```python
In [1]: import pandas as pd
        import numpy as np
        import os
        import glob
```

```python
In [2]: path = "./NA"
        files = [file for file in os.listdir(path) if not file.startswith('.')]
        # Ignore hidden files
        SL44_NA_TSM = pd.DataFrame()
        for file in files:
            current_data = pd.read_excel(path+"/"+file)
            SL44_NA_TSM = pd.concat([SL44_NA_TSM, current_data])
```

```python
In [3]: path = "./NA"
        files = [file for file in os.listdir(path) if not file.startswith('.')]
        # Ignore hidden files
        SL44_NA_AVAMAR = pd.DataFrame()
        for file in files:
            current_data = pd.read_excel(path+"/"+file,1)
            SL44_NA_AVAMAR = pd.concat([SL44_NA_AVAMAR, current_data])
```

```python
In [9]: SL44_NA_AVAMAR.shape
```

```
Out[9]: (1430, 9)
```

```python
In [10]: SL44_NA_AVAMAR.columns
```

```
Out[10]: Index(['Server Name', 'SLA Date', 'Actual Start Time', 'Client Name',
               'Backup Status', 'Backup Rerun (Y/N)', 'Incident', 'Failures',
               'Backup Final Result'],
              dtype='object')
```

```python
In [7]: SL44_NA_TSM.drop(['SiteCode','ScheduleName','Duration(HH:MM:SS)','DataT
        ransfered(MB)','Performance(B/Sec)'],axis=1,inplace=True)
```

```
In [8]: SL44_NA_AVAMAR.drop(['Site_Code','Backup Type'],axis=1,inplace=True)
```

```
In [11]: SL44_NA_AVAMAR.rename(columns={'Actual Start Time':'StartDate','Server
          Name':'BackupServer', 'SLA Date':'BackupDay', 'Backup Rerun (Y/N)': 'B
         ackup re-run(Y/N)','Failures':'Reason for the Backup Failures','Backup
          Final Result':'Backup Final Outcome','Client Name':'ClientName', 'Back
         up Status':'BackupStatus',},inplace=True)
```

```
In [12]: SL44_NA_AVAMAR.columns
```

```
Out[12]: Index(['BackupServer', 'BackupDay', 'StartDate', 'ClientName', 'BackupS
         tatus',
                'Backup re-run(Y/N)', 'Incident', 'Reason for the Backup Failure
         s',
                'Backup Final Outcome'],
               dtype='object')
```

```
In [13]: SL44_NA_TSM.shape
```

```
Out[13]: (4419, 9)
```

```
In [16]: set(SL44_NA_TSM.columns) - set(SL44_NA_AVAMAR.columns)
```

```
Out[16]: set()
```

```
In [15]: SL44_NA_TSM.rename(columns={' Backup re-run(Y/N)':'Backup re-run(Y/N)'
         },inplace=True)
```

```
In [17]: SL44_NA_TSM.head()
```

Out[17]:

	BackupServer	BackupDay	StartDate	ClientName	BackupStatus	Backup re-run(Y/N)	Incident	Reason for the Backup Failures

	BackupServer	BackupDay	StartDate	ClientName	BackupStatus	Backup re-run(Y/N)	Incident	Reason for the Backup Failures
0	CASIBAK003	2021-02-14 00:00:00	2021-02-13 00:00:00	CAPPVOC005	Completed	NaN	NaN	NaN
1	CASIBAK003	2021-02-14 00:00:00	2021-02-13 00:00:00	CASIMSM003	Completed	NaN	NaN	NaN
2	CASIBAK003	2021-02-14 00:00:00	2021-02-13 00:00:00	CASIMSS003	Completed	NaN	NaN	NaN
3	CASIBAK003	2021-02-14 00:00:00	2021-02-13 00:00:00	CASIBAK004	Completed	NaN	NaN	NaN
4	CASIBAK003	2021-02-14 00:00:00	2021-02-13 00:00:00	CAPDSMT016	Completed	NaN	NaN	NaN

```
In [18]: SL44_NA_FINAL = pd.concat([SL44_NA_AVAMAR,SL44_NA_TSM],ignore_index=True)
```

```
In [19]: SL44_NA_FINAL
```

Out[19]:

	BackupServer	BackupDay	StartDate	ClientName	BackupStatus	Backup re-run(Y/
0	AVAIAUN003.FDNET.COM	2021-02-14 00:00:00	Sat Feb 13 19:00:41 PST 2021	ffspdfp001.fdnet.com	Activity completed successfully.	Na

	BackupServer	BackupDay	StartDate	ClientName	BackupStatus	Backu... run(Y/
1	AVAIAUN003.FDNET.COM	2021-02-14 00:00:00	Sat Feb 13 19:00:38 PST 2021	avppapp070.fdnet.com	Activity completed successfully.	Na
2	AVAIAUN003.FDNET.COM	2021-02-14 00:00:00	Sat Feb 13 19:00:38 PST 2021	sfsimss002.fdnet.com	Activity completed successfully.	Na
3	AVAIAUN003.FDNET.COM	2021-02-14 00:00:00	Sat Feb 13 19:00:37 PST 2021	sfppdba003.fdnet.com	Activity completed successfully.	Na
4	AVAIAUN003.FDNET.COM	2021-02-14 00:00:00	Sat Feb 13 19:00:38 PST 2021	sfppapp002.fdnet.com	Activity completed successfully.	Na
...	...	...	...	...	...	
5844	GVSIBAK006	2021-02-09 00:00:00	2021-02-08 00:00:00	GVPTFIL011	Completed	Na
5845	GVSIBAK006	2021-02-09 00:00:00	2021-02-08 00:00:00	GVSPAUD001_NEW	Completed	Na
5846	GVSIBAK006	2021-02-09 00:00:00	2021-02-08 00:00:00	GVPDDBA048	Completed	Na
5847	GVSIBAK006	2021-02-09 00:00:00	2021-02-08 00:00:00	GVPPDBA047	Completed	Na
5848	GVSIBAK006	2021-02-09 00:00:00	2021-02-08 00:00:00	GVSPDBA050	Completed	Na

5849 rows × 9 columns

```
In [20]: SL44_NA_FINAL.to_excel("SL44_NA_FINAL.xlsx",index=False)

In [21]: path = "./SA"
         files = [file for file in os.listdir(path) if not file.startswith('.')]
         # Ignore hidden files
         SL44_SA_TSM = pd.DataFrame()
         for file in files:
             current_data = pd.read_excel(path+"/"+file)
             SL44_SA_TSM = pd.concat([SL44_SA_TSM, current_data])

In [22]: path = "./SA"
         files = [file for file in os.listdir(path) if not file.startswith('.')]
         # Ignore hidden files
         SL44_SA_AVAMAR = pd.DataFrame()
         for file in files:
             current_data = pd.read_excel(path+"/"+file,1)
             SL44_SA_AVAMAR = pd.concat([SL44_SA_AVAMAR, current_data])

In [24]: SL44_SA_TSM.drop(['ScheduleName','Duration(HH:MM:SS)','DataTransfered(M
         B)','Performance(B/Sec)'],axis=1,inplace=True)

In [30]: SL44_SA_TSM.columns

Out[30]: Index(['BackupServer', 'BackupDay', 'StartDate', 'ClientName', 'BackupS
         tatus',
                ' Backup re-run(Y/N)', 'Incident', 'Reason for the Backup Failur
         es',
                'Backup Final Outcome'],
               dtype='object')

In [29]: SL44_SA_TSM.shape

Out[29]: (1189, 9)

In [28]: SL44_SA_TSM.drop(['M', 'Site Code', 'Site_Code'],axis=1,inplace=True)
```

```
In [32]:   SL44_SA_AVAMAR.rename(columns={'Actual Start Time':'StartDate','Server
           Name':'BackupServer', 'SLA Date':'BackupDay', 'Backup Rerun (Y/N)': 'B
           ackup re-run(Y/N)','Failures':'Reason for the Backup Failures','Backup
           Final Result':'Backup Final Outcome','Client Name':'ClientName', 'Back
           up Status':'BackupStatus',},inplace=True)
```

```
In [33]:   SL44_SA_AVAMAR.drop(['Site_Code','Backup Type'],axis=1,inplace=True)
```

```
In [35]:   SL44_SA_AVAMAR.shape
```

```
Out[35]:   (379, 9)
```

```
In [36]:   SL44_SA_AVAMAR.columns
```

```
Out[36]:   Index(['BackupServer', 'BackupDay', 'StartDate', 'ClientName', 'BackupS
           tatus',
                   'Backup re-run(Y/N)', 'Incident', 'Reason for the Backup Failure
           s',
                   'Backup Final Outcome'],
                  dtype='object')
```

```
In [37]:   SL44_SA_TSM.shape
```

```
Out[37]:   (1189, 9)
```

```
In [38]:   set(SL44_SA_TSM.columns) - set(SL44_SA_AVAMAR.columns)
```

```
Out[38]:   {' Backup re-run(Y/N)'}
```

```
In [41]:   set(SL44_SA_TSM.columns) - set(SL44_SA_AVAMAR.columns)
```

```
Out[41]:   set()
```

```
In [40]:   SL44_SA_TSM.rename(columns={' Backup re-run(Y/N)':'Backup re-run(Y/N)'
           },inplace=True)
```

```
In [42]:  SL44_SA_FINAL = pd.concat([SL44_SA_AVAMAR,SL44_SA_TSM],ignore_index=Tru
          e)
```

```
In [43]:  SL44_SA_FINAL.shape
```

Out[43]: (1568, 9)

```
In [44]:  path = "./RG"
          files = [file for file in os.listdir(path) if not file.startswith('.')]
          # Ignore hidden files
          SL44_RG_TSM = pd.DataFrame()
          for file in files:
              current_data = pd.read_excel(path+"/"+file)
              SL44_RG_TSM = pd.concat([SL44_RG_TSM, current_data])
```

```
In [45]:  path = "./RG"
          files = [file for file in os.listdir(path) if not file.startswith('.')]
          # Ignore hidden files
          SL44_RG_AVAMAR = pd.DataFrame()
          for file in files:
              current_data = pd.read_excel(path+"/"+file,1)
              SL44_RG_AVAMAR = pd.concat([SL44_RG_AVAMAR, current_data])
```

```
In [46]:  SL44_RG_TSM.drop(['SiteCode','ScheduleName','Duration(HH:MM:SS)','DataT
          ransfered(MB)','Performance(B/Sec)'],axis=1,inplace=True)
```

```
In [47]:  SL44_RG_AVAMAR.rename(columns={'Actual Start Time':'StartDate','Server
           Name':'BackupServer', 'SLA Date':'BackupDay', 'Backup Rerun (Y/N)': 'B
          ackup re-run(Y/N)','Failures':'Reason for the Backup Failures','Backup
           Final Result':'Backup Final Outcome','Client Name':'ClientName', 'Back
          up Status':'BackupStatus',},inplace=True)
```

```
In [48]:  SL44_RG_AVAMAR.drop(['Site_Code','Backup Type'],axis=1,inplace=True)
```

```
In [49]:  SL44_RG_AVAMAR.shape
```

Out[49]: (945, 10)

```
In [56]:   SL44_RG_TSM.shape

Out[56]:   (17182, 9)

In [55]:   set(SL44_RG_TSM.columns) - set(SL44_RG_AVAMAR.columns)

Out[55]:   {'Backup\xa0Final\xa0Outcome',
            'Reason\xa0for\xa0the\xa0Backup\xa0Failures',
            '\xa0Backup\xa0re-run(Y/N)'}

In [52]:   SL44_RG_TSM.to_excel("SL44_RG_TSM.xlsx",index=False)

In [54]:   SL44_RG_TSM = pd.read_excel("SL44_RG_TSM.xlsx")

In [57]:   SL44_RG_TSM.rename(columns={' Backup re-run(Y/N)':'Backup re-run(Y/N)'
           },inplace=True)

In [58]:   SL44_RG_FINAL = pd.concat([SL44_RG_AVAMAR,SL44_RG_TSM],ignore_index=Tru
           e)

In [59]:   SL44_RG_FINAL.shape

Out[59]:   (18127, 13)

In [60]:   SL44_RG_FINAL.columns

Out[60]:   Index(['BackupServer', 'BackupDay', 'StartDate', 'ClientName', 'BackupS
           tatus',
                  'Backup re-run(Y/N)', 'Incident', 'Reason for the Backup Failure
           s',
                  'Backup Final Outcome', ' ', ' Backup re-run(Y/N)',
                  'Reason for the Backup Failures', 'Backup Final Outcome'],
                 dtype='object')

In [62]:   SL44_RG_TSM = pd.read_excel("SL44_RG_TSM.xlsx")
```

In [63]: SL44_RG_TSM

Out[63]:

	BackupServer	BackupDay	StartDate	ClientName	BackupStatus	Backup re-run(Y/N)	Inci
0	RGSIBAK004	2021-02-02 00:00:00	2021-02-01 18:00:00	RGPDBDA126	Completed	NaN	
1	RGSIBAK004	2021-02-02 00:00:00	2021-02-01 18:00:05	RGPQBDA224	Completed	NaN	
2	RGSIBAK004	2021-02-02 00:00:00	2021-02-01 18:00:02	RGPDBDA125	Completed	NaN	
3	RGSIBAK004	2021-02-02 00:00:00	2021-02-01 18:00:53	RGPPCCS001	Completed	NaN	
4	RGSIBAK004	2021-02-02 00:00:00	2021-02-01 18:00:45	RGPISKL001	Completed	NaN	
...	...	...	...	...	...	...	
17177	RGSIBAK006	2021-02-09 00:00:00	2021-02-08 21:30:09	RGPPDOM008_NEW	Completed	NaN	
17178	RGSIBAK006	2021-02-09 00:00:00	2021-02-08 23:30:57	RGPQDBA168	Completed	NaN	
17179	RGSIBAK006	2021-02-09 00:00:00	2021-02-08 22:00:00	RGBPLNM126	Missed	Yes	
17180	RGSIBAK006	2021-02-09 00:00:00	2021-02-08 19:00:00	RGPQMDM203	Missed	Yes	
17181	RGSIBAK006	2021-02-09 00:00:00	2021-02-08 19:00:00	RGSIBAK007	Missed	No	

17182 rows × 9 columns

In [64]:
```python
SL44_RG_FINAL.to_excel("SL44_RG_FINAL.xlsx",index=False)
```

In [66]:
```python
SL44_RG_FINAL = pd.read_excel("SL44_RG_FINAL.xlsx")
```

In [67]:
```python
SL44_RG_FINAL.shape
```

Out[67]: (18127, 9)

In [68]:
```python
path = "./APAC"
files = [file for file in os.listdir(path) if not file.startswith('.')]
# Ignore hidden files
SL44_APAC_TSM = pd.DataFrame()
for file in files:
    current_data = pd.read_excel(path+"/"+file)
    SL44_APAC_TSM = pd.concat([SL44_APAC_TSM, current_data])
```

In [69]:
```python
path = "./APAC"
files = [file for file in os.listdir(path) if not file.startswith('.')]
# Ignore hidden files
SL44_APAC_AVAMAR = pd.DataFrame()
for file in files:
    current_data = pd.read_excel(path+"/"+file,1)
    SL44_APAC_AVAMAR = pd.concat([SL44_APAC_AVAMAR, current_data])
```

In [70]:
```python
SL44_APAC_TSM.drop(['SiteCode','ScheduleName','Duration(HH:MM:SS)','DataTransfered(MB)','Performance(B/Sec)'],axis=1,inplace=True)
```

In [71]:
```python
SL44_APAC_AVAMAR.rename(columns={'Actual Start Time':'StartDate','Server Name':'BackupServer', 'SLA Date':'BackupDay', 'Backup Rerun (Y/N)':'Backup re-run(Y/N)','Failures':'Reason for the Backup Failures','Backup Final Result':'Backup Final Outcome','Client Name':'ClientName', 'Backup Status':'BackupStatus',},inplace=True)
```

```
In [72]: SL44_APAC_AVAMAR.drop(['Site_Code','Backup Type'],axis=1,inplace=True)
```

```
In [76]: SL44_APAC_AVAMAR.shape
```
Out[76]: (2912, 9)

```
In [75]: SL44_APAC_TSM.shape
```
Out[75]: (3860, 10)

```
In [80]: set(SL44_APAC_TSM.columns) - set(SL44_APAC_AVAMAR.columns)
```
Out[80]: set()

```
In [78]: SL44_APAC_TSM.drop(['SHSIBAK001'],axis=1,inplace=True)
```

```
In [79]: SL44_APAC_TSM.rename(columns={' Backup re-run(Y/N)':'Backup re-run(Y/
         N)'},inplace=True)
```

```
In [81]: SL44_APAC_FINAL = pd.concat([SL44_APAC_AVAMAR,SL44_APAC_TSM],ignore_ind
         ex=True)
```

```
In [82]: SL44_APAC_FINAL.shape
```
Out[82]: (6772, 9)

```
In [83]: SL44_IND_FINAL = pd.concat([SL44_SA_FINAL,SL44_NA_FINAL,SL44_APAC_FINAL
         ,SL44_RG_FINAL],ignore_index=True)
```

```
In [84]: SL44_IND_FINAL.to_excel("SL44_IND_FINAL.xlsx",index=False)
```

```
In [85]: SL44_IND_FINAL.shape
```
Out[85]: (32316, 9)

```
In [86]: path = "./EMEA"
         files = [file for file in os.listdir(path) if not file.startswith('.')]
         # Ignore hidden files
         SL44_EMEA_TSM = pd.DataFrame()
         for file in files:
             current_data = pd.read_excel(path+"/"+file)
             SL44_EMEA_TSM = pd.concat([SL44_EMEA_TSM, current_data])
```

```
In [87]: SL44_EMEA_TSM.shape
```

```
Out[87]: (1906, 14)
```

```
In [88]: path = "./EMEA"
         files = [file for file in os.listdir(path) if not file.startswith('.')]
         # Ignore hidden files
         SL44_EMEA_AVAMAR = pd.DataFrame()
         for file in files:
             current_data = pd.read_excel(path+"/"+file,1)
             SL44_EMEA_AVAMAR = pd.concat([SL44_EMEA_AVAMAR, current_data])
```

```
In [89]: SL44_EMEA_AVAMAR.shape
```

```
Out[89]: (739, 14)
```

```
In [90]: SL44_EMEA_FINAL = pd.concat([SL44_EMEA_AVAMAR,SL44_EMEA_TSM],ignore_ind
         ex=True)
```

```
In [91]: SL44_EMEA_FINAL.drop(['Duration(HH:MM:SS)','DataTransferredKB','Perform
         ance(MB/sec)'],axis=1,inplace=True)
```

```
In [92]: SL44_EMEA_FINAL.drop(['SiteCode','ScheduleName'],axis=1,inplace=True)
```

```
In [93]: SL44_EMEA_FINAL.rename(columns={'Backup re-run':'Backup re-run(Y/N)','O
         utcome':'BackupStatus','Final outcome':'Backup Final Outcome','Reason':
         'Reason for the Backup Failures','StartDateTime':'StartDate'},inplace=T
         rue)
```

```
In [94]: set(SL44_EMEA_FINAL.columns) - set(SL44_IND_FINAL.columns)
```

Out[94]: {'Backup Final outcome', 'Reason for Backup Failures'}

```
In [95]: SL44_EMEA_FINAL.rename(columns={'Backup Final outcome':'Backup Final Ou
         tcome','Reason for Backup Failures':'Reason for the Backup Failures'},i
         nplace=True)
```

```
In [96]: set(SL44_EMEA_FINAL.columns) - set(SL44_IND_FINAL.columns)
```

Out[96]: set()

```
In [97]: path = "./EMEAN"
         files = [file for file in os.listdir(path) if not file.startswith('.')]
         # Ignore hidden files
         SL44_EMEAN_TSM = pd.DataFrame()
         for file in files:
             current_data = pd.read_excel(path+"/"+file)
             SL44_EMEAN_TSM = pd.concat([SL44_EMEAN_TSM, current_data])
```

```
In [98]: path = "./EMEAN"
         files = [file for file in os.listdir(path) if not file.startswith('.')]
         # Ignore hidden files
         SL44_EMEAN_AVAMAR = pd.DataFrame()
         for file in files:
             current_data = pd.read_excel(path+"/"+file,1)
             SL44_EMEAN_AVAMAR = pd.concat([SL44_EMEAN_AVAMAR, current_data])
```

```
In [99]: SL44_EMEAN_TSM.shape
```

Out[99]: (5997, 14)

```
In [100]: SL44_EMEAN_AVAMAR.shape
```

Out[100]: (1686, 14)

```
In [101]: SL44_EMEAN_FINAL = pd.concat([SL44_EMEAN_AVAMAR,SL44_EMEAN_TSM],ignore_
```

```
                    index=True)

In [102]:  SL44_EMEAN_FINAL.shape

Out[102]:  (7683, 14)

In [103]:  SL44_EMEAN_FINAL.drop(['Duration(HH:MM:SS)','DataTransferredKB','Perfor
           mance(MB/sec)'],axis=1,inplace=True)

In [104]:  SL44_EMEAN_FINAL.drop(['SiteCode','ScheduleName'],axis=1,inplace=True)

In [105]:  set(SL44_EMEAN_FINAL.columns) - set(SL44_IND_FINAL.columns)

Out[105]:  {'Backup Final outcome',
            'Backup re-run',
            'Reason for Backup Failures',
            'StartDateTime'}

In [106]:  SL44_EMEAN_FINAL.rename(columns={'Backup Final outcome':'Backup Final O
           utcome','Reason for Backup Failures':'Reason for the Backup Failures',
           'Backup re-run':'Backup re-run(Y/N)','StartDateTime':'StartDate'},inpla
           ce=True)

In [107]:  set(SL44_EMEAN_FINAL.columns) - set(SL44_IND_FINAL.columns)

Out[107]:  set()

In [108]:  SL44_FEB_FINAL = pd.concat([SL44_IND_FINAL,SL44_EMEA_FINAL,SL44_EMEAN_F
           INAL],ignore_index=True)

In [109]:  SL44_FEB_FINAL.shape

Out[109]:  (42644, 9)

In [110]:  SL44_FEB_FINAL.to_excel("SL44_FEB_FINAL.xlsx",index=False)
```

```python
In [111]: df = pd.read_excel('SL44_FEB_FINAL.xlsx')

In [112]: df.shape

Out[112]: (42320, 7)

In [113]: df.loc[df.BackupServer == 'ABSIMSM002.FDNET.COM','Region']='UAE'
          df.loc[df.BackupServer == 'assibak004.fdnet.com','Region']='SPAIN'
          df.loc[df.BackupServer == 'AVAIAUN003.FDNET.COM','Region']='USA'
          df.loc[df.BackupServer == 'assibak004.fdnet.com','Region']='SPAIN'
          df.loc[df.BackupServer == 'AVSIBAK007','Region']='USA'
          df.loc[df.BackupServer == 'BOPIAUN001.FDNET.COM','Region']='Chili'
          df.loc[df.BackupServer == 'BZSIBAK003.FDNET.COM','Region']='Netherland
          s'
          df.loc[df.BackupServer == 'CASIBAK003','Region']='Canada'
          df.loc[df.BackupServer == 'DAPIAUN001.FDNET.COM','Region']='USA'
          df.loc[df.BackupServer == 'DWPIAUN001.FDNET.COM','Region']='USA'
          df.loc[df.BackupServer == 'dxsimsm001.FDNET.COM','Region']='Kazakhstan'
          df.loc[df.BackupServer == 'DHSIMSS002.FDNET.COM','Region']='Nepal'
          df.loc[df.BackupServer == 'fbpiaun001','Region']='UK'
          df.loc[df.BackupServer == 'FBSIBAK003.FDNET.COM','Region']='UK'
          df.loc[df.BackupServer == 'fbsibak003.fdnet.com','Region']='UK'
          df.loc[df.BackupServer == 'FCPIAUN001.FDNET.COM','Region']='Phillipine
          s'
          df.loc[df.BackupServer == 'FCSIBAK002','Region']='Phillipines'
          df.loc[df.BackupServer == 'fnpiaun001','Region']='South Africa'
          df.loc[df.BackupServer == 'FNSIBAK002.fdnet.com','Region']='South Afric
          a'
          df.loc[df.BackupServer == 'FXPIAUN001.FDNET.COM','Region']='China'
          df.loc[df.BackupServer == 'glpiaun001','Region']='Poland'
          df.loc[df.BackupServer == 'glsibak003.fdnet.com','Region']='Poland'
          df.loc[df.BackupServer == 'GVAIAUN003.FDNET.COM','Region']='USA'
          df.loc[df.BackupServer == 'GVSIBAK005','Region']='USA'
          df.loc[df.BackupServer == 'GVSIBAK006','Region']='USA'
          df.loc[df.BackupServer == 'HNPIAUN001.FDNET.COM','Region']='Canada'
          df.loc[df.BackupServer == 'HOSIBAK005','Region']='USA'
          df.loc[df.BackupServer == 'HQPIAUN001.FDNET.COM','Region']='USA'
          df.loc[df.BackupServer == 'HYSIBAK002.FDNET.COM','Region']='UK'
          df.loc[df.BackupServer == 'INSIBAK001','Region']='India'
```

```python
df.loc[df.BackupServer == 'JKPIAUN001.FDNET.COM','Region']='USA'
df.loc[df.BackupServer == 'KCSIMSS002.FDNET.COM','Region']='Kuwait'
df.loc[df.BackupServer == 'KFSIMSS001','Region']='USA'
df.loc[df.BackupServer == 'kvpiaun001','Region']='Kazakhstan'
df.loc[df.BackupServer == 'kvsibak002.fdnet.com','Region']='Kazakhstan'
df.loc[df.BackupServer == 'LBSIBAK002','Region']='USA'
df.loc[df.BackupServer == 'LPSIBAK002','Region']='USA'
df.loc[df.BackupServer == 'LVPIAUN001.FDNET.COM','Region']='USA'
df.loc[df.BackupServer == 'MEPIAUN001.FDNET.COM','Region']='Australia'
df.loc[df.BackupServer == 'MFSIMSS003.FDNET.COM','Region']='Russia'
df.loc[df.BackupServer == 'MLSIBAK002','Region']='Phillipines'
df.loc[df.BackupServer == 'NBPIAUN001.FDNET.COM','Region']='USA'
df.loc[df.BackupServer == 'OVPIAUN001.FDNET.COM','Region']='Chili'
df.loc[df.BackupServer == 'PEPIAUN001.FDNET.COM','Region']='Australia'
df.loc[df.BackupServer == 'PESIBAK003','Region']='Australia'
df.loc[df.BackupServer == 'PESIBAK004','Region']='Australia'
df.loc[df.BackupServer == 'RGPIAUN003.FDNET.COM','Region']='USA'
df.loc[df.BackupServer == 'RGSIBAK006','Region']='USA'
df.loc[df.BackupServer == 'RGSIBAK004','Region']='USA'
df.loc[df.BackupServer == 'SASIBAK002.FDNET.COM','Region']='South Africa'
df.loc[df.BackupServer == 'SGPIAUN001.FDNET.COM','Region']='USA'
df.loc[df.BackupServer == 'SHSIBAK001','Region']='China'
df.loc[df.BackupServer == 'SJSIMSS002','Region']='USA'
df.loc[df.BackupServer == 'SNPIAUN001.FDNET.COM','Region']='Chili'
df.loc[df.BackupServer == 'SNSIBAK002','Region']='Chili'
df.loc[df.BackupServer == 'TGSIMSS001.FDNET.COM','Region']='Kuwait'
df.loc[df.BackupServer == 'tqspdfp001.fdnet.com','Region']='Saudi Arabia'
df.loc[df.BackupServer == 'TTPIAUN001.FDNET.COM','Region']='USA'
df.loc[df.BackupServer == 'ukpiaun001','Region']='South Africa'
df.loc[df.BackupServer == 'UKSIBAK002.FDNET.COM','Region']='South Africa'
df.loc[df.BackupServer == 'VASIBAK002','Region']='Canada'
df.loc[df.BackupServer == 'UNSIMSS001','Region']='USA'
df.loc[df.BackupServer == 'VCPIAUN001.FDNET.COM','Region']='Chili'
df.loc[df.BackupServer == 'VTSIMSS001','Region']='Phillipines'
df.loc[df.BackupServer == 'XDPIAUN001.FDNET.COM','Region']='USA'
df.loc[df.BackupServer == 'XESIMSS002.FDNET.COM','Region']='Saudi Africa'
```

```
df.loc[df.BackupServer == 'yhpiaun001','Region']='Kazakhstan'
df.loc[df.BackupServer == 'yjpiaun001','Region']='Kazakhstan'
df.loc[df.BackupServer == 'ZBSIMSS001.fdnet.com','Region']='Kuwait'
df.loc[df.BackupServer == 'ZLPIAUN001.FDNET.COM','Region']='USA'
df.loc[df.BackupServer == 'hasibak004.fdnet.com','Region']='Netherland
s'
df.loc[df.BackupServer == 'FCSIBAK002 ','Region']='Phillipines'
df.loc[df.BackupServer == 'MLSIBAK003','Region']='Phillipines'
```

In [114]: `df.head()`

Out[114]:

	BackupServer	BackupDay	StartDate	ClientName	BackupStatus	Backup re-run(Y/N)
0	LVPIAUN001.FDNET.COM	2021-02-13 00:00:00	Fri Feb 12 18:00:02 PET 2021	lvpimss001.fdnet.com	Completed	No
1	LVPIAUN001.FDNET.COM	2021-02-13 00:00:00	Fri Feb 12 18:00:02 PET 2021	lvppdcx001.fdnet.com	Completed	No
2	LVPIAUN001.FDNET.COM	2021-02-13 00:00:00	Fri Feb 12 18:00:01 PET 2021	lvppfnp001.fdnet.com	Completed	No
3	SNPIAUN001.FDNET.COM	2021-02-13 00:00:00	Fri Feb 12 20:00:14 CLST 2021	lpppfnp001.fdnet.com	Completed	No
4	SNPIAUN001.FDNET.COM	2021-02-13 00:00:00	Fri Feb 12 20:00:14 CLST 2021	snppfil007.fdnet.com	Completed	No

```
In [115]: df['BackupDay']=pd.to_datetime(df.BackupDay)
```

```
In [116]: df['day_of_week'] = df['BackupDay'].dt.day_name
```

```
In [117]: df['SL'] = df['BackupDay'].dt.day_name()
```

```
In [118]: df.head()
```

Out[118]:

	BackupServer	BackupDay	StartDate	ClientName	BackupStatus	Backup re-run(Y/N)
0	LVPIAUN001.FDNET.COM	2021-02-13	Fri Feb 12 18:00:02 PET 2021	lvpimss001.fdnet.com	Completed	No
1	LVPIAUN001.FDNET.COM	2021-02-13	Fri Feb 12 18:00:02 PET 2021	lvppdcx001.fdnet.com	Completed	No
2	LVPIAUN001.FDNET.COM	2021-02-13	Fri Feb 12 18:00:01 PET 2021	lvppfnp001.fdnet.com	Completed	No
3	SNPIAUN001.FDNET.COM	2021-02-13	Fri Feb 12 20:00:14 CLST 2021	lpppfnp001.fdnet.com	Completed	No
4	SNPIAUN001.FDNET.COM	2021-02-13	Fri Feb 12 20:00:14 CLST 2021	snppfil007.fdnet.com	Completed	No

```
In [119]: df.drop(['day_of_week'],axis=1,inplace=True)
```

In [120]: `df.head()`

Out[120]:

	BackupServer	BackupDay	StartDate	ClientName	BackupStatus	Backup re-run(Y/N)
0	LVPIAUN001.FDNET.COM	2021-02-13	Fri Feb 12 18:00:02 PET 2021	lvpimss001.fdnet.com	Completed	No
1	LVPIAUN001.FDNET.COM	2021-02-13	Fri Feb 12 18:00:02 PET 2021	lvppdcx001.fdnet.com	Completed	No
2	LVPIAUN001.FDNET.COM	2021-02-13	Fri Feb 12 18:00:01 PET 2021	lvppfnp001.fdnet.com	Completed	No
3	SNPIAUN001.FDNET.COM	2021-02-13	Fri Feb 12 20:00:14 CLST 2021	lpppfnp001.fdnet.com	Completed	No
4	SNPIAUN001.FDNET.COM	2021-02-13	Fri Feb 12 20:00:14 CLST 2021	snppfil007.fdnet.com	Completed	No

In [121]: `df.shape`

Out[121]: `(42320, 9)`

In [122]: `df.to_excel("SL44_02_FINAL.xlsx",index=False)`

In [ ]: