In [2]:
```
# Importing all the libraries

import time
import pandas as pd
import xml.etree.ElementTree as ET
```

In [9]:
```
# TASK 1.1
# Reading XML and converting to CSV

start_time = time.time()

tree = ET.parse("data/US_XML_AddFeed_20100101_20100107.xml")
root = tree.getroot()

get_range = lambda col: range(len(col))
l = [{r[i].tag:r[i].text for i in get_range(r)} for r in root]

df = pd.DataFrame.from_dict(l)

# Exporting to CSV file
df.to_csv("data/cleaned_file.csv")

end_time = time.time()
time_taken = end_time - start_time

print(f"It took the file {time_taken} seconds to convert from XML t

print(f"There are {len(df.axes[0])} rows and {len(df.axes[1])} colu
```

```
It took the file 5.913424015045166 seconds to convert from XML to
CSV
There are 23422 rows and 57 columns
```

In [21]:
```
# TASK 1.2
# Missing values

print(list(df.columns))

# Ratio of missing for selected columns / variables
print(df['ConsolidatedONET'].isnull().sum() / len(df.axes[0]) * 100
print(df['ConsolidatedInferredNAICS'].isnull().sum() / len(df.axes[
```

```
['JobID', 'CleanJobTitle', 'JobDomain', 'CanonCity', 'CanonCountry
', 'CanonState', 'JobDate', 'JobText', 'JobURL', 'PostingHTML', 'S
ource', 'JobReferenceID', 'Email', 'CanonEmployer', 'Latitude', 'L
ongitude', 'CanonIntermediary', 'Telephone', 'CanonJobTitle', 'Can
onCounty', 'DivisionCode', 'MSA', 'LMA', 'InternshipFlag', 'Consol
idatedONET', 'CanonCertification', 'CanonSkillClusters', 'CanonSki
lls', 'IsDuplicate', 'IsDuplicateOf', 'CanonMaximumDegree', 'Canon
MinimumDegree', 'CanonOtherDegrees', 'CanonPreferredDegrees', 'Can
onRequiredDegrees', 'CIPCode', 'StandardMajor', 'MaxExperience', '
MinExperience', 'ConsolidatedInferredNAICS', 'BGTOcc', 'MaxAnnualS
alary', 'MaxHourlySalary', 'MinAnnualSalary', 'MinHourlySalary', '
YearsOfExperience', 'CanonJobHours', 'CanonJobType', 'CanonPostalC
ode', 'CanonYearsOfExperienceCanonLevel', 'CanonYearsOfExperienceL
evel', 'ConsolidatedTitle', 'Language', 'BGTSubOcc', 'Consolidated
DegreeLevels', 'MaxDegreeLevel', 'MinDegreeLevel']
2.506190760823158
16.66382033985142
```

In [28]:
```
# TASK 2.2
# Top 5 occupations that are in demand
df.groupby(['ConsolidatedONET'])['ConsolidatedONET'].count().reset_
```

Out[28]:

	ConsolidatedONET	count
419	41401200	1938
304	29114100	844
408	41203100	730
403	41101100	590
89	15113200	582

In [ ]: