",
            "rangeEnd": "50",
            "mode": "inclusive"
        },
        {
            "dataElement": "GenderTyp",
            "form": "DemogrFITBIR",
            "repeatableGroup": "Subject Demographics",
            "value": [
                "Female"
            ]
        }
    ]
}
```



**Figure 1: Example of CSV Input**

```

queryurl = "https://" + BRICSIstance + "/gateway/query-api/data/csv"

headers = {
    'accept': 'application/zip',
    'Content-type': 'application/json',
    'Authorization': 'Bearer ' + token }

query = requests.post(queryurl, headers=headers, json=filterForQuery)
query

<Response [200]>

```

**Figure 2: Example of JSON Input**

```

queryurl ="https://" + BRICSIstance + "/gateway/query-api/data/json"

headers = {
    'accept': 'application/json',
    'Content-type': 'application/json',
    'Authorization': 'Bearer ' + token }

query = requests.post(queryurl, headers=headers, json=filterForQuery)
query

<Response [200]>

```

## Output CSV

<b>Output Format</b>	Text
----------------------	------

## Output Description

<b>Output Object</b>	<b>Required (Yes/No)</b>
CSV Format with Data	Yes

**Figure 3: Example of Output**

```

pd.set_option('display.max_columns', None)
dataset = query.text
texttddf = StringIO(dataset)
fitbir_data = pd.read_csv(texttddf, sep=",", low_memory=False)
fitbir_data.head()

```

Information from multiple forms is joined into a single table by GUID as in the Web Query Tool

GUID	GCS.StudyID	GCS.Dataset	GCS.Main.GUID	GCS.Main.ASSOCIATEDGUID	GCS.Main.SubjectIDNum	GCS.Main.AgeYrs	DemogrFITBIR.SubjectDemographics.GenderTyp	DemogrFITBIR.SubjectDemographics.SexSubjectGenotyp	DemogrFITBIR.SubjectDemographics.SexSubjectGenotypOTH	DemogrFITBIR.SubjectDemographics.HandPrefTyp	DemogrFITBIR.SubjectDemographics.RaceUSACat
0	TBITW423YBL	267	FITBIR-DATA0011942	TBITW423YBL	NaN	NaN	27	Female	NaN	NaN	Left hand
1	TBITW423YBL	267	FITBIR-DATA0011942	TBITW423YBL	NaN	NaN	27	Female	NaN	NaN	Left hand
2	TBITW423YBL	267	FITBIR-DATA0011942	TBITW423YBL	NaN	NaN	27	Female	NaN	NaN	Left hand
3	TBITW423YBL	267	FITBIR-DATA0011942	TBITW423YBL	NaN	NaN	27	Female	NaN	NaN	Left hand
4	TBITW423YBL	267	FITBIR-DATA0011942	TBITW423YBL	NaN	NaN	27	Female	NaN	NaN	Left hand

Note: GCS (Glasgow Coma Scale) information may be collected at more than one time point. Above each row corresponds to one GCS measurement as is distinguished by GCS.Main.DaysSinceBaseline. The above patient has multiple measurements at different time intervals. The demographic information for the subject is joined and repeated for each of these measurement rows.



## Output Json

Output Format	JSON
---------------	------

### Output Description

Output Object	Required (Yes/No)
GUID	Yes
Form Structure Shortname	Yes
Study ID	Yes
Dataset ID	Yes
Form Structure Repeatable Group	Yes
Data Elements and Associated Data	Yes

Figure 4: Example of JSON Output of data

```
jsondata = query.json()
jsondata

[{'guid': 'TBITW423YBL',
 'forms': [{'name': 'GCSV1.2',
            'studyId': 'FITBIR-STUDY0000267',
            'datasetId': 'FITBIR-DATA0011942',
            'repeatableGroups': [{"name': 'Main',
                      'data': [[[{'GUID': 'TBITW423YBL'},
                                  {'SubjectIDNum': ''},
                                  {'AgeYrs': '27'},
                                  {'VitStatus': ''},
                                  {'VisitDate': ''},
                                  {'SiteName': '1'},
                                  {'DaysSinceBaseline': '0.740277778'},
                                  {'CaseControlInd': 'Case'},
                                  {'GeneralNotesTxt': 'CAMRI'}]]},
                      {'name': 'Form Administration',
                       'data': [[[{'ContextType': 'Other, specify'},
                                  {'ContextTypeOTH': '2727'},
                                  {'DataSource': 'Other, specify'},
                                  {'DataSourceOTH': '21588'}]]]}]}]
```