# 01a_onet

May 26, 2024

# 1 Load Data from O*NET

## 1.1 Content

1. Call API to get all Jobs
2. get details for each job
3. save to db

```
[6]: # imports
     import requests
     from requests.auth import HTTPBasicAuth
     import json
     from db import get_database
```

## Call API to get all Jobs

```
[7]: # import authentication data
     with open('./infos.json') as f:
         infos = json.load(f)
         onet = infos['onet']
         onetUsername = onet['username']
         onetPassword = onet['password']
```

```
[8]: # get all available jobs from onet
     occupationsUrl = "https://services.onetcenter.org/ws/online/occupations/"
     headers = {'Accept': 'application/json'}
     initialRequest = requests.get(occupationsUrl, auth=HTTPBasicAuth(onetUsername,␣
      ↪onetPassword), headers=headers)

     if initialRequest.status_code == 200:
         data = initialRequest.json()
         # the initial request only returns 20 jobs, so we need to make another␣
      ↪request to get all the jobs.
         # the initial request includes the total of available jobs
         total = data['total']
     else:
         print(f"Request failed with status code {initialRequest.status_code}")

     # get all jobs
```

```
getAllOccupations = requests.get(occupationsUrl,
  ↪auth=HTTPBasicAuth(onetUsername, onetPassword), headers=headers,
  ↪params={'start': 1, 'end': total})

if getAllOccupations.status_code == 200:
    data = getAllOccupations.json()
    occupations = data['occupation']
else:
    print(f"Request failed with status code {getAllOccupations.status_code}")
```

## Get details for each Job

```
[9]:  # Fetch details for each job
      summaries=[]

      for occupation in occupations:
          # without ?display=long onet only returns up to 5 items per category
          summaryUrl = f"https://services.onetcenter.org/ws/online/occupations/
        ↪{occupation['code']}/details?display=long"
          summaryRequest = requests.get(summaryUrl, auth=HTTPBasicAuth(onetUsername,
        ↪onetPassword), headers=headers)

          if summaryRequest.status_code == 200:
              occupation_data = summaryRequest.json()

              summaries.append(occupation_data)
          else:
              print(f"Request for occupation {occupation['code']} failed with status
        ↪code {summaryRequest.status_code}")
```

## Save to DB

```
[10]:  # save the data to the database
       dbname = get_database()

       collection_name = dbname["onet"]

       # Convert summaries to a list of dictionaries
       summaries_dict = [summary for summary in summaries]

       # Insert the dictionaries into the MongoDB collection
       collection_name.insert_many(summaries_dict)

       print("total jobs: ", len(summaries_dict))
```

```
total jobs:  1016
```