In [ ]:
```
import numpy as np
import pandas as pd
from datetime import datetime
import datetime
import time
import openpyxl
!pip install xlsxwriter
import xlsxwriter
import string
import sys
```

In [ ]:
```
source = 'https://docs.google.com/spreadsheets/d/e/**********************************************/pub?output=csv'
#Link encoded due to policies
```

Read and clean data

In [ ]:
```
df = pd.read_csv(source)
df.columns = ['Thoigian', 'lienlac','email', 'khuvuc', 'cho', 'ban', 'danhgia', 'gopy','ten', 'hinhanh', 'hinhanh2', 'hinhanh1']
df = df.dropna(how='all')
df = df.reset_index(drop=True)
df = df.fillna(value='')
def lam_sach_list(chuoi):
  list1 = chuoi.split('~')
  list2 = []
  count = 0
  for item in list1:
    item1 = item.strip()
    list2.append(item1)
  if len(list2[-1])<6 or list2[-1][0:5] != 'https://':
    for i in range(0, len(chuoi)):
      guard = chuoi[i:i+8]
      if guard == 'https://':
        break
      letter = chuoi[i]
      if letter == '~':
        count+=1
    list1 = chuoi.split('~', count)
    list2 = []
    for item in list1:
      item1 = item.strip()
      list2.append(item1)
  list2 = list(filter(None, list2))
  return list2
def lam_sach_list2(chuoi):
  list1 = []
  for item in chuoi.split('~'):
    item = item.strip()
    list1.append(item)
  return list1

df
#Output cannot be viewed due to policies
```

In [ ]:
```
source = pd.read_csv(source)
if source.shape[0] == 0:
  limit_row = 0
  limit_code = -1
else:
  limit_code = source['code'].iloc[source.shape[0]-1]
  limit_row = source.shape[0]

limit_row
#Output cannot be viewed due to policies
```

Parse data to their correct categories

In [ ]:
```
cho=[]
ban=[]
khuvuc=[]
lienlac=[]
ten=[]
hinhanh = []
for i in range(0, df.shape[0]):
    cho.append(list(filter(None, df['cho'].iloc[i].title().split('\n'))))
    ban.append(list(filter(None, df['ban'].iloc[i].title().split('\n'))))
    lienlac.append(df['lienlac'].iloc[i])
    khuvuc.append(df['khuvuc'].iloc[i])
    ten.append(df['ten'].iloc[i])
    if len(df['hinhanh']).iloc[i])!=0:
      hinhanh.append(df['hinhanh'].iloc[i])
    else:
      hinhanh.append('**************') #Encoded string
    lienlac = list(filter(None, lienlac))
    khuvuc = list(filter(None, khuvuc))
    ten = list(filter(None, ten))
    hinhanh = list(filter(None, hinhanh))

cho
#Output cannot be viewed due to policies
```

In [ ]:
```
name = dict()
image = dict()
contact = dict()
loce = dict()
sach=ban
list_abc={}
for i in range(0, len(sach)):
    if sach[i] == ['']:
      sach[i] = []
    for item in cho[i]:
      list_abc = lam_sach_list2(item)
      for m in range(0, len(list_abc)):
          list_abc[m] = str(list_abc[m]).strip()
      sach[i].append(list_abc)
    else:
      sach_item=[]
      for item in cho[i]:
        if cho[i] != ['']:
          sach[i].append(item)
      for n in range(0, len(sach[i])):
        list_item1 = lam_sach_list2(sach[i][n])
        if len(list(filter(None, list_item1))) != 0:
          for k in range(0, len(list_item1)):
              list_item1[k] = str(list_item1[k]).strip()
          sach[i][n] = list_item1
i=0
for item in ten:
    name[i] = item
    i+=1
i=0
for item in lienlac:
    contact[i] = item
    i+=1
i=0
for item in hinhanh:
    image[i] = item
    i+=1
i=0
for item in khuvuc:
    loce[i] = item
    i+=1

sach
#Output cannot be viewed due to policies
```

In [ ]:
```
tieude=dict()
tagia=dict()
mota=dict()
giaban=dict()
giabia=dict()
for i in range(0, len(sach)):
  tieude[i]=[]
  tagia[i]=[]
  mota[i]=[]
  giaban[i]=[]
  giabia[i]=[]
  sach[i] = list(filter(None, sach[i]))
  for n in range(0,len(sach[i])):
      tieude[i].append(sach[i][n][0])
      tagia[i].append(sach[i][n][1])
      mota[i].append(sach[i][n][2])
      if len(sach[i][n])>5:
        if (sach[i][n][3]).isdigit() and (sach[i][n][4]).isdigit():
          giabia[i].append("{:,}".format(int(sach[i][n][3]))).replace(',','.'))
          giaban[i].append("{:,}".format(int(sach[i][n][4]))).replace(',','.'))
        else:
          giabia[i].append(sach[i][n][3])
          giaban[i].append(sach[i][n][4])
      else:
          giabia[i].append('không có')
          giaban[i].append('không có')

tieude
#Output cannot be viewed due to policies
```

Manage files' directories (Google Drive)

In [ ]:
```
hinhanh_dict = dict()
for i in range(0, df.shape[0]):
  if len(hinhanh[i]) != 0:
    hinhanh_dict[i] = hinhanh[i].split(', ')
  else:
    hinhanh_dict[i] = []
```

In [ ]:
```
def get_file_id(link):
  target = ''
  for i in range(0, 33):
    letter = link[i]
    target += letter
  if target == 'https://drive.google.com/open?id=':
    fileid = link[i+1:]
  else:
    fileid = ''
  return fileid
```

In [ ]:
```
fileid = dict()
for i in range(0, len(hinhanh_dict)):
  fileid[i] = []
  for link in hinhanh_dict[i]:
    if len(hinhanh_dict[i]) != 0:
      fileid[i].append(get_file_id(link))
```

In [ ]:
```
from pydrive.auth import GoogleAuth
from pydrive.drive import GoogleDrive
from google.colab import auth, files
from oauth2client.client import GoogleCredentials
```

In [ ]:
```
auth.authenticate_user()
gauth = GoogleAuth()
gauth.credentials = GoogleCredentials.get_application_default()
drive = GoogleDrive(gauth)
```

In [ ]:
```
def createnewfolder(name, parents):
  folder_metadata = {'title' : name, 'mimeType' : 'application/vnd.google-apps.folder', 'parents' : [{'id': parents}]}
  folder = drive.CreateFile(folder_metadata)
  folder.Upload()
  return folder['id']
def movefiletofolder(file_id, new_parent):
  files = drive.auth.service.files()
  file  = files.get(fileId= file_id, fields='parents').execute()
  prev_parents = ','.join(p['id'] for p in file.get('parents'))
  file  = files.update(fileId = file_id,
                       addParents = new_parent,
                       removeParents = prev_parents,
                       fields = 'id, parents').execute()
```

In [ ]:
```
folder_code = dict()

for i in range(limit_code+1, df.shape[0]):
  folderid = createnewfolder(name=df['ten'].iloc[i], parents='**************************') #Encoded string
  folder_code[i] = folderid
  for code in fileid[i]:
    if len(fileid[i][0]) != 0:
      movefiletofolder(file_id=code, new_parent=folderid)
```

In [ ]:
```
folder_link = dict()
default_string = 'https://drive.google.com/open?id='
for i in folder_code.keys():
  folder_link[i] = default_string + folder_code[i]

folder_link
#Output cannot be viewed due to policies
```

Construct final dataframe

In [ ]:
```
codelist=[]
for n in tieude.keys():
    for i in range(0, len(tieude[n])):
        codelist.append(n)
s = pd.Series(codelist)
```

In [ ]:
```
excelsf=pd.DataFrame()
excelsf['code'] = s
```

In [ ]:
```
m=0
s = pd.Series(codelist)
for i in list(set(codelist)):
    tua = tieude[i]
    for n in range(0, len(tua)):
        s[m] = tua[n]
        m+=1
excelsf['Sách'] = s
m=0
s = pd.Series(codelist)
for i in tagia.keys():
    tua = tagia[i]
    for n in range(0, len(tua)):
        s[m] = tua[n]
        m+=1
excelsf['Tác giả'] = s
m=0
s = pd.Series(codelist)
for i in mota.keys():
    tua = mota[i]
    for n in range(0, len(tua)):
        s[m] = tua[n]
        m+=1
excelsf['Mô tả'] = s
m=0
s = pd.Series(codelist)
for m in range(limit_row, excelsf.shape[0]):
    s[m] = folder_link[excelsf['code'].iloc[m]]
excelsf['Hình ảnh'] = s
m=0
s = pd.Series(codelist)
for i in giabia.keys():
    gia = giabia[i]
    for n in range(0, len(gia)):
        s[m] = gia[n]
        m+=1
excelsf['Giá bìa'] = s
m=0
s = pd.Series(codelist)
for i in giaban.keys():
    gia = giaban[i]
    for n in range(0, len(gia)):
        s[m] = gia[n]
        m+=1
excelsf['Giá bán'] = s
s = pd.Series(codelist)
for m in range(0, excelsf.shape[0]):
    s[m] = name[excelsf['code'].iloc[m]]
excelsf['Tên'] = s
s = pd.Series(codelist)
for m in range(0, excelsf.shape[0]):
    s[m] = contact[excelsf['code'].iloc[m]]
excelsf['Thông tin liên lạc'] = s
s = pd.Series(codelist)
for m in range(0, excelsf.shape[0]):
    s[m] = loce[excelsf['code'].iloc[m]]
excelsf['Khu vực'] = s

excelsf.head(10)
#Output cannot be viewed due to policies
```

Update new data to spreadsheet

In [ ]:
```
from google.colab import auth
auth.authenticate_user()

import gspread
from oauth2client.client import GoogleCredentials
```

In [ ]:
```
gc = gspread.authorize(GoogleCredentials.get_application_default())
```

In [ ]:
```
worksheet = gc.open('BẢNG TRA CỨU chatbot').sheet1
for row in range(limit_row, excelsf.shape[0]):
  for col in range(0, len(excelsf.columns)):
    data = excelsf[excelsf.columns[col]].iloc[row]
    if type(data) == np.int64:
      data = int(data)
    worksheet.update_cell(row=2, col=1, data)
```

In [ ]:
```
worksheet = gc.open('BẢNG TRA CỨU').sheet1
for row in range(limit_row, excelsf.shape[0]):
  for col in range(1, len(list(excelsf.columns))):
    data = excelsf[excelsf.columns[col]].iloc[row]
    if type(data) == np.int64:
      data = int(data)
    worksheet.update_cell(row=2, col, data)
```