```
# -*- coding: utf-8 -*-
"""Capstone_Project_Census_Data.ipynb

Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/1ibym0wwvvB4hpH8Fw9dkkKjxhBXUBAka
"""

import pandas as pd

df=pd.read_csv('/content/census_2011-1.csv')
df.head()

df.info()

"""**TASK1:Renaming Column Names**"""

df.rename(columns={'State name':'State/UT','District
name':'District','Male_Literate':'Literate_Male','Female_Literate':'Literate_Female','Rur
al_Households':'Households_Rural','Urban_Households':'Households_Urban','Age_Group_0_29':
'Young_and_Adult','Age_Group_30_49':'Middle_Aged','Age_Group_50':'Senior_Citizen','Age
not stated':'Age_Not_Stated'},inplace=True)

df

"""**TASK2 : Renaming State/UT**"""

def capitalize_words_except_and(text):
  words=text.split()
  capitalized_words=[word.title() if word.lower() != 'and' else word.lower() for word in
words]
  return ' '.join(capitalized_words)

df['State/UT']=df['State/UT'].apply(capitalize_words_except_and)

df

"""**TASK3:New State/UT**"""

telangana_district=['Adilabad','Nizamabad','Warangal','Hyderabad','Medak','Karimnagar','R
angareddy','Mahbubnagar','Nalgonda','Khammam']
df.loc[df['District'].isin(telangana_district),'State/UT']='Telangana'

df[df['District']=='Hyderabad']

ladakh_district=['Leh(Ladakh)','Kargil']
df.loc[df['District'].isin(ladakh_district),'State/UT']='Ladakh'

df.query(" `State/UT` =='Telangana'")

df.query(" `State/UT` =='Ladakh'")

df

df.isnull().sum()

for i in df.columns:
  print(i)

df.isnull().sum()

df.head()

"""**missing data before processing**"""
```

```
missing_percentage_before = (df.isnull().sum()/len(df)) * 100
print(missing_percentage_before)

"""**Task 4: Processing Missing Data**"""

df['Population'].fillna(df['Young_and_Adult']+df['Middle_Aged']+df['Senior_Citizen']+df['
Age_Not_Stated'],inplace=True)
df['Middle_Aged'].fillna(df['Population']-
(df['Young_and_Adult']+df['Senior_Citizen']+df['Age_Not_Stated']),inplace=True)
df['Senior_Citizen'].fillna(df['Population']-
(df['Young_and_Adult']+df['Middle_Aged']+df['Age_Not_Stated']),inplace=True)
df['Age_Not_Stated'].fillna(df['Population']-
(df['Young_and_Adult']+df['Middle_Aged']+df['Senior_Citizen']),inplace=True)
df['Young_and_Adult'].fillna(df['Population']-
(df['Middle_Aged']+df['Senior_Citizen']+df['Age_Not_Stated']),inplace=True)
df['Households'].fillna(df['Households_Urban']+df['Households_Rural'],inplace=True)
df['Literate'].fillna(df['Literate_Male']+df['Literate_Female'],inplace=True)

df['Male'].fillna(df['Population']-df['Female'],inplace=True)

df['Female'].fillna(df['Population']-df['Male'],inplace=True)

df

"""**missing data after processing**"""

missing_percentage_after = (df.isnull().sum()/len(df)) * 100
print(missing_percentage_after)

df

"""**Task 5 Saving data to MongoDB**"""

pip install pymongo

import pymongo
client=pymongo.MongoClient("mongodb+srv://Meher_123:Meher_123@cluster0.1uwyev4.mongodb.ne
t/",27017)
db=client.Census_Data
collection=db.census

"""**converting columns into lower case for storing in MongoDB**"""

df
df.columns = df.columns.str.lower()

df

""" **Conversion of tabular data into key value pairs**"""

df_dict=df.to_dict("records") # for storing tabular data in key-value pairs
df_dict

"""**pushing the data into MongoDB**"""

collection.insert_many(df_dict)

"""**Fetching the collection data**"""

for i in collection.find({},{"_id":0}):
    print(i)

data=[]
for i in collection.find({},{"_id":0}):
  data.append(i)
```

```
data

"""**Fetching data from MongoDB**"""

df=pd.DataFrame(data)
df

"""**Task 6: Uploading the data into relational db**"""

pip install sqlalchemy==1.4.16

import sqlalchemy

#sqlalchemy.create_engine("SQL_NAME://USER_NAME:password@EndPoint:port_no/db")

engine=sqlalchemy.create_engine("postgresql://postgres:rootpassword@database-
3.cp6cus640rvj.ap-south-1.rds.amazonaws.com:5432/postgres")

df.to_sql(name="census_data",con=engine,if_exists="replace")
df.columns = df.columns.str.lower()

df
```