HLN CONSULTING
Innovators in Health Informatics

FHIR® for Immunization Information Systems (IIS)

FHIR in a Nutshell

- Fast Healthcare Interoperability Resources (FHIR)
- Next generation HL7 standard
- Set of Resources and a modern RESTful API for accessing them

FHIR in a Nutshell

- **FHIR Bulk Data** – Export large amounts of data from a FHIR server
- **SMART on FHIR** – health app interface based on FHIR/open standards
- **CDS Hooks** – CDS services to be triggered within clinician workflow
- **Subscriptions** – allows clients to request notifications on events
- **FHIR Extended Operations** – supports additional use cases
- **Implementation Guides** – standards-based ways of using FHIR in specific domains

HLN CONSULTING

Why FHIR for Immunization Data Access?

- More accessible to general purpose developers than HL7 v2
- Modern APIs, software, tools, resources, and support
- FHIR Bulk Query:
  - Efficiently access up-to-date immunization data
  - Reduce redundant queries
  - Easy to parse bulk data format
  - Modern authentication/authorization framework

HLN CONSULTING

Immunization FHIR Example

Patient Info

First Name: TEST Middle: A Last: HIMSS-PATIENT

Date of Birth: 04/20/2022 Age: 10m 16d

VALID DOSES					NEXT DUE
COVID-19 2 valid doses	10/20/2022 COV19 PF PFR 6m-5 6m 0d	11/20/2022 COV19 PF PFR 6m-5 7m 0d			Due Now (On or after 01/15/2023) Dose 3 Covid Record
Influenza					Due Now (On or after 10/20/2022) Dose 1
Hepatitis B 3 valid doses	04/20/2022 HepB ped/adol 0m 0d	06/20/2022 VAXELIS 2m 0d	08/20/2022 VAXELIS 4m 0d [1]	10/20/2022 VAXELIS 6m 0d	End of Series
DTaP/DT/Tdap/Td 3 valid doses	06/20/2022 VAXELIS 2m 0d	08/20/2022 VAXELIS 4m 0d	10/20/2022 VAXELIS 6m 0d		Due in the Future (07/20/2023 - 12/17/2023) Dose 4
Pneumo 1 valid doses	06/20/2022 PCV15 2m 0d				Past Due
Polio 3 valid doses	06/20/2022 VAXELIS 2m 0d	08/20/2022 VAXELIS 4m 0d	10/20/2022 VAXELIS 6m 0d		Due in the Future (04/20/2026 - 05/17/2029) Dose 4
Hib 3 valid doses	06/20/2022 VAXELIS 2m 0d	08/20/2022 VAXELIS 4m 0d	10/20/2022 VAXELIS 6m 0d		Due in the Future (04/20/2023 - 09/16/2023) Dose 4
Rotavirus 0 valid doses					Maximum Age Reached

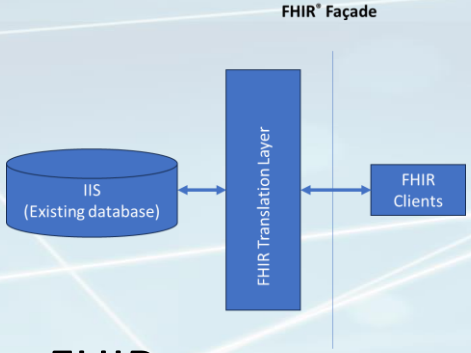

```
"name": [
    {
        "text": "TEST A HIMSS-PATIENT",
        "family": "HIMSS-PATIENT",
        "given": [
            "TEST",
            "A"
        ]
    }
],
"birthDate": "2022-04-20",
```



```
"status": "completed",
"vaccineCode": {
    "coding": [
        {
            "system": "http://hl7.org/fhir/sid/cvx",
            "code": "215",
            "display": "PCV15"
        }
    ],
    "text": "Pneumococcal conjugate PCV15, polysaccharide CRM197 con
},
"patient": {
    "reference": "11469529",
    "type": "Patient"
},
"occurrenceDateTime": "2022-06-20",
"recorded": "2023-01-01T07:10:39-05:00",
"primarySource": true,
"manufacturer": {
    "reference": "MSD",
    "type": "Organization",
    "display": "Merck & Co., Inc."
},
"lotNumber": "FK0101",
"expirationDate": "2024-01-01",
"site": {
    "coding": [
        {
            "system": "http://terminology.hl7.org/CodeSystem/v2-0163
            "code": "LA",
            "display": "LEFT ARM"
        }
```


*Courtesy of the Rhode Island IIS

FHIR Server Models



- **FHIR Façade**
  - Data translation – translate FHIR REST calls to the underlying legacy database or service
  - Intermediate FHIR server – synchronize native FHIR storage to underlying legacy database or service
- **Native FHIR server**
  - FHIR storage is the operational data store

Rhode Island IIS Implementation

- Rhode Island Child and Adult Immunization Registry (RICAIR)
- FHIR Façade Model using open source HAPI FHIR server and SMART Backend Services Authorization
- Bulk Query - Helios FHIR Accelerator for Public Health
  - Match and Bulk Match
  - Predefined Groups, or search and define custom groups
  - Query Patient, Immunization, ImmunizationRecommendation, and ImmunizationEvaluation resources
  - Download up to 100k patients or more in a fraction of the time of HL7 v2 QBP/RSP

Rhode Island IIS Implementation

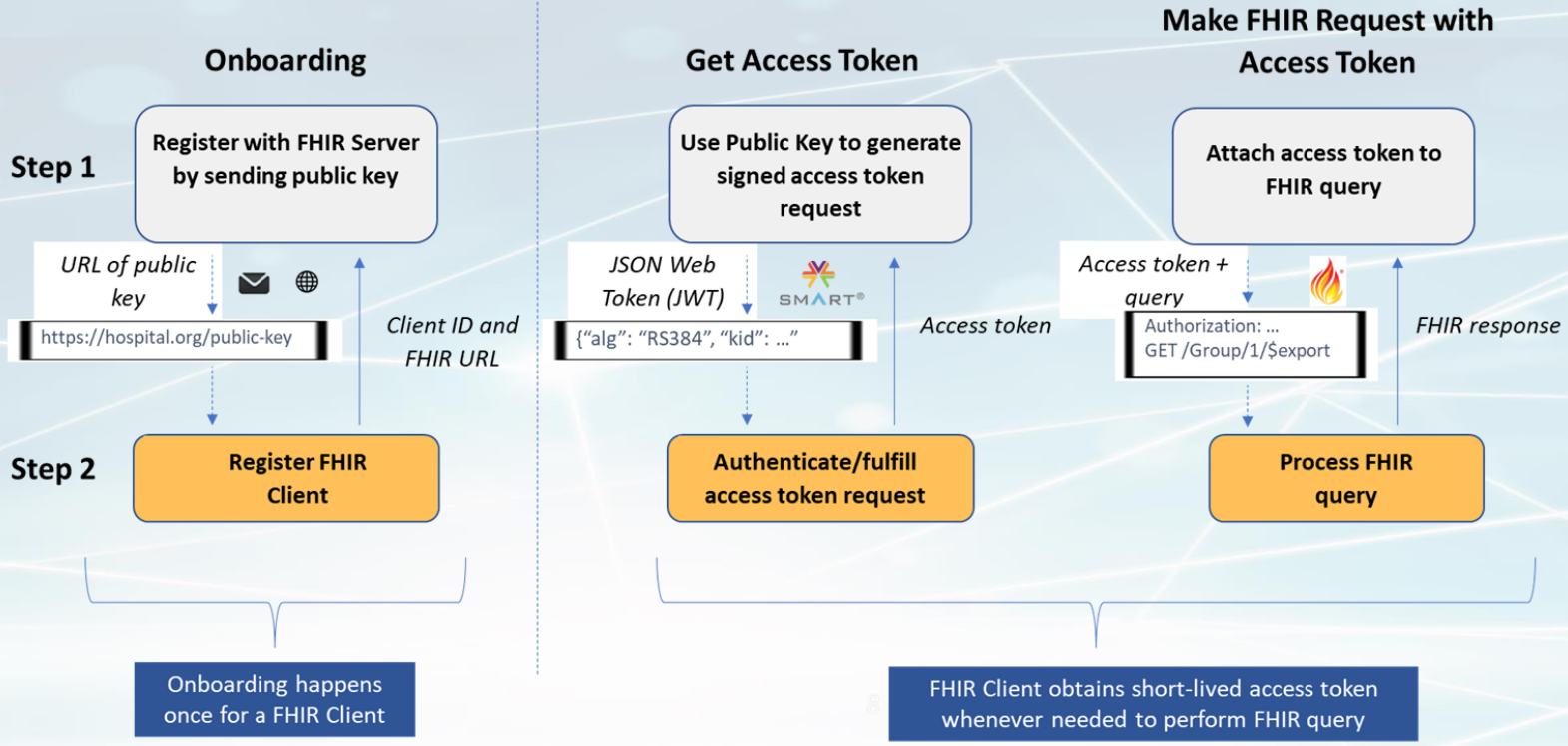
Functionality	FHIR functionality
Bulk Data Export	Group/[id]/$export
Patient Match	Patient/$match
Creation of Group resource	POST /Group {“member” […] }
Add Patient to Group	Group/[id]/$member-add
Remove Patient from Group	Group/[id]/$member-remove
Security/authentication – SMART Backend Services (OAuth/JWT)	/.well-known/smart-configuration POST /auth/token
Bulk Match	Patient/$match with [1..*]
HL7 v2-Mediated Matching (Automatic Group Creation)	FHIR Group/[id] maintained based on HL7v2 messages

FHIR IIS Authentication

- HL7 v2 Authentication
  - Typically involves username/password – tedious to share and can leak or be weak
  - Some use certificate authentication, but also tedious

- SMART Backend Services Authentication
  - Makes it possible to not have to share passwords with senders during onboarding
  - Partners provide IIS a URL containing their public key
  - Leverages public key authentication to securely provide FHIR access tokens

FHIR IIS Authentication Workflow



Using FHIR Bulk Data for App Modernization

- Replace legacy IIS web applications with modern front-ends that communicate with the FHIR back-end:
  - Immunization Data Entry/Update
  - Other potential uses:
    - Immunization Display, Patient Demographics, SMART Health Cards, School Forms, etc.
    - SMART on FHIR apps
    - Consumer apps

HLN CONSULTING

FHIR IIS demonstration

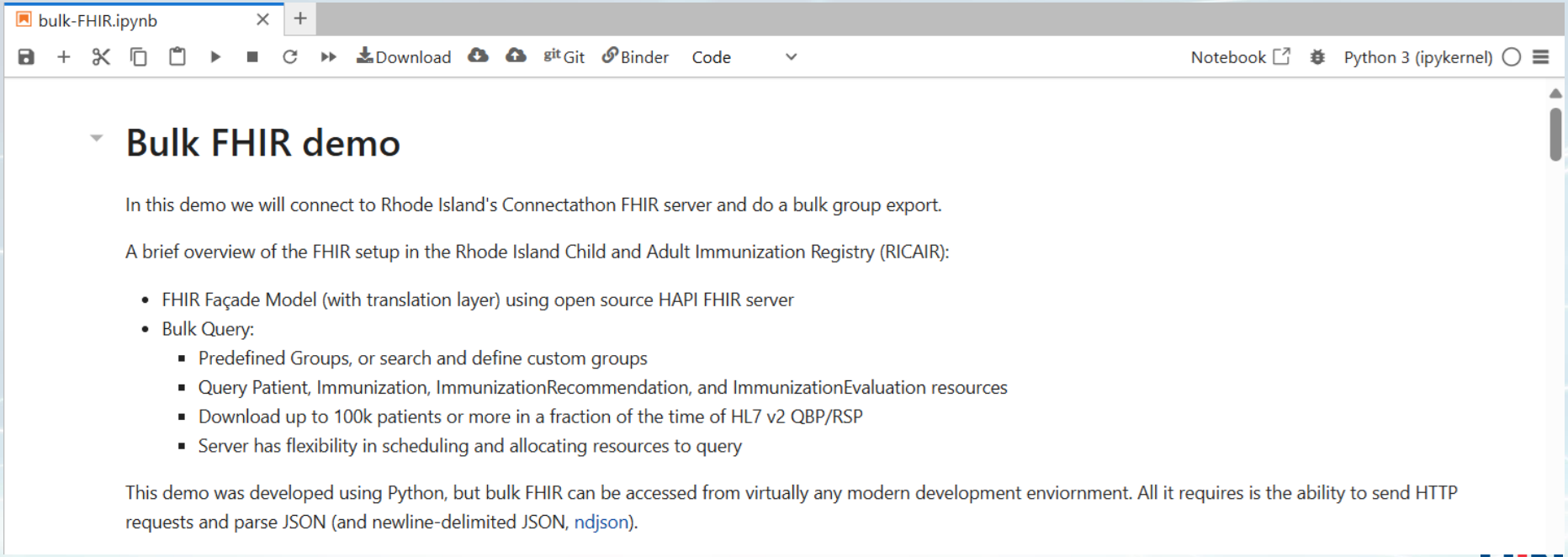
Python "Notebook"

Python Notebooks are interactive web applications that allow you to create and share documents containing live code, visualizations, and narrative text.

- Incredibly useful for HL7 Connectathons
- Participants can run and document their code in a readable and reproducible manner
- Excellent platform for collaboration

HLN CONSULTING

Demonstration screenshots



bulk-FHIR.ipynb

Download Git Binder Code Notebook Python 3 (ipykernel)

Setup dependencies

First we need to do some setup in Python. We import our dependencies for HTTP requests, date and time parsing, logging, etc.:

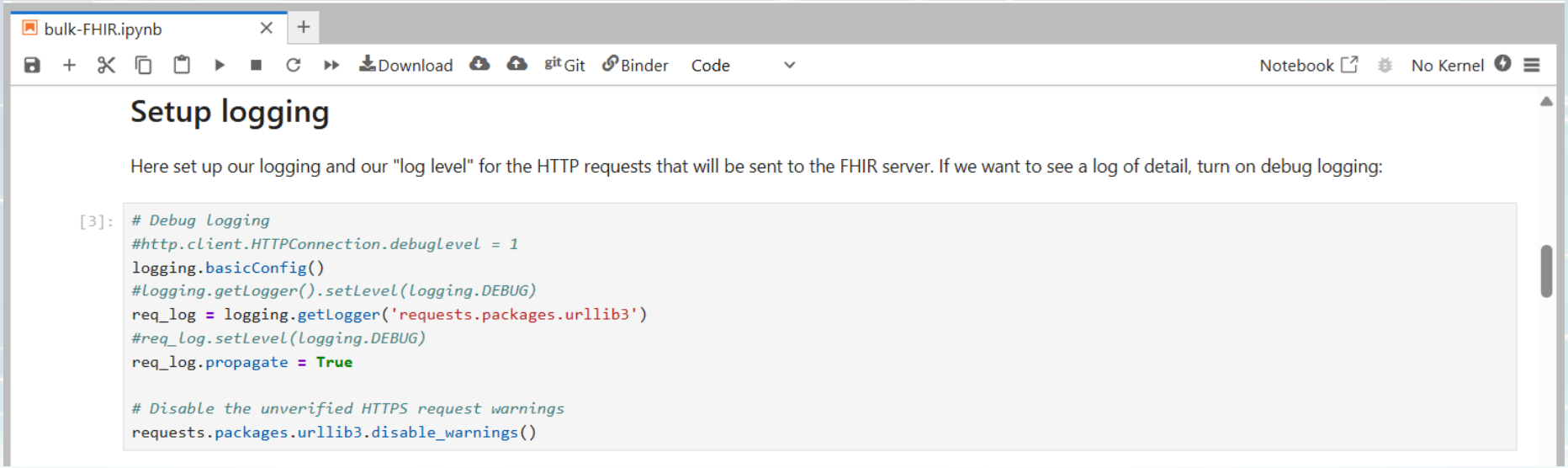
[1]:
```
pip install -q -r requirements.txt
```

```
Note: you may need to restart the kernel to use updated packages.
```

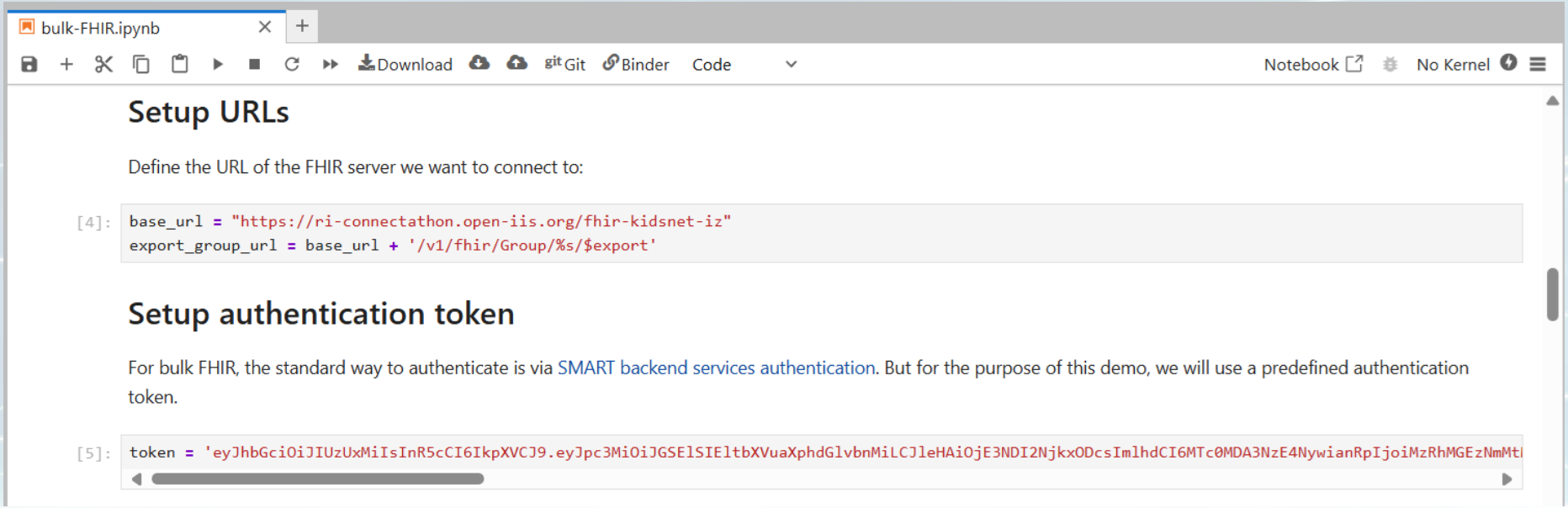
[2]:
```
#
# for the FHIR operations
#

import os.path
from datetime import datetime
from time import sleep
import requests as requests
import logging
import json
from pyfhirclient.FhirClient import FhirClient

#
# for presenting the data
#

import pandas
from IPython.display import JSON, HTML
import plotly.express as plotly
```

29 AM

Demonstration screenshots



Demonstration screenshots



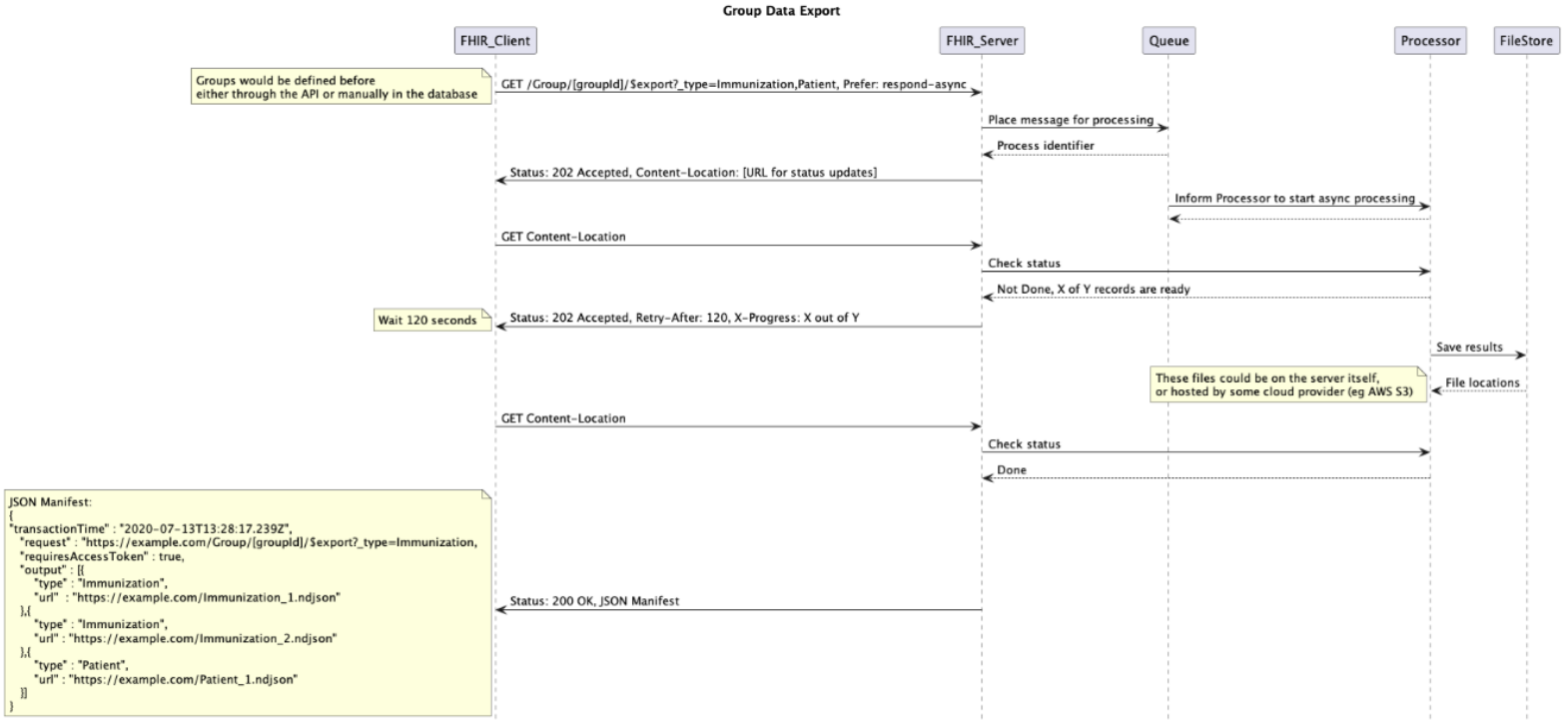
bulk-FHIR.ipynb

Download Git Binder Code Notebook No Kernel

Group export, polling, and save output functionality

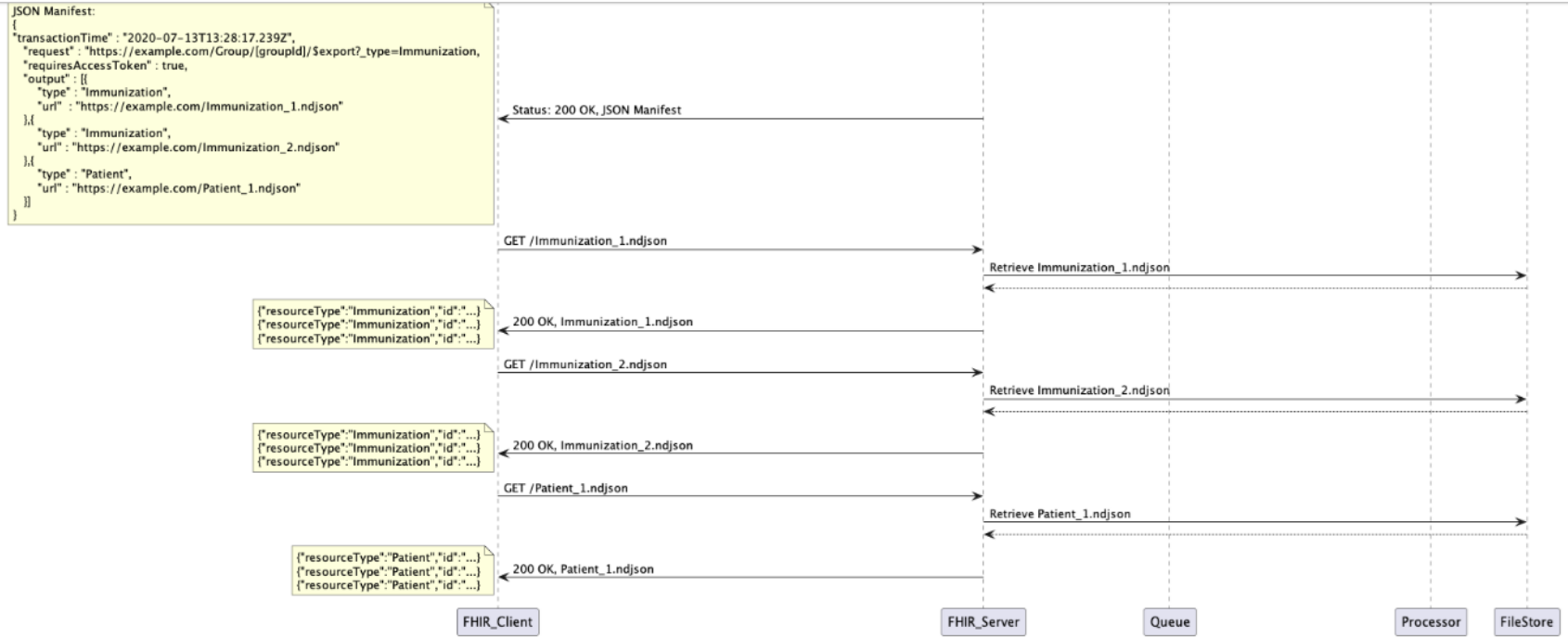
These functions actually do the work of the FHIR bulk export workflow:

1. Start Group Export
2. Poll
3. Save Output



bulk-FHIR.ipynb

Download Git Binder Code Notebook No Kernel



bulk-FHIR.ipynb

Download Git Binder Code Notebook No Kernel

Start Group Export

This function actually sends the group export request and then gets the URL of the place that we should poll to check to see if the output is ready. The request contains:

1. An authorization header containing the authentication token
2. A "Prefer" header that tells the FHIR server that this is to be an asynchronous workflow where we will check back later on the status of the request
3. The Group ID we want

[6]:
```
def start_group_export(token, group_id):
    headers = {
        "Authorization": "Bearer " + token,
        "Prefer": "respond-async"
    }

    response = requests.get(url=export_group_url % group_id, headers=headers, verify=False)
    if not response.ok:
        raise Exception("Unable to start export: " + str(response))

    print (f"Bulk group export started...")

    print (f"Sent request to: {export_group_url % group_id}")
    print (f"Server returned poll status URL: {response.headers['Content-Location']}")


    return response.headers['Content-Location']
```

HLN CONSULTING

bulk-FHIR.ipynb

Download Git Binder Code Notebook No Kernel

Poll

This function queries the URL provided by the FHIR server in the group export step in order to see if the ndjson files are ready.

For the purposes of the demo, when the server tells us to come back in 2 minutes (or at some later time) to check again, we override that and just wait 5 seconds.

Once the server is ready, it will provide URLs for downloading the ndjson (bulk) files for each FHIR resource that will be provided. By default, the RI FHIR server returns:

- Patient,
- Immunization,
- ImmunizationEvaluation, and
- ImmunizationRecommendation

[7]:

```
def poll(token, poll_url):
    headers = {
        "Authorization": "Bearer " + token
    }

    while True:
        print (f"Checking poll status URL {poll_url}...")

        response = requests.get(url=poll_url, headers=headers, verify=False)
        if not response.ok:
            raise Exception("Unable to poll status")

        if response.status_code == 200:
            print("Export is complete.")

            print ("Response: " + json.dumps(response.json()['output'], indent=4))
            return response.json()

        if 'X-Progress' in response.headers:
            progress = response.headers['X-Progress']
            print("Progress: " + progress)
```

Demonstration screenshots

bulk-FHIR.ipynb

Download Git Binder Code Notebook No Kernel

```
if 'X-Progress' in response.headers:
    progress = response.headers['X-Progress']
    print("Progress: " + progress)

if 'Retry-After' in response.headers:
    retry_after = response.headers['Retry-After']
    if retry_after.isnumeric():
        if retry_after == '120':
            retry_after = '5'
        print("Waiting " + retry_after + " seconds")
        sleep(int(retry_after))
    else:
        wait_until = datetime.strptime(retry_after, '%a, %d %b %Y %H:%M:%S GMT')
        seconds = (wait_until - datetime.now()).seconds
        print("Waiting until " + str(wait_until) + " which is " + str(seconds) + " seconds from now.")
        seconds = 5
        sleep(seconds)
```

CONSULTING

Save Output

Once the FHIR server tells us that the ndjson files are ready, we take the URLs provided from that "ready" response and we get them, print them out, and save them.

You can uncomment the "print" lines below in order to actually show the ndjson output and the pretty-printed JSON objects from the ndjson.

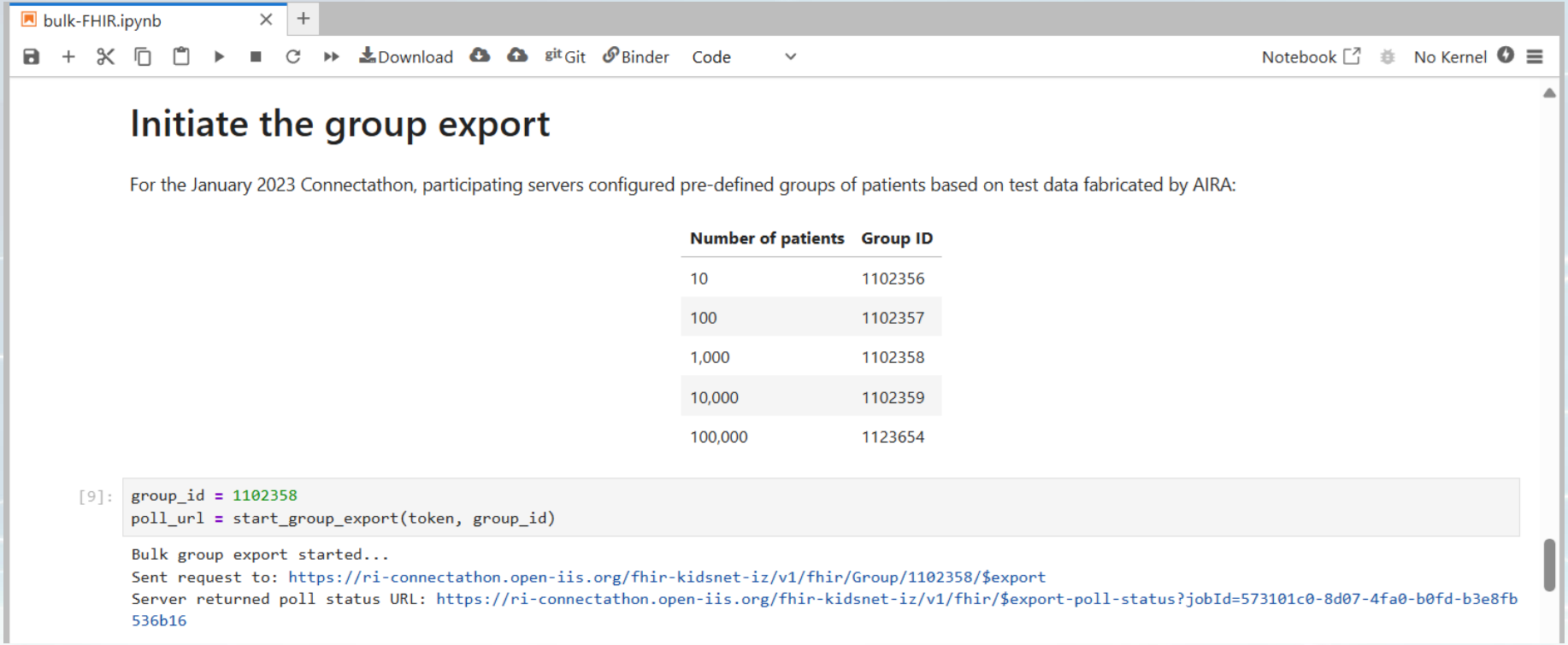
Reading the ndjson stream one line at-a-time

A typical way to read the ndjson response is to download it and load the entire ndjson file into memory.

An alternative that can be more suitable for large payloads is to read it one line at a time using the stream option and iter_lines function in the Requests library.

[8]:
```
def save_output_streaming(token, output):

    headers = {
        "Authorization": "Bearer " + token
    }

    file_list = []

    for entry in output['output']:
        type_ = entry['type']
        url: str = entry['url']
        print("type    : " + type_)
        print("url     : " + url)

        response = requests.get(url=url, headers=headers, stream=True, verify=False)
        if not response.ok:
            raise Exception("Unable to retrieve output: " + url)

        file_name = url.split('/')[-1]
        with open(file_name, "w") as file:
            for line in response.iter_lines():
                #print ("Response line: " + json.dumps(json.loads(line), indent=4))
                file.write(line.decode("utf-8"))
                file.write("\n")
```

Demonstration screenshots


bulk-FHIR.ipynb
Download
Git
Binder
Code
Notebook
No Kernel
Initiate the group export
For the January 2023 Connectathon, participating servers configured pre-defined groups of patients based on test data fabricated by AIRA:
Number of patients
Group ID
10
1102356
100
1102357
1,000
1102358
10,000
1102359
100,000
1123654
[9]:
group_id = 1102358
poll_url = start_group_export(token, group_id)
Bulk group export started...
Sent request to: https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/fhir/Group/1102358/$export
Server returned poll status URL: https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/fhir/$export-poll-status?jobId=573101c0-8d07-4fa0-b0fd-b3e8fb536b16

bulk-FHIR.ipynb

Download Git Binder Code Notebook No Kernel

Check the poll status URL to see if the bulk export is done

[10]:
```
output = poll(token, poll_url)
print (f"Output from FHIR server includes {len(output['output'])} ndjson files:")
for file in output['output']:
    print(f" URL: {file['url']}")

print ('In an interactive demo, the JSON can be manipulated here:')
JSON(output)
```

```
Checking poll status URL https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/fhir/$export-poll-status?jobId=573101c0-8d07-4fa0-b0fd-b3e8fb536b16...
Export is complete.
Response: [
    {
        "type": "Immunization",
        "url": "https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/export/output/573101c0-8d07-4fa0-b0fd-b3e8fb536b16/Immunization_1.ndjson",
        "count": 2000
    },
    {
        "type": "ImmunizationEvaluation",
        "url": "https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/export/output/573101c0-8d07-4fa0-b0fd-b3e8fb536b16/ImmunizationEvaluation_1.ndjson",
        "count": 2000
    },
    {
        "type": "ImmunizationRecommendation",
        "url": "https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/export/output/573101c0-8d07-4fa0-b0fd-b3e8fb536b16/ImmunizationRecommendation_1.ndjson",
        "count": 16000
    },
    {
        "type": "Patient",
        "url": "https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/export/output/573101c0-8d07-4fa0-b0fd-b3e8fb536b16/Patient_1.ndjson",
        "count": 1000
    }
]
Output from FHIR server includes 4 ndjson files:
```

HLN CONSULTING

bulk-FHIR.ipynb

Download Git Binder Code Notebook No Kernel

```
]
Output from FHIR server includes 4 ndjson files:
 URL: https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/export/output/573101c0-8d07-4fa0-b0fd-b3e8fb536b16/Immunization_1.ndjson
 URL: https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/export/output/573101c0-8d07-4fa0-b0fd-b3e8fb536b16/ImmunizationEvaluation_1.ndjson
 URL: https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/export/output/573101c0-8d07-4fa0-b0fd-b3e8fb536b16/ImmunizationRecommendation_1.ndjson
 URL: https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/export/output/573101c0-8d07-4fa0-b0fd-b3e8fb536b16/Patient_1.ndjson
In an interactive demo, the JSON can be manipulated here:
```

[10]:

Find...

root
transactionTime "2025-03-04T14:28:30.820Z"
request "https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/fhir/Group/1102358/$export"
requiresAccessToken true
output [] 4 items
0
type "Immunization"
url "https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/export/output/573101c0-8d07-4fa0-b0fd-b3e8fb536b16/Immunization_1.ndjson"
count 2000
1
type "ImmunizationEvaluation"
url "https://ri-connectathon.open-iis.org/fhir-kidsnet-iz/v1/export/output/573101c0-8d07-4fa0-b0fd-b3e8fb536b16/ImmunizationEvaluation_1.ndjson"
count 2000
2
3
error [] 0 items

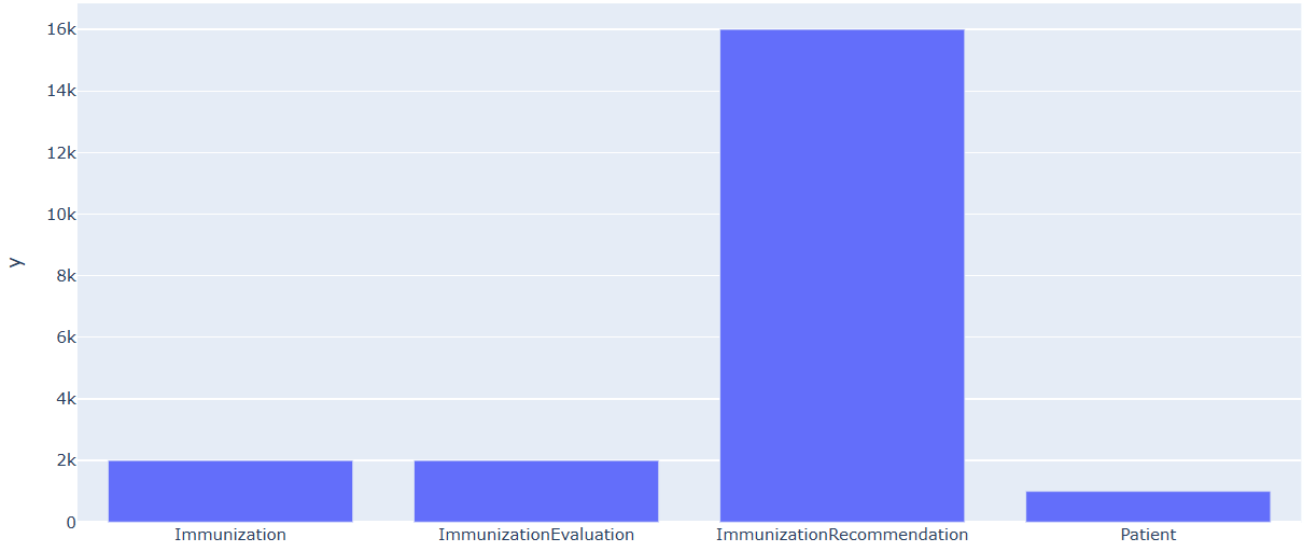
HLN CONSULTING

bulk-FHIR.ipynb

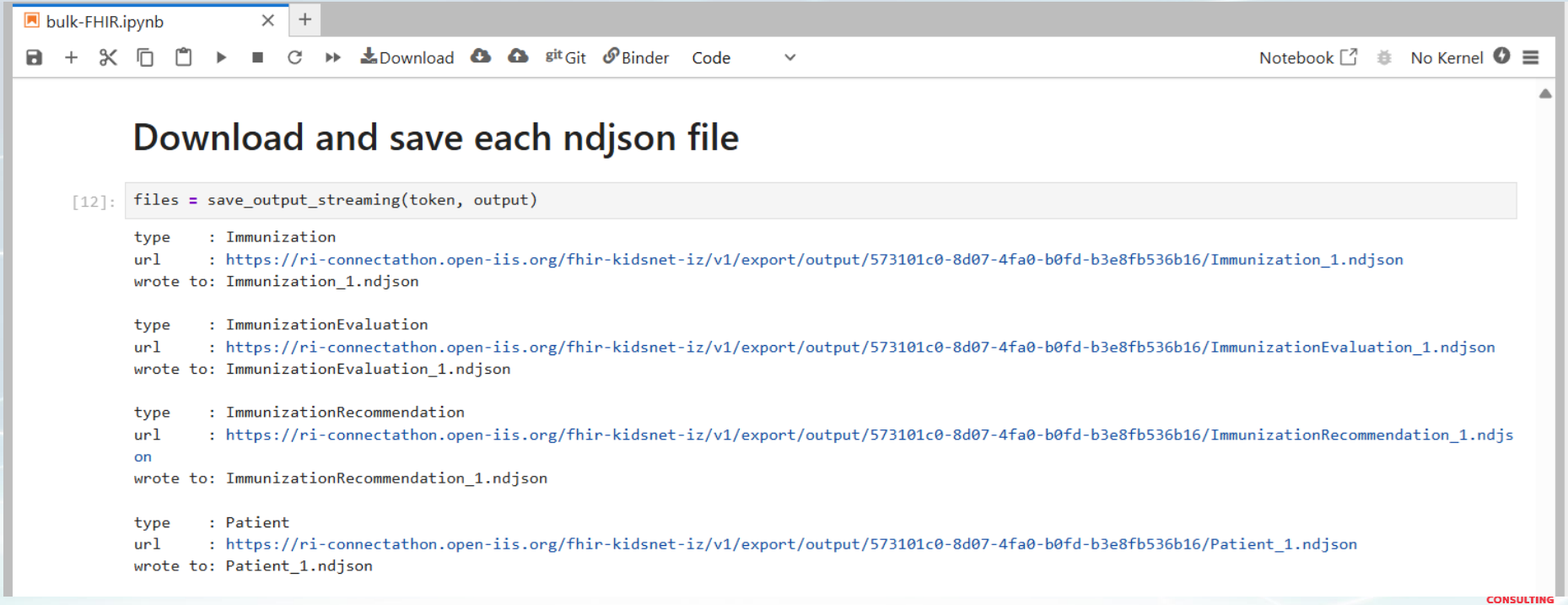
Download Git Binder Code Notebook No Kernel

Optionally, visualize the output in a bar chart

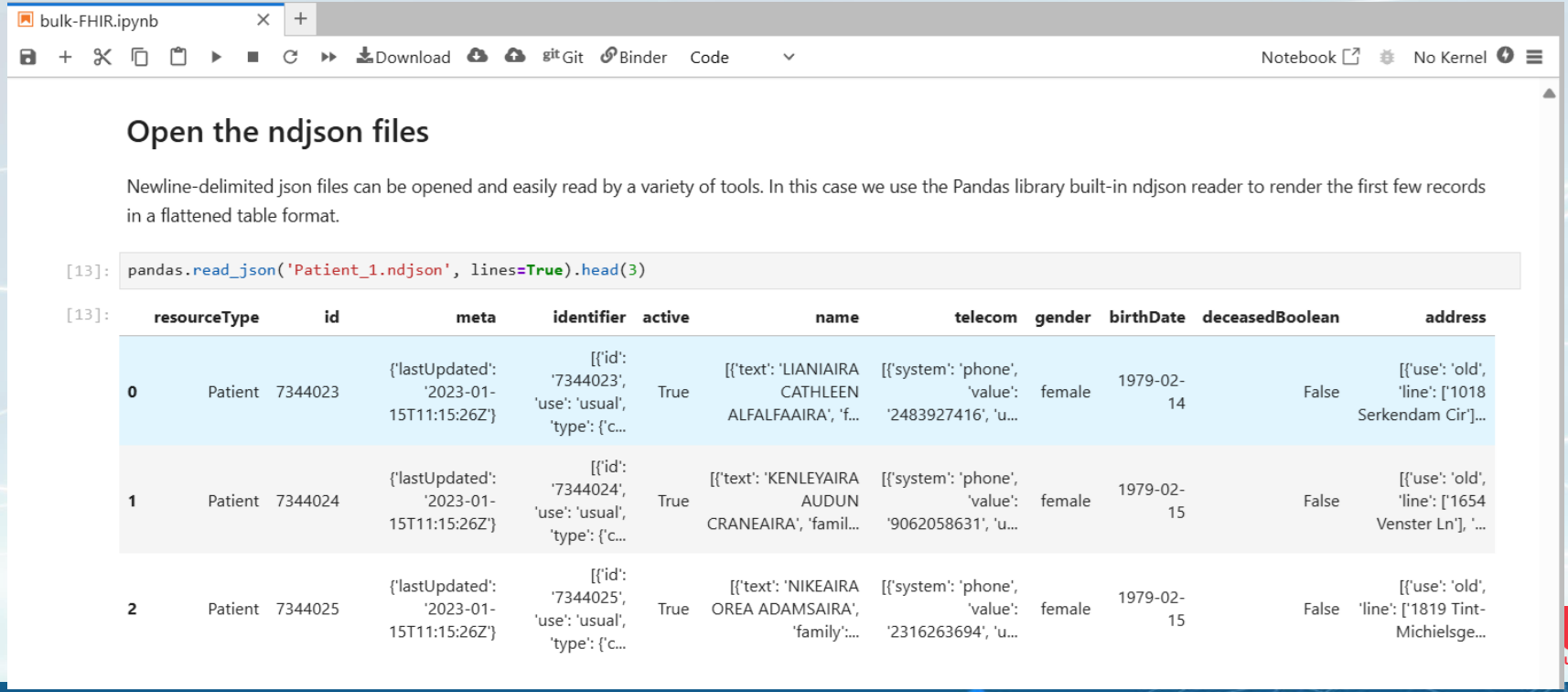
[11]:
```
#
# chart each FHIR resource with its number of records

resources = []
counts = []
for resource in output['output']:
    resources.append(resource['type'])
    counts.append(resource['count'])
fig = plotly.bar(x=resources, y=counts)
fig.show()
```



Demonstration screenshots



Demonstration screenshots



Open the ndjson files

Newline-delimited json files can be opened and easily read by a variety of tools. In this case we use the Pandas library built-in ndjson reader to render the first few records in a flattened table format.

[13]:
```
pandas.read_json('Patient_1.ndjson', lines=True).head(3)
```

[13]:

	resourceType	id	meta	identifier	active	name	telecom	gender	birthDate	deceasedBoolean	address
0	Patient	7344023	{'lastUpdated': '2023-01-15T11:15:26Z'}	[{'id': '7344023', 'use': 'usual', 'type': {'c...	True	[{'text': 'LIANIAIRA CATHLEEN ALFALFAAIRA', 'f...	[{'system': 'phone', 'value': '2483927416', 'u...	female	1979-02-14	False	[{'use': 'old', 'line': ['1018 Serkendam Cir']...
1	Patient	7344024	{'lastUpdated': '2023-01-15T11:15:26Z'}	[{'id': '7344024', 'use': 'usual', 'type': {'c...	True	[{'text': 'KENLEYAIRA AUDUN CRANEAIRA', 'famil...	[{'system': 'phone', 'value': '9062058631', 'u...	female	1979-02-15	False	[{'use': 'old', 'line': ['1654 Venster Ln'], '...
2	Patient	7344025	{'lastUpdated': '2023-01-15T11:15:26Z'}	[{'id': '7344025', 'use': 'usual', 'type': {'c...	True	[{'text': 'NIKEAIRA OREA ADAMSAIRA', 'family':...	[{'system': 'phone', 'value': '2316263694', 'u...	female	1979-02-15	False	[{'use': 'old', 'line': ['1819 Tint-Michielsge...

Demonstration screenshots

```
[14]: pandas.read_json('Immunization_1.ndjson', lines=True).head(3)
```

[14]:

	resourceType	id	meta	extension	identifier	status	vaccineCode	patient	occurrenceDateTime	recorde
0	Immunization	11159799	{'lastUpdated': '2023-01-14T11:43:09Z'}	[{'url': 'https://health.ri.gov/vaccine-catego...	[{'use': 'usual', 'value': '11159799'}, {'use'...	completed	{'coding': [{'system': 'http://hl7.org/fhir/si...	{'id': '7344023', 'reference': 'Patient/734402...	2021-01-19T00:00:00-04:00	2023-01-14T11:43:09
1	Immunization	11159800	{'lastUpdated': '2023-01-14T11:43:09Z'}	[{'url': 'https://health.ri.gov/vaccine-catego...	[{'use': 'usual', 'value': '11159800'}, {'use'...	completed	{'coding': [{'system': 'http://hl7.org/fhir/si...	{'id': '7344023', 'reference': 'Patient/734402...	2021-03-04T00:00:00-04:00	2023-01-14T11:43:09
2	Immunization	11159801	{'lastUpdated': '2023-01-14T11:43:09Z'}	[{'url': 'https://health.ri.gov/vaccine-catego...	[{'use': 'usual', 'value': '11159801'}, {'use'...	completed	{'coding': [{'system': 'http://hl7.org/fhir/si...	{'id': '7344024', 'reference': 'Patient/734402...	2021-01-19T00:00:00-04:00	2023-01-14T11:43:09

Demonstration screenshots

```
[15]: pandas.read_json('ImmunizationEvaluation_1.ndjson', lines=True).head(3)
```

[15]:

	resourceType	id	meta	status	patient	date	targetDisease	immunizationEvent	doseStatus
0	ImmunizationEvaluation	819120438	{'lastUpdated': '2023-01-15T11:15:26Z'}	completed	{'reference': 'Patient/7344024'}	2023-01-15 11:15:26+00:00	{'coding': [{'system': 'http://hl7.org/fhir/Va...	{'reference': 'Immunization/11159801', 'identi...	{'coding': [{'system' 'http://terminology.hl7..
1	ImmunizationEvaluation	819120439	{'lastUpdated': '2023-01-15T11:15:26Z'}	completed	{'reference': 'Patient/7344024'}	2023-01-15 11:15:26+00:00	{'coding': [{'system': 'http://hl7.org/fhir/Va...	{'reference': 'Immunization/11159802', 'identi...	{'coding': [{'system' 'http://terminology.hl7..
2	ImmunizationEvaluation	819120440	{'lastUpdated': '2023-01-15T11:15:26Z'}	completed	{'reference': 'Patient/7344025'}	2023-01-15 11:15:26+00:00	{'coding': [{'system': 'http://hl7.org/fhir/Va...	{'reference': 'Immunization/11159803', 'identi...	{'coding': [{'system' 'http://terminology.hl7..

HLN CONSULTING

Demonstration screenshots

```
[16]: pandas.read_json('ImmunizationRecommendation_1.ndjson', lines=True).head(3)
```

[16]:

	resourceType	id	meta	patient	date	recommendation
0	ImmunizationRecommendation	410205242	{'lastUpdated': '2023-01-15T11:15:26Z'}	{'id': '7344025', 'reference': 'Patient/734402...	2023-01-15 11:15:26+00:00	[{'vaccineCode': [{'coding': [{'system': 'urn:...
1	ImmunizationRecommendation	410205243	{'lastUpdated': '2023-01-15T11:15:26Z'}	{'id': '7344025', 'reference': 'Patient/734402...	2023-01-15 11:15:26+00:00	[{'vaccineCode': [{'coding': [{'system': 'urn:...
2	ImmunizationRecommendation	410205244	{'lastUpdated': '2023-01-15T11:15:26Z'}	{'id': '7344025', 'reference': 'Patient/734402...	2023-01-15 11:15:26+00:00	[{'vaccineCode': [{'coding': [{'system': 'urn:...

```
[ ]:
```