

FHIR client CRUDS

FHIR Client CRUDS examples CRUDS -> Create, Read, Update, Delete, Search examples

Python version in notebook -> 3.X

FHIR client used -> <https://github.com/smart-on-fhir/client-py> (<https://github.com/smart-on-fhir/client-py>)

examples partially adapted from [smart-on-fhir/client-py](https://github.com/smart-on-fhir/client-py) documentation

FHIR server used -> <http://test.fhir.org/r3> (<http://test.fhir.org/r3>) by Health Intersections, Grahame Grieve

```
In [51]: import json
from fhirclient import client

def pretty(js):
    """ pretty print a json object """
    return json.dumps(js, indent=2)

# Initialize client to access a given FHIR server

#settings = {
#    'app_id': 'my_web_app',
#    'api_base': 'https://fhir-open-api-dstu2.smarthealthit.org'
#}
settings = {
    'app_id': 'my_web_app',
    'api_base': 'http://test.fhir.org/r3'
}

smart = client.FHIRClient(settings=settings)
```

```
In [67]: import fhirclient.models.patient as pat
patient = pat.Patient.read('100', smart.server)
print("Patient name: " + smart.human_name(patient.name[0]))
print("Patient BirthDate: " + patient.birthDate.isostring)
print("-----")
print("Patient as a json object...")
print(pretty(patient.as_json()))
print("-----")
print("Patient as a text...")
print(patient.text.div)
```

```
Patient name: Gordon Garrett
Patient BirthDate: 1935-05-08
```

```
-----
Patient as a json object...
```

```
{
  "id": "100",
```

```

"meta": {
  "lastUpdated": "2017-11-28T08:33:16Z",
  "versionId": "1"
},
"text": {
  "div": "<div xmlns=\"http://www.w3.org/1999/xhtml\">Garrett, Gordon. MRN:\n
          577419</div>",
  "status": "generated"
},
"active": true,
"birthDate": "1935-05-08",
"gender": "male",
"identifier": [
  {
    "system": "https://github.com/projectcypress/cypress/patient",
    "type": {
      "coding": [
        {
          "code": "SS",
          "system": "http://hl7.org/fhir/v2/0203"
        }
      ]
    }
  },
  {
    "value": "577419"
  }
],
"managingOrganization": {
  "reference": "Organization/1"
},
"name": [
  {
    "family": "Garrett",
    "given": [
      "Gordon"
    ],
    "use": "official"
  }
],
"resourceType": "Patient"
}

```

Patient as a text...

```

<div xmlns="http://www.w3.org/1999/xhtml">Garrett, Gordon. MRN:
    577419</div>

```

```

In [53]: import fhirclient.models.procedure as proc
search = proc.Procedure.where(struct={'subject': 'Patient/319',
                                     'status': 'completed'})
procedures = search.perform_resources(smart.server)
for procedure in procedures:
    print(pretty(procedure.as_json()))

# to get the raw Bundle instead of resources only, you can use:
bundle = search.perform(smart.server)
print(pretty(bundle.as_json()))

{

```

```

    "id": "1",
    "meta": {
      "lastUpdated": "2017-12-01T18:41:01Z",
      "profile": [
        "http://hl7.org/fhir/us/core/StructureDefinition/us-core-procedure"
      ],
      "versionId": "1"
    },
    "code": {
      "coding": [
        {
          "code": "410123007",
          "display": "Medication regimen compliance education",
          "system": "http://snomed.info/sct"
        }
      ]
    },
    "identifier": [
      {
        "system": "urn:ietf:rfc:3986",
        "value": "urn:uuid:0bc79f35-1431-42b5-881a-974d1fd28430"
      }
    ],
    "performedPeriod": {
      "end": "2016-01-22",
      "start": "2016-01-22"
    },
    "status": "completed",
    "subject": {
      "reference": "Patient/319"
    },
    "resourceType": "Procedure"
  }
}
{
  "id": "497a1789-da28-46f1-bab0-b5ffa816b6",
  "meta": {
    "lastUpdated": "2018-02-18T08:14:05Z"
  },
  "entry": [
    {
      "fullUrl": "http://test.fhir.org/r3/Procedure/1",
      "resource": {
        "id": "1",
        "meta": {
          "lastUpdated": "2017-12-01T18:41:01Z",
          "profile": [
            "http://hl7.org/fhir/us/core/StructureDefinition/us-core-procedure"
          ],
          "versionId": "1"
        },
        "code": {
          "coding": [
            {
              "code": "410123007",
              "display": "Medication regimen compliance education",

```

```

        "system": "http://snomed.info/sct"
    }
]
},
"identifier": [
    {
        "system": "urn:ietf:rfc:3986",
        "value": "urn:uuid:0bc79f35-1431-42b5-881a-974d1fd28430"
    }
],
"performedPeriod": {
    "end": "2016-01-22",
    "start": "2016-01-22"
},
"status": "completed",
"subject": {
    "reference": "Patient/319"
},
"resourceType": "Procedure"
},
"search": {
    "mode": "match"
}
}
],
"link": [
    {
        "relation": "self",
        "url": "http://test.fhir.org/r3/Procedure?_format=application/fhir+json&search-id=78af07f8-79b4-4c66-8d06-948a4510cf&&subject=Patient/319&status=completed&_sort=_id"
    }
],
"total": 1,
"type": "searchset",
"resourceType": "Bundle"
}

```

```

In [68]: # Resource Creation
#
# Patient Creation
import fhirclient.models.humanname as hn

patient = pat.Patient()
name = hn.HumanName()

name.given = ['Peter2']
name.family = 'Parker2'
patient.name = [name]
print("Creating patient...")
print(pretty(patient.as_json()))
result = patient.create(smart.server)
print("-----")
print("Returned results...")
print(pretty(result))
print("-----")
print("Returned patient id: " + str(result["id"]))

```

```
#
# Get back Patient with the given id
patient = pat.Patient.read(result["id"], smart.server)
print("-----")
print("FHIR now returns this, if queried...")
print(pretty(patient.as_json()))
```

Creating patient...

```
{
  "name": [
    {
      "family": "Parker2",
      "given": [
        "Peter2"
      ]
    }
  ],
  "resourceType": "Patient"
}
```

Returned results...

```
{
  "resourceType": "Patient",
  "id": "726884",
  "meta": {
    "versionId": "1",
    "lastUpdated": "2018-02-18T09:50:19Z"
  },
  "name": [
    {
      "family": "Parker2",
      "given": [
        "Peter2"
      ]
    }
  ]
}
```

Returned patient id: 726884

FHIR now returns this, if queried...

```
{
  "id": "726884",
  "meta": {
    "lastUpdated": "2018-02-18T09:50:19Z",
    "versionId": "1"
  },
  "name": [
    {
      "family": "Parker2",
      "given": [
        "Peter2"
      ]
    }
  ],
  "resourceType": "Patient"
}
```

```
In [69]: # Resource Update
#
# Patient Update
name = hn.HumanName()
name.given = ['Peter3']
name.family = 'Parker3'
patient.name = [name]
print("Updating patient...")
print(pretty(patient.as_json()))
result = patient.update(smart.server)
print("-----")
print("Returned results...")
print(pretty(result))
```

Updating patient...

```
{
  "id": "726884",
  "meta": {
    "lastUpdated": "2018-02-18T09:50:19Z",
    "versionId": "1"
  },
  "name": [
    {
      "family": "Parker3",
      "given": [
        "Peter3"
      ]
    }
  ],
  "resourceType": "Patient"
}
```

Returned results...

```
{
  "resourceType": "Patient",
  "id": "726884",
  "meta": {
    "versionId": "2",
    "lastUpdated": "2018-02-18T09:51:58Z"
  },
  "name": [
    {
      "family": "Parker3",
      "given": [
        "Peter3"
      ]
    }
  ]
}
```

```
In [70]: # Resource Delete
#
# Patient Delete
print("Deleting patient...")
print(pretty(patient.as_json()))
result = patient.delete(smart.server)
print("-----")
print("Returned results...")
print(pretty(result))
```

Deleting patient...

```
{
  "id": "726884",
  "meta": {
    "lastUpdated": "2018-02-18T09:50:19Z",
    "versionId": "1"
  },
  "name": [
    {
      "family": "Parker3",
      "given": [
        "Peter3"
      ]
    }
  ],
  "resourceType": "Patient"
}
```

Returned results...
null

In []: