

# Anjuman-I-Islam's

# AKBAR PEERBHOY COLLEGE OF COMMERCE AND ECONOMICS

M.S Road Do Taki, Mumbai- 400008

# **Data Science**

Submitted by

\_\_\_\_\_

Guided by – Prof. Hanna Saadya Shaikh

University of Mumbai

MSC IT (Information Technology) – Semester 1

Academic Year 2024-2025

# **INDEX**

Practical No		Title	Date	Sign
Practical 1		Create a data model using Cassandra	08-10-23	
Practical 2		Conversion from different formats to HORUS	14-10-23	
	A	CSV to HORUS		
	В	XML to HORUS		
	С	JSON to HORUS		
	D	MySQL Database to HORUS		
	Е	Picture to HORUS		
	F	Video to HORUS		
	G	Audio to HORUS		
Practical 3		Utilities and Auditing	21-10-23	
	A	Fixer Utilities		
	В	Data Binning or Bucketing		
	С	Aggregation of Data		
	D	Outlier Detection		
	Е	Audit		
Practical 4		Retrieving Data	28-10-23	
	A	Data Processing		
	В	Retrieve different attributes of data		
	С	Data Pattern		
	D	Loading IP_DATA_ALL.csv		

	E Building a diagram for scheduling of jobs			
	F	Connecting other Data Sources		
Practical 5		Assessing Data	04-11-23	
Practical 6		Processing Data	11-11-23	
Practical 7		Transforming Data	18-11-23	
Practical 8		Organizing Data	25-11-23	
	A	Horizontal Style		
	В	Vertical Style		
	С	Island Style		
	D	Secure Vault Style		
Practical 9		Reporting Data	02-12-23	
	A	Create a network routing diagram		
	В	Directed Acyclic Graph		
	С	Graphics		

# **Practical 1**

# **Creating Data Model using Cassandra**

## Cassandra Data Model

```
Create keyspace keyspace1 with replication = {'class':'SimpleStrategy','replication factor': 3};
Use keyspace1;
Create table dept ( dept_id int PRIMARY KEY, dept_name text, dept_loc text);
Create table emp (emp id int PRIMARY KEY, emp name text, dept id int, email text, phone text);
Insert into dept (dept id, dept name, dept loc) values (1001, 'Accounts', 'Mumbai');
Insert into dept (dept id, dept name, dept loc) values (1002, 'Marketing', 'Delhi');
Insert into dept (dept_id, dept_name, dept_loc) values (1003, 'HR', 'Chennai');
Insert into emp (emp_id, emp_name, dept_id, email, phone) values (1001, 'ABCD',1001,
'abcd@company.com', '1122334455');
Insert into emp (emp_id, emp_name, dept_id, email, phone) values (1002, 'DEFG',1001,
'defg@company.com', '2233445566');
Insert into emp (emp_id, emp_name, dept_id, email, phone) values (1003, 'GHIJ',1002,
'ghij@company.com', '3344556677');
Insert into emp (emp_id, emp_name, dept_id, email, phone) values (1004, 'JKLM', 1002,
'jklm@company.com', '4455667788');
Insert into emp (emp_id, emp_name, dept_id, email, phone) values (1005, 'MNOP',1003,
'mnop@company.com', '5566778899');
Insert into emp (emp_id, emp_name, dept_id, email, phone) values (1006, 'MNOP',1003,
'mnop@company.com', '5566778844');
select * from emp;
select * from dept;
update dept set dept name='Human Resource' where dept id=1003;
delete from emp where emp_id=1006;
select * from emp;
```

```
_ D X
C:\Windows\system32\cmd.exe - cqlsh
cqlsh:keyspace1> Insert into emp ( emp_id, emp_name, dept_id, email, phone ) val , ues (1004, 'JKLM',1002, 'jklm@company.com', '4455667788'); cqlsh:keyspace1> Insert into emp ( emp_id, emp_name, dept_id, email, phone ) val ues (1005, 'MNOP',1003, 'mnop@company.com', '5566778899'); cqlsh:keyspace1> Insert into emp ( emp_id, emp_name, dept_id, email, phone ) val ues (1006, 'MNOP',1003, 'mnop@company.com', '5566778844'); cqlsh:keyspace1> select * from emp;
  | emp_name | phone
                                                                                                                 5566778844
4455667788
5566778899
1122334455
3344556677
2233445566
                                          mnop@company.com
jk1m@company.com
mnop@company.com
abcd@company.com
ghij@company.com
defg@company.com
                                                                                                      MNOP
JKLM
MNOP
ABCD
GHIJ
       1004
1005
                               1002
                               1003
1001
1002
1001
       1002
                                                                                                       DEFG
(6 rows)
cqlsh:keyspace1> select * from dept;
  dept_id | dept_loc | dept_name
                             Mumbai | Accounts
Chennai | HR
Delhi | Marketing
          1003
1002
(3 rows)
cqlsh:keyspace1> update dept set dept_name='Human Resource' where dept_id=1003;
cqlsh:keyspace1> select * from dept;
  dept_id | dept_loc | dept_name
                             Mumbai | Accounts
Chennai | Human Resource
Delhi | Marketing
                     1003
1002
(3 rows)
cqlsh:keyspace1> delete from emp where emp_id=1006;
cqlsh:keyspace1> select * from emp;
  emp_id | dept_id | email
                                                                                         | emp_name | phone
                                                                                                                 4455667788
5566778899
1122334455
3344556677
2233445566
                                          jklm@company.com
mnop@company.com
abcd@company.com
ghij@company.com
defg@company.com
       1004
1005
1001
                               1002
1003
1001
                                                                                                       JKLM
MNOP
ABCD
                                                                                                       DEFG
(5 rows)
```

# **Practical 2**

# **Conversion from different formats to HORUS formats**

# A) Csv to Horus

import pondes as nd
import pandas as pd
# Input Agreement ====================================
sInputFileName='C:/VKHCG/05-DS/9999-Data/Country_Code.csv'
InputData=pd.read_csv(sInputFileName,encoding="latin-1")
print('Input Data Values =======')
print(InputData)
print('=======')
# Processing Rules ====================================
ProcessData=InputData
# Remove columns ISO-2-Code and ISO-3-CODE
ProcessData.drop('ISO-2-CODE', axis=1,inplace=True)
ProcessData.drop('ISO-3-Code', axis=1,inplace=True)
# Rename Country and ISO-M49
ProcessData.rename(columns={'Country': 'CountryName'}, inplace=True)
ProcessData.rename(columns={'ISO-M49': 'CountryNumber'}, inplace=True)
# Set new Index
ProcessData.set_index('CountryNumber', inplace=True)
# Sort data by CurrencyNumber
ProcessData.sort_values('CountryName', axis=0, ascending=False, inplace=True)
print('Process Data Values ==========')
print(ProcessData)
print('========')
# Output Agreement ====================================
OutputData=ProcessData
sOutputFileName='C:/VKHCG/05-DS/9999-Data/HORUS-CSV-Country.csv'
OutputData.to_csv(sOutputFileName, index = False)
print('CSV to HORUS - Done')
# Utility done
- · · · <b>· /</b> · · · · ·

	nfile('H:/VKHCG/05-DS/9999	-Data/CSV2HOF	RUS.py', w	dir='H:/V		217 218 219	 Tajikistan Tanzania, United Republic of Thailand	TJ TZ TH	TJK TZA THA	762 834 764
Innut Data \	Values ========					220	Timor-Leste	TL	TLS	626
Input Data V		ISO-2-CODE IS		ISO-M49		221	Togo	TG	TGO	768
0	Afghanistan	AF	AFG	4		222	Tokelau Tonga	TK TO	TKL TON	772 776
1	Aland Islands	AX	ALA	248		224	Trinidad and Tobago	TT	TTO	780
2	Albania	AL	ALB	8		225	Tunisia	TN	TUN	788
3	Algeria	DZ	DZA	12		226	Turkey	TR	TUR	792
4	American Samoa	AS	ASM	16		227 228	Turkmenistan Turks and Caicos Islands	TM TC	TKM TCA	795 796
5	Andorra	AD	AND	20		229	Tuvalu	TV	TUV	798
6 7	Angola Anguilla	AO AI	AGO AIA	24 660		230	Uganda	UG	UGA	800
8	Antarctica	AQ	ATA	10		231 232	Ukraine United Arab Emirates	UA AE	UKR ARE	804 784
9	Antigua and Barbuda	AG	ATG	28		233	United Kingdom	· GB	GBR	826
10	Argentina	AR	ARG	32		234	United States of America	US	USA	840
11	Armenia	AM	ARM	51		235	US Minor Outlying Islands	UM	UMI	581
12	Aruba	ΑМ	ABW	533		236 237	Uruguay Uzbekistan	UY UZ	URY UZB	858 860
13	Australia	AU	AUS	36			OLDERIS CON	02	020	000
14 15	Austria	AT A7	AUT AZE	40 31						
16	Azerbaijan Bahamas	BS BS	BHS	44						
17	Bahrain	BH	BHR	44						
18	Bangladesh	BD	BGD	50						
19	Barbados	BB	BRB	52						
20	Belarus	BY	BLR	112						
21	Belgium	BE	BEL	56						
22	Belize	BZ	BLZ	84						
23	Benin	ВЭ	BEN	204						
24	Bermuda	BM	BMU	60		238	Vanuatu	VU	VUT	548
25	Bhutan	BT	BTN	64		239 240	Venezuelaÿ(Bolivarian Republic) Viet Nam	VE VN	VEN VNM	862 704
26 27	Bolivia Bosnia and Herzegovina	BO BA	BOL BIH	68 70		241	Virgin Islands, US	VI	VIR	850
28	Botswana	BW	BWA	72		242	Wallis and Futuna Islands	WF	WLF	876
29	Bouvet Island	BV	BVT	74		243	Western Sahara Yemen	EH YE	ESH YEM	732 887
243 244 245 246	Western Sahara Yemen Zambia Zimbabwe	EH YE ZM ZW	ESH YEM ZMB ZWE	732 887 894 716						
[247 rows x 4	columns]									
Process Data	Values ======									
	values		-		64		Bhutan			
CountryNumber	Count	ryName			64 60 204		Bermuda			
CountryNumber 716	. Count	ryName .mbabwe	=		60 204 84		Bermuda Benin Belize			
716 894	. Count	ryName mbabwe Zambia	-		60 204 84 56		Bermuda Benin Belize Belgium			
716 894 887 732	. Count Zi Western	ryName mbabwe Zambia Yemen Sahara	=		60 204 84 56 112 52		Bermuda Benin Belize Belgium Belarus Barbados			
716 894 887 732 876	. Count Zi Western Wallis and Futuna J	ryName mbabwe Zambia Yemen Sahara slands	=		60 204 84 56 112		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh			
716 894 887 732 876 850	. Zi Zi Western Wallis and Futuna J Yirgin Islar	ryName .mbabwe Zambia Yemen Sahara .slands	=		60 204 84 56 112 52 59		Bermuda Benin Belize Belgium Belarus Barbados			
716 894 887 732 876 850 704 862	Count  Zi  Western  Wallis and Futuna 1  Virgin Islar  Vi  Venezuelaÿ(Bolivarian Rey	mbabwe Zambia Yemen Sahara slands ds, US ds, US	-		60 204 84 56 112 52 59		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh			
716 894 887 732 876 850 704 862 548	Count Zi Western Wallis and Futuna l Virgin Isla Virgin Isla Virguesuelaÿ(Bolivarian Re	ryName  Imbabwe Zambia Yemen Sahara islands ids, US et Nam public) Vanuatu	-		60 204 84 56 112 52 59		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh			
716 894 887 732 876 850 704 862 548 860	Count Zi Western Wallis and Futuna l Virgin Islam Vi Venezuelaÿ(Bolivarian Re) Uzbe	cryName  mbabwe  Zambia  Yemen  Sahara  islands  ids, US  et Nam  public)  (anuatu  ekistan  lruguay	-		60 204 84 56 112 52 59		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh			
716 894 887 732 876 850 704 862 548 860 858 858	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Vivenezuelaÿ(Bolivarian Rep Uzb Uzb Uzb United States of A	cryName  mbabwe Zambia Yemen Sahara islands ids, US et Nam public) Yanuatu ekistan ruguay merica	-		60 204 84 56 112 52 59		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh			
716 894 887 732 876 850 704 862 548 860 858 840	Count  Zi  Western  Wallis and Futuna 1  Virgin Islam  Vir	mbabwe Zambia Yemen Sahara Sialands ids, US et Nam wublic) Yanuatu Kistan Iruguay merica	-		60 204 84 56 112 52 59		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh			
716 894 887 732 876 850 794 862 548 860 858 840 826 784	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Virgin Islam Vi  Venezuelaÿ(Bolivarian Rep Uzb Uzb Uzb United States of A United Arab E	mbabwe Zambia Yemen Sahara Sialands ids, US et Nam wublic) Yanuatu Kistan Iruguay merica	•		60 204 84 56 112 52 59		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh			
716 894 887 732 876 850 704 862 548 860 858 840 826 784 884 884 884 884 884 884 886 886 886 8	Count  Zi  Western  Wallis and Futuna 3  Virgin Islam  Vi  Venezuelaÿ(Bolivarian Reg  Ub  Ub  United States of A  United Arab E  United Arab E	mbabwe Zambia Yemen Sahara Sahara Silands ids, US et Nam uublic) Anuatu kkistan Iruguay merica ingdom iirates Kraine	•		60 204 84 56 112 52 59		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh			
716 884 887 732 732 876 859 704 862 548 860 858 840 826 784 804 804	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Vi Venezuelaÿ(Bolivarian Rep United States of # United Arab En United Arab En US Minor Outlying 1	mbabwe Zambia Yemen Sahara Sahara Salands ds, US et Nam ublic) 'anuatu ekistan 'ruguay merica tirates lkraine Uganda Salands			60 204 84 56 112 52 50 48		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain			
716 884 887 732 876 732 876 859 859 704 862 548 868 858 848 848 848 884 888 884 889 884 889 884	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Vi Venezuelaÿ(Bolivarian Rep United States of # United Arab En United Arab En US Minor Outlying 1 Turks and Caicos 1	mbabwe Zambia Yemen Sahara Silands dds, US et Nam sublic) Yanuatu skistan Iruguay merica dingdom dirates Kkraine Uganda Silands	•		60 204 84 56 112 52 50 48		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain			
716 894 887 732 876 850 7704 862 862 868 869 858 840 826 784 880 880 880 798	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Vivenezuelaÿ(Bolivenan Rep Uzb Uzb United Nunited Nunited Arab E United Arab E Us Minor Outlying 1 Turks and Caicos 1 Turks	mbabwe Zambia Yemen Sahara Salands ds, US ect Nam wublic) Yamuatu Kistan Hruguay merica ingdom inates Kraine Uganda Salands Tuvalu Salands Tuvalu Salands	-		60 204 84 56 112 52 50 48		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahamas Azerbaijan Austria			
716 894 887 732 876 732 876 859 704 858 859 862 858 868 858 840 8784 8804 8804 8804 8804 8796 7992	Count Zi Western Wallis and Futuna I Virgin Islam Virgin Islam Vi Venezuelaÿ(Bolivarian Rep United States of # United Arab En	mbabwe Zambia Yemen Sahara Sahara Silands dds, US et Nam Jubic Nam	•		60 204 84 56 112 52 50 48		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahrain  Bahamas Azerbaijan Australia Aruba			
716 894 887 732 876 732 876 859 764 859 862 548 869 858 840 886 884 8804 8804 8804 798 799 799 796	Count Zi Western Wallis and Futuna I Virgin Islam Virgin Islam Vi Venezuelaÿ(Bolivarian Rep United States of # United Arab En	mbabwe Zambia Yemen Sahara Silands Suds, US et Nam uublic) Amuatu kistan Iruguay merica iingdom dirates Kraine Uganda Silands Suds Uganda Silands Tuvalu Silands Tuvalu Turkey unisia Tobago	•		60 204 84 56 112 52 50 48 44 43 44 31 40 36 533 51 32		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahrain  Bahamas Azerbaijan Australia Aruba Armenia Argentina			
716 894 887 732 876 732 876 859 704 862 548 860 858 840 826 784 884 894 996 797 784 887 796 798	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Vivenezuelaÿ(Bolivarian Ret Uzbe United Nunited Nunited Arab E United Arab E Us Minor Outlying 1 Turks and Caicos 1 Turks	mbabwe Zambia Za	:		60 204 84 56 112 52 50 48 44 44 41 40 36 533 51 32 28	Anti	Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahamas Azerbaijan Aastralia Aruba Armenia Argentina			
716 894 887 732 876 732 876 859 704 865 859 704 880 862 646 804 800 804 800 779 792 788 7792 7880 7772	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Vivenezuelaÿ(Bolivarian Ret Uzbe United Nunited Nunited Arab E United Arab E Us Minor Outlying 1 Turks and Caicos 1 Turks	mbabwe Zambia Wemen Zambia Yemen Sahara Silands dds, US et Nam ublic) Manualti Wistan Ingola Manualti Wistan Ingola Miraga Wemerica ingdom dirates Kraine Uganda Silands Tuvalu Silands Tuvalu Turkey unisia Tobago Tonga Okelau	•		60 204 84 56 1112 52 50 48 44 44 31 31 31 34 36 553 31 31 32 28 10 660	Antt	Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahrain  Bahrain  Austrolia Austrolia Austrolia Argentina ingua and Barbuda Antarctica Anguilla			
716 894 887 732 876 732 876 859 794 4 862 548 862 858 849 884 880 8804 8804 8804 8807 796 795 797 797 798	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Virgin Islam Viv Venezuelaÿ(Bolivarian Rep Uzbe United States of # United Arab En United Arab En United Arab En Turks and Caicos 1 Turkmen Trinidad and	mbabwe Zambia mbabwe Zambia Magawa Zambia Sahara Salands Sakus Set Nam nublic) Manushi kistan Iruguay merica iningdom dirates Kraine Uganda Salands Tuvalu Salands mistan Turkey unisia Tobago Tonga Okelau Togo -Leste	:		60 204 84 56 112 55 84 48 44 31 40 31 40 533 53 53 53 52 22 28 10 66 66 24	Ant	Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahrain  Bahrain  Austria Austria Aruba Armenia Argentina Andria Andria Anguilla Anguilla Anguilla			
716 894 887 732 876 732 876 859 704 862 548 860 858 840 826 784 884 894 967 784 889 786 787 787 798 798 796 772 768 666 6764	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Vivenezuelaÿ(Bolivarian Ret United Nuried Nuried Nuried Nuried Arab E  United Arab E  Us Minor Outlying 1  Turks and Caicos 1  Trimidad and	mbabwe Zambia Wemen Zambia Silands Sil			60 204 84 51 52 52 52 54 48 44 31 40 36 533 51 32 28 19 20 24 20 16	Ant	Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahamas Azerbaijan Austria Aruba Armenia Argentina dantratica Angulaa Antarctica Angulaa Andorra America Samoa			
716 894 887 732 876 732 876 859 794 4 862 548 862 858 849 884 880 8804 8804 8804 8807 796 795 797 797 798	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Vivenezuelaÿ(Bolivarian Rep United States of A United Arab E United Arab E United Arab E Turks and Caicos 1 Turkmen Trinidad and Timor Timor Tanzania, United E	mbebwe Zambia Wemen Zambia Yemen Sahara Silands dds, US et Nam uublic) Annuatu kkistan Iruguay merica iingdom dirates Kraine Uganda Silands Tuvalu Silands Tuvalu Viguay unisia Tuka Viguay unisia Tobago Tonga okelau Togo	:		60 204 84 56 1112 52 50 48 44 43 44 31 40 36 553 553 553 553 553 53 53 52 22 24 29	Ant	Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahrain  Bahrain  Bahrain  Austria Austria Austria Armenia Armenia Armenia Angula Andorra Anguilla Andorra Anguilla Andorra American Samoa Algeria			
716 894 887 732 876 732 876 859 704 4 887 784 888 889 884 884 880 884 880 884 880 786 797 796 795 797 797 798 779 788 777 768 777 768 762 764	Count  Zi  Western Wallis and Futuna I Virgin Islam Virgin Islam Vivenezuelaÿ(Bolivarian Rep Uzbe United States of A United Arab En United Arab En United Arab En United Arab En Turks and Caicos I Turkmen Trinidad and Timor Timor Tanzania, United Repui	rryName mbbbwe Zambia Mbbbwe Zambia Sahara Silands Sahara Silands Saky US et Nam Namustu Kkistan Hruguay merica Hruguay merica Silands Silands Silands Tuvalu Uganda Silands Tuvalu Uganda Silands Tuvalu Uganda Silands Tuvalu Lislands Tuvalu Lislands Tuvalu Lislands Hrukey Lislands L	:		60 204 84 56 1112 52 50 48 44 43 44 31 40 36 533 533 533 531 522 28 29 19 10 20 21 20 21 21 21 21 21 21 21 21 21 21 21 21 21	Ant	Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahrain  Bahrain  Bahrain  Bahrain  Austria Austria Australia Armenia Armenia Angula Angula Angula Anguilla Andorra American Samoa Algeria Albania Aland Islands			
716 894 887 732 876 732 876 859 704 8659 862 548 860 858 840 826 784 884 894 987 784 888 796 775 792 788 789 776 772 768 626 626 624	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Virgi	mbabwe Zambia wa Wanna Zambia Sahara Silands dds, US et Nam Namina Wanna			60 204 84 51 52 52 50 48 44 31 40 36 533 51 32 28 29 29 29 29 29 21 21 28	Antt	Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahamas Azerbaijan Austria Aruba Armenia Argentina igu and Barbuda Antarctica Anguilla Andorra America Samoa Algeria Albania			
716 894 887 732 876 732 876 859 704 8659 862 548 860 858 840 826 784 884 894 978 787 787 788 898 796 775 792 768 626 6764 834 762 74 72	Count  Zi  Western Wallis and Futuna 1 Virgin Islam Virgi	mbabhwe Zambia mbabhwe Zambia mbabhwe Zambia mbabhwe Zambia msa sahara sislands sds, US et Nam nustu kkistan ruguay merica mirates kraine Uganda sislands Tuvalu sislands Tuvalu mistan Turkey unisia Tobago Tonga okelau Togo Leste sailand liic of kistan Licate skistan Turkey unisia Tobago Tonga okelau Togo Leste sailand liic of kistan Island strawara turkey unisia Tobago Tonga okelau Togo Leste sailand liic of kistan Island strawara turkey unisia Tobago Tonga okelau Togo Leste sailand liic of kistan Island strawara turkey unisia Tobago Tonga okelau Togo Leste sailand liic of kistan Island strawara turkawara tur			60 204 84 51 52 52 50 48 44 31 40 36 533 51 32 28 29 29 29 29 29 21 21 28		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahamas Azerbaijan Austria Aruba Armenia Aruba Argentina igua and Barbuda Angentia Angola Andorra America Anguila Angora America Anguila Angora Andorra Albenia Albania Albania Aland Islands Afghanistan			
716 894 887 732 876 732 876 859 794 4 887 796 858 849 880 880 880 880 880 876 779 780 779 780 7772 768 834 762 764	Count  Zi  Western Wallis and Futuna I Virgin Islam Vivenezuelaÿ(Bolivarian Rep United I United States of A United Arab En  US Minor Outlying I Turks and Caicos I Turkmen  Trinidad and  Timor Tanzania, United Reput Bouvet Bosnia and Herze	mbabhwe Zambia mbabhwe Zambia mbabhwe Zambia mbabhwe Zambia msa sahara sislands sds, US et Nam nustu kkistan ruguay merica mirates kraine Uganda sislands Tuvalu sislands Tuvalu mistan Turkey unisia Tobago Tonga okelau Togo Leste sailand liic of kistan Licate skistan Turkey unisia Tobago Tonga okelau Togo Leste sailand liic of kistan Island strawara turkey unisia Tobago Tonga okelau Togo Leste sailand liic of kistan Island strawara turkey unisia Tobago Tonga okelau Togo Leste sailand liic of kistan Island strawara turkey unisia Tobago Tonga okelau Togo Leste sailand liic of kistan Island strawara turkawara tur	:		60 204 84 56 1112 52 50 48 44 31 40 36 36 533 51 22 28 10 66 66 66 66 66 66 66 66 66 66 66 66 66		Bermuda Benin Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahrain  Bahrain  Bahrain  Bahrain  Austria Austria Australia Armenia Armenia Angula Angula Angula Anguilla Andorra American Samoa Algeria Albania Aland Islands			

### b) XML to Horus

```
import pandas as pd
import xml.etree.ElementTree as ET
def df2xml(data):
header = data.columns
root = ET.Element('root')
for row in range(data.shape[0]):
entry = ET.SubElement(root,'entry')
for index in range(data.shape[1]):
schild=str(header[index])
child = ET.SubElement(entry, schild)
if str(data[schild][row]) != 'nan':
child.text = str(data[schild][row])
else:
child.text = 'n/a'
entry.append(child)
result = ET.tostring(root)
return result
def xml2df(xml data):
root = ET.XML(xml data)
all records = []
for i, child in enumerate(root):
record = \{\}
for subchild in child:
record[subchild.tag] = subchild.text
all records.append(record)
return pd.DataFrame(all_records)
sInputFileName='C:/VKHCG/05-DS/9999-Data/Country_Code.xml'
InputData = open(sInputFileName).read()
print('=======')
print('Input Data Values =========')
print('=======')
print(InputData)
print('=======')
ProcessDataXML=InputData
# XML to Data Frame
ProcessData=xml2df(ProcessDataXML)
# Remove columns ISO-2-Code and ISO-3-CODE
ProcessData.drop('ISO-2-CODE', axis=1,inplace=True)
ProcessData.drop('ISO-3-Code', axis=1,inplace=True)
# Rename Country and ISO-M49
```

ProcessData.rename(columns={'Country': 'CountryName'}, inplace=True)
ProcessData.rename(columns={'ISO-M49': 'CountryNumber'}, inplace=True)
# Set new Index
ProcessData.set_index('CountryNumber', inplace=True)
# Sort data by CurrencyNumber
ProcessData.sort_values('CountryName', axis=0, ascending=False, inplace=True)
print('==============')
print('Process Data Values ========')
print('=============')
print(ProcessData)
_ print('====================================
OutputData=ProcessData
sOutputFileName='C:/VKHCG/05-DS/9999-Data/HORUS-XML-Country.csv'
OutputData.to_csv(sOutputFileName, index = False)
print('============')
print('XML to HORUS - Done')
print('============')
# Utility done ====================================

## **Output:**

In [105]: runfile('H:/VKHCG/05-DS/9999-Data/XML2HORUS.py', wdir='H:/VKHCG/05-DS/9999-Data')

Input Data Values -----

entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><entry><en ISO-3-Code><ISO-M49>533</ISO-M49><ISO-M49>533</ISO-M49></entry><entry><entry><entry>Australia</

Country><Country>Australia</Country><ISO-2-CODE>AU</ISO-2-CODE><ISO-2-CODE>AU</ISO-2 CODE><ISO-3-Code>AUS</ISO-3-Code><ISO-3-Code>AUS</ISO-3-Code><ISO-M49>36</ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><I CUDE713D9-3-LOBAPHAY/BIT-3-COBPASSO-3-COBPAS M49>31</ISO-M49></entry><entry><Country>Bahamas</Country>Country>Bahamas</Country><ISO-2-CODE>85</ISO-2-CODE><ISO-2-CODE><ISO-2-CODE><ISO-3-Code>BISO-ISO-3-Code>LSO-M99-44(/ISO-M99-XSO-M99-44(/ISO-M99-X)-(entry)-country>Sahrain
 Country>Country>Bahrain
 Country>COUNTry>Bahrain
 Country>COUNTry>Bahrain
 Country>COUNTry>Bahrain
 Country>COUNTry>COUNTry>Bahrain
 Country>COUNTRY>COUNTRY>CO CLUMP31-98-01-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0998/(1)-0-1-00-0988/(1)-0-1-

Code><ISO-M49>52</ISO-M49><ISO-M49>52</ISO-M49></entry><entry><Country>Belarus</ Code+13.3-489324/130-14993-130-14993-130-14993-(entry)-centry/scountry/selaruss/ Country/scountry/selaruss/Country/sel3-2-c005894/150-2-c0058-150-2-c005894/150-2-c005-150-2-c005894/150-2-c005-150-3-c0069188-4/150-3-c006918 CODE><ISO-3-Code>BLZ</ISO-3-Code><ISO-3-Code>BLZ</ISO-3-Code><ISO-M49><ISO CUDEY13D-3-C00891K2/13D-3-C00874SD-3-C00874SD-3-C00874SD-3-C00874SD-3-M9934K2/3D-M9934K2/3D-M993K2/3D-M993K2/3D-M993K2/3D-M993K2/3D-M993K2/3D-3-C00874SD-3

9

 $\label{localization} {\tt CODE}{\footnotesize <} {\tt ISO-3-Code}{\footnotesize >} {\tt BMU}{\footnotesize <} {\tt ISO-3-Code}{\footnotesize <} {\tt ISO-3-Code}{\footnotesize <} {\tt ISO-1000}{\footnotesize <} {\tt ISO-10000}{\footnotesize <} {\tt ISO-1000}{\footnotesize <} {\tt ISO-1000}{\footnotesize <} {\tt ISO-1000}{\footnotesize <} {\tt ISO-10$ M49x68x/ISO-M49x/entryx<countryx8hutanx/CountryxCountryx8hutanx/CountryxISO-2-CODEx8T</ISO-2-CODEx8T</ISO-2-CODEx8T</ISO-2-CODEx8T</ISO-3-Codex8TMx/ISO-3-Code ISO-3-Code><ISO-M9>64</ISO-M9><ISO-M9>64</ISO-M9>64</ISO-M9></entry><entry><country>8olivia</ri></ra>
Country><Country>80livia
Country>80livia
Country>6150-2-C00E>80</ISO-2-C00E>C00E>C00-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</ISO-2-C00E>80</I CODE><ISO-3-Code>BOL</ISO-3-Code><ISO-3-Code>CISO-3-Code><ISO-M49>68</ISO-M49><ISO-M49>CISO-M M49>68//ISO-M49>/entry><country>Bosnia and Herzegovina</country>Country>Bosnia and Herzegovina</country>Country>Bosnia and Herzegovina</country>CISO-2-CODE>BA</ISO-2-CODE>BA</ISO-2-CODE>BA</ISO-2-CODE>BA</ISO-2-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE>BA</ISO-3-CODE> ISO-3-Code><ISO-3-Code>BIH</ISO-3-Code><ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</ISO-N49>70</IS entry><Country>&Country>Botswana</Country>Country>Botswana</Country>Country>Botswana</Country>CODE>BIK/ISO-2-CODE>BIK/ISO-3-Code>BIMA</ISO-3-Code>CODE>BIK/ISO-3-Code>BIMA</ISO-3-Code>BIMA</ISO-3-Code>BIMA</ISO-3-Code>BIMA Code><ISO-M49>72</ISO-M49><ISO-M49>72</ISO-M49></entry><country>Bouvet Island</ Country><Country>Bouvet Island</Country>ISO-2-CODE>BV</ISO-2-CODE>ISO-2-CODE>ISO-2-CODE>ISO-3-Code>BV</ISO-3-CODE>ISO-3-CODE>BV</ISO-3-CODE>ISO-3-CODE>BVT</ISO-3-CODE>ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-CODE>BVT</ISO-3-C M49>74</ISO-M49></entry><entry><Country>Brazil</Country><Country>Brazil</Country>Count CODE>BR</ISO-2-CODE>ISO-2-CODE>BR</ISO-2-CODE>SRA/ISO-3-Code>BRA</ISO-3-Code>SRA</ISO-3-Code>ISO-3-Islands</Country><Country>British Virgin Islands</Country><ISO-2-CODE>VG</ISO-2-CODE><ISO-2 CODE>VG</ISO-2-CODE><ISO-3-Code>VGB</ISO-3-Code><ISO-3-Code>VGB</ISO-3-Code><ISO-1449>92</ ISO-M49><ISO-M49>92</ISO-M49></entry><entry><Country>British Indian Ocean Territory</ Country><Country>British Indian Ocean Territory</Country><ISO-2-CODE>IO</ISO-2-CODE><ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE>IO</ISO-2-CODE> CODE:10/TSO-2-CODE:XISO-3-Code:IOTC/ISO-3-Code:XISO-3-ISO-2-CODE><ISO-3-Code>BRNK/ISO-3-Code><ISO-3-Code>BRNK/ISO-3-Code><ISO-M49>96</ISO-M49><ISO-M49>96</ISO-M49></entry><entry><Country>Bulgaria</Country><Country>Bulgaria</ Country><ISO-2-CODE>BG</ISO-2-CODE>KISO-2-CODE>BG</ISO-2-CODE>KISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR</ISO-3-Code>BGR< Code><ISO-3-Code>BGR</ISO-3-Code><ISO-M49>100</ISO-M49>100</ISO-M49>100</ISO-M49></ entry><entry><Country>Burkina Faso</Country><Country>Eurkina Faso</Country><ISO-2-CODE>BF</ ISO-2-CODE>KISO-2-CODE>BF</ISO-2-CODE>KISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-Code>BFA</ISO-3-C Country>Country>Burundi</Country>Country>Country>Country>Country>Burundi</Country>Country>Country>Country>Burundi</Country>Country>Country>Country>Burundi</Country>Country>Country>Country>Burundi</Country>Country>Country>Country>Country>Burundi</Country> CODE><ISO-3-Code>BDI</ISO-3-Code><ISO-3-Code>BDI</ISO-3-Code><ISO-M49>108</ISO-M49><ISO-M49> M49>108</ISO-M49></entry><country><Country><Country><Country><ISO-2-CODE>KH</ISO-2-CODE>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-3-Code>KH</ISO-Code><ISO-3-Code>KHM</ISO-3-Code><ISO-M49>116</ISO-M49><ISO-M49>116</ISO-M49>/ entry><entry><Country>Cameroon</Country>Country>Cameroon</Country>Country>Cameroon</Country>Cameroon</Country>Cameroon</Country>Cameroon</Country>CISO-2-CODE>CI%(/ISO-2-CODE>CISO-3-Code>CMR</ISO-3-Code>CMR</ISO-3-Code>CMR</ISO-3-Code>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-COde>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CODE>CMR</ISO-3-CMR</ISO-3-CMR</ISO-3-CMR</ISO-3-CMR</ISO-3-CMR</ISO-3-CM Code><ISO-M49>120</ISO-M49><ISO-M49>120</ISO-M49></entry><country>Country>Canada</ Country><Country>Canada</Country><ISO-2-CODE>CA</ISO-2-CODE><ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2-CODE>CA</ISO-2 CODE><ISO-3-Code>CANK/ISO-3-Code><ISO-3-Code>CANK/ISO-3-Code>CISO-M49>124</ISO-M49>C M49>124</ISO-M49></entry><entry><Country>Cape Verde</Country><Country>Cape Verde</

Country>\ISO-2-CODE>CV</ISO-2-CODE>ISO-2-CODE>CV</ISO-2-CODE>CISO-3-CODE>CISO-3-CODE>CVISO-3-CODE>CVISO-3-CODE>CVISO-3-CODE>CVISO-3-CODE>CVISO-3-CODE>CVISO-3-CODE>CVISO-3-CODE>CVISO-3-CODE>CODE>CVISO-3-CODE>CVISO-

2

African Republic</Country><Country>Central African Republic</Country><ISO-2-CODE>CF</ISO-2-CODE><ISO-2-CODE>CF</ISO-2-CODE><ISO-3-Code>CAF</ISO-3-Code><ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-Code>CAF</ISO-3-COde>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</ISO-3-CODE>CAF</I Code><ISO-M49>140</ISO-M49><ISO-M49>140</ISO-M49></entry><entry><Country>Chad</ Country><Country>Chad</Country><ISO-2-CODE>TD</ISO-2-CODE><ISO-2-CODE>TD</ISO-2-CODE>TD</ISO-3-Code>TCD</ISO-3-Code><ISO-3-Code>TCD</ISO-3-Code><ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49>148</ISO-M49 M49></entry><entry><Country>Chile</Country><Country>Chile</Country><ISO-2-CO0E>CL</ISO-2-CODE><ISO-2-CODE>CL</ISO-2-CODE><ISO-3-Code>CHL</ISO-3-Code><ISO-3-Code>CHL</ISO-3-Code><ISO-M49>152</ISO-M49><ISO-M49>152</ISO-M49></entry><entry><Country>China</ Country>Country>China</Country><ISO-2-CODE>CN</ISO-2-CODE><ISO-2-CODE>CN</ISO-2-CODE><ISO-3-Code>CHN</ISO-3-Code><ISO-3-Code>CHN</ISO-3-Code><ISO-N49>156</ISO-N49><ISO-M49>156</ISO-M49></entry><entry><Country>Hong Kong, SAR China</Country><Country>Hong Kong, SAR China</Country><ISO-2-CODE>HK</ISO-2-CODE><ISO-2-CODE>HK</ISO-2-CODE><ISO-3-Code>HKG</ ISO-3-Code><ISO-3-Code>HKG</ISO-3-Code><ISO-M49>344</ISO-M49><ISO-M49>344</ISO-M49></ entry><Country>Macao, SAR China</Country><Country>Macao, SAR China</Country><ISO-2-CODE>MO</ISO-2-CODE><ISO-2-CODE>MO</ISO-2-CODE><ISO-3-Code>MAC</ISO-3-Code><ISO-3-Code>MAC</ ISO-3-Code><ISO-M49>446</ISO-M49><ISO-M49>446</ISO-M49></entry><entry><Country>Christmas Island</Country>Country>Christmas Island</Country><ISO-2-CODE>CX</ISO-2-CODE>CISO-2-CODE>CX</ISO-2-CODE><ISO-3-Code>CXR</ISO-3-Code><ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-Code>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-COde>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CXR</ISO-3-CODe>CX

ISO-M49><ISO-M49>166</ISO-M49></entry><entry><Country>Colombia</Country><Country>Colombia</ Country><ISO-2-CODE>CO</ISO-2-CODE><ISO-2-CODE>CO</ISO-2-CODE>CO</ISO-3-Code>COL</ISO-3-Code><ISO-3-Code>COL</ISO-3-Code><ISO-M49>170</ISO-M49>170</ISO-M49>170</ISO-M49></ entry><entry><Country>Comoros</Country>Country>Comoros</Country><ISO-2-CODE>KH</ISO-2-CODE><ISO-2-CODE>KVK/ISO-2-CODE><ISO-3-Code>COM</ISO-3-Code><ISO-3-Code>COM</ISO-3 Code><ISO-M49>174</ISO-M49><ISO-M49>174</ISO-M49></entry><country>Congo&#255; (Brazzaville)</Country><Country>Congo&#255;(Brazzaville)</Country><ISO-2-CODE>CG</ISO-2-Code><ISO-M99178</ISO-M9><ISO-M99178</ISO-M99178</ISO-M992</td>
Country>Country>Congo, (Kinshasa)

Country>Country>Congo, (Kinshasa)
Country>CODE>CD
CODE>CD ISO-2-CODE><ISO-3-Code>CODe>/ISO-3-Code><ISO-3-Code>CODe>/ISO-3-Code><ISO-M49>180</ISO-M49><ISO-M49>180</ISO-M49></entry><entry><Country>Cook Islands</Country><Country>Cook Islands</Country><ISO-2-C00E>CK</ISO-2-C00E>CK</ISO-2-C00E>CK</ISO-2-C00E>CK</ISO-3-Code>C0K</ISO-3-Code>C0K</ISO-3-C0DE>CK</ISO-3-C0DE>CISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184</ISO-149>184 entry><entry><Country>Costa Rica</Country><Country>Costa Rica</Country><ISO-2-CODE>CR</ ISO-2-CODEXTSO-2-CODEXCRX/ISO-2-CODEXCTSO-3-CodeXCRIX/ISO-3-CodeXISO-3-CodeXCRIX/ISO-3-CodeXCRIX/ISO-3-CodeXCRIX/ISO-3-CodeXISO-3-CodeXISO-N49>1887/ISO-N49>X887/ISO-N49-X88/ISO-N49-X88/ISO-N49-X88/ISO-N49-XISO-X88/ISO-X88/ISO-X88/ISO-X88/ISO-X88/ISO-X8 d'Ivoire</Country><Country>C&#147;te d'Ivoire</Country><ISO-2-CODE>CI</ISO-2-CODE><ISO-2-CODE>CI</ISO-2-CODE><ISO-3-Code>CIV</ISO-3-Code><ISO-3-Code>CIV</ISO-3-Code><ISO-1449>384</ ISO-N49><ISO-N49>384</ISO-N49></entry><entry><Country>Croatia</country>Country>Croatia</country><ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR</ISO-2-CODE>HR< Code><ISO-3-Code>HRV</ISO-3-Code><ISO-M49>191</ISO-M49><ISO-M49>191</ISO-M49>191</ISO-M49></ entry><entry><Country>Country>Cuba</Country>Country>Cuba</Country><ISO-2-CODE>CISO-2-CODE>CISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-Code>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO-3-CODE>CUB</ISO Code><ISO-M49>192</ISO-M49><ISO-M49>192</ISO-M49></entry><country>Country>Cyprus</ Country>Country>Cyprus</Country><ISO-2-CODE>CY</ISO-2-CODE>CISO-2-CODE>CY</ISO-2-CODEX COMPANY SPECIAL PROPERTY OF A CONTRACT M49>196</ISO-M49></entry><entry><Country>Czech Republic</Country>CZech Republic</ Country><ISO-2-CODE>CZ</ISO-2-CODE>CISO-2-CODE>CZ</ISO-2-CODE>CISO-3-Code>CZE</ISO-3-Code><ISO-3-Code>CZE</ISO-3-Code><ISO-M49>203</ISO-M49>203</ISO-M49>203</ISO-M49>2 entry><entry><Country>Denmark</Country>Country>Denmark</Country><ISO-2-CODE>DK</ISO-2-CODE><ISO-2-CODE>DK</ISO-2-CODE>XISO-3-Code>DNK</ISO-3-Code>CODE>XISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>DNK</ISO-3-Code>D Code>xISO-M49>208</ISO-M49>208</ISO-M49>208</ISO-M49>/entry><country>Ofibouti</Country>Country>Ofibouti</Country><COUNTRY>Ofibouti</Country>Country>Ofibouti</Country><ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-2-CODE>DI</ISO-CODE><ISO-3-Code>D3I</ISO-3-Code><ISO-3-Code>D3I</ISO-3-Code><ISO-M49>262</ISO-M49><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><ISO-M49>CODE><IS M49>262</ISO-M49></entry><country>Dominica</country>Country>Dominica</ Country><ISO-2-CODE>DM</ISO-2-CODE>CISO-2-CODE>DM</ISO-2-CODE>CODE>DM</ISO-3-Code>DMA</ISO-3 Code><ISO-3-Code>ONA</ISO-3-Code><ISO-N49>212</ISO-N49><ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</ISO-N49>212</I

entry><entry><Country>Dominican Republic</Country>Country>Dominican Republic</ Country><ISO-2-CODE>DO</ISO-2-CODE>SISO-2-CODE>OD</ISO-2-CODE>SISO-3-Code>DOM</ISO-3-Code>CODE>SISO-3-Code>DOM</ISO-3-Code>SIS entry><Country><Country><Country><Country><Country><Country><ISO-2-CODE>EC</ISO-2entry/sentry/sequentry/country/sequentry/country/sequent M49>818</ISO-M49></entry><entry><Country>El Salvador</Country><Country>El Salvador</ Countryx<TSO-2-CODE>SVK/ISO-2-CODE>SISO-2-CODE>SVK/ISO-2-CODE>SISO-3-Code>SLVK/ISO-3-Code>SLVK contry>centry>countrySequatorial Guinear(Country>Country>Equatorial Guinear(Country>C150-2-CODE50g/ISO-2-CODE×ISO-2-CODE50g/ISO-2-CODE50g/ISO-2-CODE50g/ISO-3-Code50g/ISO M99/232/T30-M99/c/entryy<CountryyEstonia</CountryyEstonia</CountryyEstonia</CountryYEStonia</CountryYESTOnia</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</COUNTRYYESTONIA</c> Country><Country>Ethiopia</Country><ISO-2-CODE>ET</ISO-2-CODE>TSO-2-CODE>TSO-2-CODE>TSO-2-CODE>TSO-2-CODE>TSO-2-CODE>TSO-2-CODE>TSO-2-CODE>TSO-3-Code>ETH</ISO-3-Code>TSO-3-Code M89x231t/ISO-M49x/entryx/countryxFalkland Islands (Malvinas)t/
CountryxCountryxFalkland Islands (Malvinas)c/CountryxISO-2-CODE>Fk/ISO-2-CODE>KISO-3-Code>Fk/ISO-2-CODE>KISO-3-Code>Fk/ISO-2-CODE>KISO-3-Code>Fk/ISO-ISO-M99/ISO-M99/288/ISO-M99/8/Rentry/Kentry/Kountry/Faror Islands/Country/Sountry/Faror Islands/Country/Kountry/Kountry/Faror Islands/Country/Kountry/Kountry/Kountry/Kountry/Karore Islands/Country/Kountry/Karore Islands/Country/Kountry/Karore Islands/Country/Karore Islands/C entry><Country>Fiji</Country>Country>Fiji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</Country>Siji</C Code><ISO-M49>242</ISO-M49><ISO-M49>242</ISO-M49><ISO-M49>242</ISO-M49></entry><country>Finland</country><Country>Finland</country>Finland</country>Fix|ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-2-CODE>Fi</ISO-CODE><ISO-3-Code>FIN</ISO-3-Code><ISO-3-Code>FIN</ISO-3-Code><ISO-M49>246</ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49><ISO-M49>< M49>246</TSO-M49></entry><entry><entry><country>France</country><Country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>France</country>Franc ISO-3-Code><ISO-M49>258</ISO-M49><ISO-M49>258</ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49></ISO-M49>< ISO-2-CODE><ISO-3-Code>GUF</ISO-3-Code><ISO-3-Code>GUF</ISO-3-Code><ISO-M49>254</ISO-M49><ISO-W49>254</ISO-W49></entry><country>French Polynesia</country>Country>French Polynesia</country>Country>French Polynesia</country><ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-2-CODE>F</ISO-ISO-3-Code>Code>ISO-3-Code>Country><p Territories</Country><ISO-2-CODE>TF</ISO-2-CODE><ISO-2-CODE>TF</ISO-2-CODE><ISO-3-Code>ATF</

# c) JSON to HORUS Format

# Utility Start JSON to HORUS ====================================
import pandas as pd
# Input Agreement ====================================
sInputFileName='C:/VKHCG/05-DS/9999-Data/Country_Code.json'
InputData=pd.read_json(sInputFileName, orient='index', encoding="latin-1")
print('Input Data Values ========')
print(InputData)
print('======')
# Processing Rules ====================================
ProcessData=InputData
# Remove columns ISO-2-Code and ISO-3-CODE
ProcessData.drop('ISO-2-CODE', axis=1,inplace=True)
ProcessData.drop('ISO-3-Code', axis=1,inplace=True)
# Rename Country and ISO-M49
ProcessData.rename(columns={'Country': 'CountryName'}, inplace=True)
ProcessData.rename(columns={'ISO-M49': 'CountryNumber'}, inplace=True)
# Set new Index
ProcessData.set_index('CountryNumber', inplace=True)
# Sort data by CurrencyNumber
ProcessData.sort_values('CountryName', axis=0, ascending=False, inplace=True)
print(Process Data Values ==========')
print(ProcessData)
print('=======') # Output Agreement ====================================
Output Agreement ————————————————————————————————————
sOutputFileName='c:/VKHCG/05-DS/9999-Data/HORUS-JSON-Country.csv'
OutputData.to_csv(sOutputFileName, index = False)
print('JSON to HORUS - Done')
# Utility done ====================================

# **Output:**

In [117	]: runfile('H:/VKHCG/05-DS/99	99-Data/JSON2H0	ORUS.py',	wdir='H:/VK	224		Tonga d and Tobago	TO TT	TON TTO	776 780
Input F	Data Values =======				225 226		Tunisia Turkev	TN TR	TUN	788 792
	Country	ISO-2-CODE ISO	0-3-Code		227		Turkmenistan	TM	TKM	795
0 1	Afghanistan Aland Islands	AF AX	AFG ALA	4 248	228 229	Turks and Ca		TC TV	TCA	796 798
2	Albania	AL	ALB	8	230		Tuvalu Uganda	UG	TUV UGA	800
4	Algeria American Samoa	DZ AS	DZA ASM	12 16	231		Ukraine	UA	UKR	804
5	Andorra	AD	AND	20	232 233		rab Emirates ited Kingdom	AE GB	ARE GBR	784 826
6 7	Angola Anguilla	AO AI	AGO AIA	24 660	234	United State	s of America	US	USA	840
8	Antarctica	AQ	ATA	10	235	US Minor Outl	ying Islands	UM	UMI	581
9 10	Antigua and Barbuda Argentina	AG AR	ATG ARG	28 32	236 237		Uruguay Uzbekistan	UY UZ	URY UZB	858 860
11	Armenia	AM	ARM	51			02001120001			
12 13	Aruba Australia	AW AU	ABW AUS	533 36						
14	Austria	AT	AUT	40						
15 16	Azerbaijan Bahamas	AZ BS	AZE BHS	31 44						
17	Bahrain	BH	BHR	48						
18 19	Bangladesh Barbados	BD BB	BGD BRB	50 52						
20	Belarus		BLR	112						
21 22	Belgium Belize	BE BZ	BEL BLZ	56 84						
23	Benin	BJ	BEN	204						
24 25	Bermuda		BMU BTN	60 64	238		Vanuatu	VU	VUT	548
26	Bhutan Bolivia	BO	BOL	68	239 240		an Republic) Viet Nam	VE VN	VEN	862 704
27	Bosnia and Herzegovina	BA	BIH	70	241		Islands, US	VI	VIR	850
28 29	Botswana Bouvet Island	BW BV	BWA BVT	72 74	242	Wallis and Fu	tuna Islands	WF	WLF	876
					243 244		stern Sahara Yemen	EH YE	ESH YEM	732 887
217 218	Tajikistan Tanzania, United Republic of	T3 TZ	TJK TZA	762 834	245		Zambia	ZM	ZMB	894
219	Thailand	TH	THA	764	246		Zimbabwe	ZW	ZWE	716
220 221	Timor-Leste Togo	TL TG	TLS TG0	626 768	[24	7 rows x 4 columns]				
222	Tokelau	TK	TKL	772		,				
	ess Data Values ==== ntryNumber		Coun	tryName imbabwe						
894			_	Zambia						
				Yemen				pl .		
887					64			Bhutan		
732				Sahara	60			Bermuda		
876	Lew	lis and F								
				Islands	204			Benin		
850				Islands ands, US	84			Belize		
850 704			n Isla		84 56			Belize Belgium		
	Venezuela		n Isla V	nds, US iet Nam	84 56 112			Belize Belgium Belarus		
704	Venezuela	Virgi	n Isla V ian Re	nds, US iet Nam	84 56 112 52			Belize Belgium Belarus Barbados		
704 862	Venezuela	Virgi	n Isla V ian Re	nds, US iet Nam public)	84 56 112			Belize Belgium Belarus		
704 862 548	Venezuela	Virgi	n Isla V ian Re Uzb	nds, US 'iet Nam :public) Vanuatu ekistan	84 56 112 52 50			Belize Belgium Belarus Barbados Bangladesh		
704 862 548 860		Virgi ıÿ(Bolivar	n Isla V ian Re Uzb	nds, US /iet Nam public) Vanuatu ekistan Uruguay	84 56 112 52 50			Belize Belgium Belarus Barbados Bangladesh		
704 862 548 860 858 840		Virgi aÿ(Bolivar nited Stat	n Isla V ian Re Uzb es of	nds, US /iet Nam :public) Vanuatu ekistan Uruguay America	84 56 112 52 50			Belize Belgium Belarus Barbados Bangladesh		
704 862 548 860 858 840 826		Virgi nÿ(Bolivar nited Stat	n Isla V ian Re Uzb es of nited	nds, US (iet Nam public) Vanuatu ekistan Uruguay America Kingdom	84 56 112 52 50			Belize Belgium Belarus Barbados Bangladesh		
704 862 548 860 858 840 826 784		Virgi aÿ(Bolivar nited Stat	n Isla V ian Re Uzb es of nited Arab E	ands, US (iet Nam (public) Vanuatu (pekistan Uruguay America Kingdom (mirates	84 56 112 52 50			Belize Belgium Belarus Barbados Bangladesh		
704 862 548 860 858 840 826 784 804		Virgi nÿ(Bolivar nited Stat	n Isla V ian Re Uzb es of nited Arab E	inds, US (iet Nam (public) Vanuatu (pekistan Uruguay America Kingdom (mirates Ukraine	84 56 112 52 50			Belize Belgium Belarus Barbados Bangladesh		
704 862 548 860 858 840 826 784 804 800	Un	Virgi y (Bolivar uited Stat U United	n Isla V ian Re Uzb es of nited Arab E	nds, US /iet Nam /public) Vanuatu ekistan Uruguay America Kingdom emirates Ukraine Uganda	84 56 112 52 50			Belize Belgium Belarus Barbados Bangladesh		
704 862 548 860 858 840 826 784 804	Un	Virgi nÿ(Bolivar nited Stat	n Isla V ian Re Uzb es of nited Arab E	nds, US /iet Nam /public) Vanuatu ekistan Uruguay America Kingdom emirates Ukraine Uganda	84 56 112 52 50			Belize Belgium Belarus Barbados Bangladesh		
704 862 548 860 858 840 826 784 804 800	Un	Virgi y (Bolivar uited Stat U United	n Isla V ian Re Uzb es of nited Arab E	nds, US /iet Nam /public) Vanuatu ekistan Uruguay America Kingdom emirates Ukraine Uganda	84 56 112 52 50			Belize Belgium Belarus Barbados Bangladesh		
704 862 548 860 858 840 826 784 804 800 581	Un	Virgi y (Bolivar uited Stat U United	n Isla V ian Re Uzb es of nited Arab E	nds, US /iet Nam /public) Vanuatu vekistan Uruguay America Kingdom inrates Ukraine Uganda Islands Tuvalu	84 56 112 52 50			Belize Belgium Belarus Barbados Bangladesh		
704 862 548 860 858 840 826 784 804 800 581 798	Un	Virgi gy(Bolivar nited Stat U United . Minor Out	n Isla V ian Re Uzb es of nited Arab E lying aicos	nds, US /iet Nam /public) Vanuatu vekistan Uruguay America Kingdom inrates Ukraine Uganda Islands Tuvalu	84 56 112 52 50			Belize Belgium Belarus Barbados Bangladesh		
704 862 548 860 858 840 826 784 804 800 581 798 796 795	Un	Virgi gy(Bolivar nited Stat U United . Minor Out	n Isla V ian Re Uzb es of nited Arab E lying aicos	nds, US /iet Nam :public) Vanuatu wekistan Uruguay America Kingdom ingrate Ukraine Uganda Islands Tuvalu Islands wenistan	84 56 112 52 50 48			Belize Belgium Belarus Barbados Bangladesh Bahrain		
704 862 548 860 858 840 826 784 804 800 581 798 796 795	Un	Virgi gy(Bolivar nited Stat U United . Minor Out	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm	nds, US (iet Nam cpublic) Vanuatu cekistan Uruguay America Kingdom cmirates Ukraine Uganda Islands Tuvalu Islands enistan Turkey	84 56 112 52 50 48			Belize Belgium Belarus Barbados Bangladesh Bahrain		
704 862 548 860 858 840 826 784 800 581 798 796 795 792 788	Un	Virgi gg(Bolivar mited Stat United A Minor Out	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm	nds, US (iet Nam cpublic) Vanuatu cekistan Uruguay America Kingdom cmirates Ukraine Uganda Islands Tuvalu Islands tenistan Turkey Tunisia	84 56 112 52 50 48 48			Belize Belgium Belarus Barbados Bangladesh Bahrain Bahamas		
704 862 548 860 858 840 826 784 800 581 798 796 795 792 788 780	Un	Virgi gg(Bolivar mited Stat United A Minor Out	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm	nds, US /iet Nam :public) Vanuatu ekistan Uruguay America Kingdom imirates Ukraine Uganda Islands Tuvalu Islands Islands interistan Turkey Tunisia	84 56 112 52 50 48 44 31 40			Belize Belgium Belarus Barbados Bangladesh Bahrain Bahrain Austria		
704 862 548 860 858 840 826 784 804 800 581 798 795 792 788 780 776	Un	Virgi gg(Bolivar mited Stat United A Minor Out	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and	nds, US /iet Nam :public) Vanuatu ekistan Uruguaya Kingdom imirates Ukraine Uganda Islands Tuvalu Islands enistan Turkay Turisia	84 56 112 52 50 48 44 31 40 36			Belize Belgium Belarus Barbados Bangladesh Bahrain Bahamas Azerbaijan Austria Australia		
704 862 548 860 858 840 826 784 804 809 581 798 795 792 788 780 776 772	Un	Virgi gg(Bolivar mited Stat United A Minor Out	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and	nds, US /iet Nam :public) Vanuatu ekistan Uruguay America Kingdom imirates Ukraine Uganda Islands Tuvalu Islands ien Islands ien Islands Turkey Tunisia I Tobago Tonga Tokelau	84 56 112 52 50 48 44 31 40 36 533			Belize Belgium Belarus Barbados Bangladesh Bahrain Bahrain Astria Austria Aruba		
704 862 548 860 858 840 826 784 800 581 798 795 792 788 780 776 772 768	Un	Virgi gg(Bolivar mited Stat United A Minor Out	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and	nds, US /iet Nam :public) Vanuatu ekistan Uruguaya Kingdom imirates Ukraine Uganda Islands Tuvalu Islands enistan Turkay Turisia	84 56 112 52 50 48 44 31 40 36 533 51			Belize Belgium Belarus Barbados Bangladesh Bahrain Bahamas Azerbaijan Australia Australia Armenia		
704 862 548 860 858 840 826 784 800 581 796 795 792 788 780 776 772	Un	Virgi gg(Bolivar mited Stat United A Minor Out	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm	nds, US /iet Nam :public) Vanuatu ekistan Uruguay America Kingdom imirates Ukraine Uganda Islands Tuvalu Islands ien Islands ien Islands Turkey Tunisia I Tobago Tonga Tokelau	84 56 112 52 50 48 44 31 40 36 533 51 32			Belize Belgium Belarus Barbados Bangladesh Bahrain Bahrain Azerbaijan Austria Australia Aruba Armenia Argentina		
704 862 548 860 858 840 826 784 800 581 798 795 792 788 780 776 772 768	Un	Virgi gg(Bolivar mited Stat United A Minor Out	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and	nds, US (iet Nam epublic) Vanuatu vekistan Uruguaya Kingdom imirates Ukraine Uganda Islands Tuvalu Islands venistan Turkai Tobago Tonga Tokelau Togo	84 56 112 52 50 48 44 31 40 36 533 51 32 28		Antigua	Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahamas Azerbaijan Austria Aruba Armenia Argentina and Barbuda		
704 862 548 860 858 840 826 784 800 581 798 795 792 788 780 772 768 626 764	Un US Tu	Virgi Wited State United A Minor Out Wirks and Ca Trinid	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo	nds, US (iet Nam cpublic) Vanuatu cekistan Uruguay America Kingdom cmirates Ukraine Uganda Islands Islands Islands Islands I Tobago Tomga Tokalau Toga Tokalau Toga Tokalau Toga	84 56 112 52 50 48 44 31 40 36 533 51 32 28 10		Antigua	Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahamas Azerbaijan Austria Aruba Armenia Argentina and Barbuda Antarctica		
704 862 548 860 858 840 804 804 806 581 796 795 795 792 788 780 776 772 768 626 764 834	Un US Tu	Virgi gg(Bolivar mited Stat United A Minor Out	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo T d Repu	nds, US (iet Nam cpublic) Vanuatu cekistan Uruguay America Kingdom cmirates Ukraine Uganda Islands Tuvalu Islands Turkey Tunisia Tobago Tonga Tonga Togo cr-Leste Chailand blic of	84 56 112 52 50 48 44 31 40 36 533 51 32 28 10 660		Antigua	Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Astraia Austraia Austraia Armenia Armenia Angentina and Barbuda Antarctica Anguilla		
704 862 548 860 858 840 804 804 806 581 796 795 795 792 788 780 776 772 764 626 764 834 762	Un US Tu	Virgi Wited State United A Minor Out Wirks and Ca Trinid	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo T d Repu	nds, US (iet Nam cpublic) Vanuatu cekistan Uruguay America Kingdom cmirates Ukraine Uganda Islands Tuvalu Islands Turkey Tunisia I Tobago Tonga Toga Toga Cor-Leste chailand cof ikistan	84 56 112 52 50 48 44 31 40 36 533 51 32 28 10 660 24		Antigua	Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahamas Azerbaijan Austria Aruba Armenia Argentina and Barbuda Antarctica Anguilla Angola		
704 862 548 860 858 840 826 784 804 800 581 796 795 772 788 780 776 772 768 626 764 834 762	Un US Tu	Virging (Bolivar, aited State U) United A Minor Out urks and C Trinid	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo T d Repu	nds, US /iet Nam :public) Vanuatu ekistan Uruguay America Kingdom imirates Ukraine Uganda Islands Tuvalu Islands Turkey Tunisia I Tobago Tonga Tokelau Togo or-Leste hailand iblic of ikistan	84 56 1112 52 50 48 44 31 40 36 533 51 32 28 10 660 24 20			Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahamas Azerbaijan Austria Aruba Armenia Argentina and Barbuda Antarctica Anguilla Angola Andorra		
704 862 548 860 858 840 826 784 800 581 798 796 795 772 768 626 764 834 762 	Un US Tu	Virging (Bolivar, aited State U) United A Minor Out urks and C Trinid	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo Taj Bouvet	nds, US /iet Nam :public) Vanuatu ekistan Uruguaya Kingdom imirates Ukraine Uganda Islands Tuvalu Islands ienistan Turkey Tunisia I Tobago Tonga Tokelau Togo or-Leste 'hailand iblic of ikistan	84 56 1112 52 50 48 44 31 40 36 533 51 32 28 10 660 24 20 16			Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahamas Azerbaijan Austria Aruba Armenia Argentina and Barbuda Antarctica Anguilla Angola Andorra		
704 862 548 860 858 8400 581 798 796 772 768 626 764 834 762	Un US Tu	Virging (Bolivar, aited State U) United A Minor Out urks and C Trinid	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo Taj Bouvet	nds, US /iet Nam :public) Vanuatu ekistan Uruguay America Kingdom imirates Ukraine Uganda Islands Tuvalu Islands Turkey Tunisia I Tobago Tonga Tokelau Togo or-Leste hailand iblic of ikistan	84 56 112 52 50 48 44 31 40 36 533 51 32 28 10 660 24 20 16 12			Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahamas Azerbaijan Austria Australia Armenia Armenia Argentina and Barbuda Antarctica Anguilla Andorra crican Samoa Algeria		
704 862 548 860 858 840 826 784 804 809 581 796 795 792 768 626 764 834 834 762 	Us Tu Tanzan	Virging (Bolivar, aited State U) United A Minor Out urks and C Trinid	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo T d Repu Taj Bouvet B	inds, US (iet Nam ipublic) Vanuatu pekistan Uruguaya Kingdom imirates Ukraine Uganda Islands Tuvalu Islands inirates Tuvalu Islands Tobago Tonga Tokelau Togo or-Leste challand blic of ikistan islanda	84 56 1112 52 50 48 48 44 31 40 36 533 51 32 28 10 660 24 20 16 12 8		Ame	Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahrain  Austria Aruba Armenia Argentina and Barbuda Antarctica Anguilla Angola Andorra rican Samoa Algeria Albania		
704 862 548 860 858 840 826 784 804 809 581 796 795 792 768 626 764 834 764 772 768	Us Tu Tanzan	Virging (Bolivar) Wited State United A Minor Out Wrks and C Trinid	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo T d Repu Taj Bouvet B d Herz	inds, US (iet Nam ipublic) Vanuatu pekistan Uruguaya Kingdom imirates Ukraine Uganda Islands Tuvalu Islands inirates Tuvalu Islands Tobago Tonga Tokelau Togo or-Leste challand blic of ikistan islanda	84 56 1112 52 50 48 44 31 40 36 533 51 32 28 10 660 24 20 16 12 8 248		Ame	Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahamas Azerbaijan Austria Aruba Armenia Argentina and Barbuda Antarctica Anguilla Anguilla Angola Andorra rican Samoa Algeria Albania and Islands		
704 862 548 860 858 840 826 784 804 800 581 798 795 792 788 626 764 834 762 772 768 626 764 834 762 772	Us Tu Tanzan	Virging (Bolivar) Wited State United A Minor Out Wrks and C Trinid	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo T d Repu Taj Bouvet B d Herz	nds, US Yiet Nam Public) Vanuatu Purguay America Kingdom Purguad Uganda Islands Islands Pursial Islands Pursial Iobago Tonga Tokela Iogo Tokela Iogo Tokela Iogo Tokela Iogo Iogo Iogo Iogo Iokela Iogo Iogo Iogo Iogo Iogo Iogo Iogo Iog	84 56 1112 52 50 48 48 44 31 40 36 533 51 32 28 10 660 24 20 16 12 8		Ame	Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahrain  Austria Aruba Armenia Argentina and Barbuda Antarctica Anguilla Angola Andorra rican Samoa Algeria Albania		
704 862 548 860 858 840 826 784 800 581 798 795 792 788 780 776 772 768 834 762 774 772 770 68	Us Tu Tanzan	Virging (Bolivar) Wited State United A Minor Out Wrks and C Trinid	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo T d Repu Taj Bouvet B d Herz	nds, US (iet Nam cpublic) Vanuatu ekistan Uruguay America Kingdom imirates Ukraine Uganda Islands Tuvalu Islands Turkey Tunisia I Tobago Tonga Toga Toga Cor-Leste chailand iblic of ikistan : Island ootswana Bolivia Bhutan	84 56 112 52 50 48 44 31 40 36 533 51 32 28 10 660 24 20 16 12 8 248 4	v 1 columns!	Ame	Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahamas Azerbaijan Austria Aruba Armenia Argentina and Barbuda Antarctica Anguilla Anguilla Angola Andorra rican Samoa Algeria Albania and Islands		
704 862 548 860 858 840 826 784 800 581 796 795 792 788 780 776 772 768 626 764 834 762  74 72 70 68 64 60	Us Tu Tanzan	Virging (Bolivar) Wited State United A Minor Out Wrks and C Trinid	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo T d Repu Taj Bouvet B d Herz	nds, US Yiet Nam Public) Vanuatu Wekistan Uruguay America Kingdom Imirates Ukraine Uganda Islands Tuvalu Islands Tuvalu Islands Tobago Tonga Tokelau Togo Or-Leste Chailand Iblic of ikistan Islands I	84 56 112 52 50 48 44 31 40 36 533 51 32 28 10 660 24 20 16 12 8 248 4	x 1 columns]	Ame Al	Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahrain  Austria Aruba Armenia Argentina and Barbuda Antarctica Anguilla Andorra rican Samoa Algeria Albania and Islands Afghanistan		
704 862 548 860 858 840 806 784 800 581 796 795 792 788 780 776 772 768 834 762  74 72 70 68 64	Us Tu Tanzan	Virging (Bolivar) Wited State United A Minor Out Wrks and C Trinid	n Isla V ian Re Uzb es of nited Arab E lying aicos Turkm ad and Timo T d Repu Taj Bouvet B d Herz	nds, US (iet Nam cpublic) Vanuatu ekistan Uruguay America Kingdom imirates Ukraine Uganda Islands Tuvalu Islands Turkey Tunisia I Tobago Tonga Toga Toga Cor-Leste chailand iblic of ikistan : Island ootswana Bolivia Bhutan	84 56 112 52 50 48 44 31 40 36 533 51 32 28 10 660 24 20 16 12 8 248 4	x 1 columns]	Ame Al	Belize Belgium Belarus Barbados Bangladesh Bahrain  Bahrain  Bahrain  Austria Aruba Armenia Argentina and Barbuda Antarctica Anguilla Andorra rican Samoa Algeria Albania and Islands Afghanistan		

# d) MySql Database to HORUS Format

import pandas as pd
import sqlite3 as sq
# Input Agreement ====================================
sInputFileName='C:/VKHCG/05-DS/9999-Data/utility.db'
sInputTable='Country_Code'
conn = sq.connect(sInputFileName)
sSQL='select * FROM ' + sInputTable + ';'
InputData=pd.read_sql_query(sSQL, conn)
print('Input Data Values =========')
print(InputData)
print('==========')
# Processing Rules ====================================
ProcessData=InputData
# Remove columns ISO-2-Code and ISO-3-CODE
ProcessData.drop('ISO-2-CODE', axis=1,inplace=True)
ProcessData.drop('ISO-3-Code', axis=1,inplace=True)
# Rename Country and ISO-M49
ProcessData.rename(columns={'Country': 'CountryName'}, inplace=True)
ProcessData.rename(columns={'ISO-M49': 'CountryNumber'}, inplace=True)
# Set new Index
ProcessData.set_index('CountryNumber', inplace=True)
# Sort data by CurrencyNumber
ProcessData.sort_values('CountryName', axis=0, ascending=False, inplace=True)
print('Process Data Values =========')
print(ProcessData)
print('=============') print('=============') # Output Agreement ====================================
# Output Agreement ====================================
OutputData=ProcessData
sOutputFileName='C:/VKHCG/05-DS/9999-Data/HORUS-CSV-Country.csv'
OutputData.to_csv(sOutputFileName, index = False)
print('Database to HORUS - Done')
# Utility done ====================================

### **Output:**

```
In [6]: runfile('C:/VKHCG/05-DS/9999-Data/DATABASE2HORUS.py', wdir='C:/VKHCG/05-DS/9999-Data')
Input Data Values ===========
    index
                                Country ISO-2-CODE ISO-3-Code ISO-M49
                                           AF
                                                   AFG
                             Afghanistan
0
1
                           Aland Islands
                                                AX
                                                         ALA
                                                                  248
                                Albania
2
3
                                Algeria
                                               DZ
                                                        DZA
                                                                  12
        3
                                              AS
                                                        ASM
4
                          American Samoa
                                                                  16
                                Andorra
5
        5
                                               AD
                                                        AND
                                                                  20
6
        6
                                  Angola
                                               AO
                                                        AGO
                                                                  24
                                Anguilla
                                                        AIA
                                                                 660
8
                              Antarctica
                                                         ATA
                                                                  10
                                               AQ
        9
                     Antigua and Barbuda
                                               AG
                                                        ATG
                                                                  28
9
                               Argentina
10
       10
                                               AR
                                                         ARG
                                                                  32
                                                         ΔRM
                                               AM
11
       11
                                Armenia
                                                                  51
                                                         ABW
12
       12
                                  Aruba
                                                AW
                                                                 533
13
       13
                               Australia
                                                ΑU
                                                         AUS
                                                                  36
14
                                Austria
                                                         AUT
15
       15
                              Azerbaijan
                                               ΑZ
                                                         AZE
                                                                  31
                                                         BHS
16
       16
                                 Bahamas
                                               BS
                                                                  44
                                                         BHR
17
       17
                                 Bahrain
                                               BH
                                                                  48
18
       18
                              Bangladesh
                                               BD
                                                         BGD
                                                                  50
19
       19
                                Barbados
                                               ВВ
                                                         BRB
                                                                  52
20
       20
                                 Belarus
                                                BY
                                                         BLR
                                                                 112
21
       21
                                 Belgium
                                               BE
                                                         BEL
                                                                  56
22
       22
                                  Belize
                                                BZ
                                                         BLZ
                                                                  84
23
                                  Benin
                                               вл
                                                         BEN
                                                                 204
       23
                                                         BMU
24
       24
                                 Bermuda
                                               BM
                                                                  60
25
       25
                                  Bhutan
                                               ВТ
                                                         BTN
                                                                  64
26
                                 Bolivia
                                                         BOL
                                                                  68
27
                   Bosnia and Herzegovina
                                                         BIH
                                                                  70
28
       28
                                Botswana
29
       29
                              Bouvet Island
                                                    ΒV
                                                              BVT
                                                                        74
..
217
                                                              TJK
                                 Tajikistan
       217
                                                                       762
              Tanzania, United Republic of
218
       218
                                                    T7
                                                              TZA
                                                                       834
219
                                                              THA
                                                                       764
       219
                                   Thailand
                                                    TH
                                Timor-Leste
220
       220
                                                              TLS
                                                                       626
                                                    TL
221
       221
                                                              TGO
                                                                       768
                                                    TG
                                       Togo
222
       222
                                    Tokelau
                                                    ΤK
                                                              TKL
                                                                       772
223
       223
                                                    то
                                                              TON
                                      Tonga
224
                      Trinidad and Tobago
       224
       225
                                     Turkey
226
       226
                                                    TR
                                                              TUR
                                                                       792
227
       227
                               Turkmenistan
                                                    TM
                                                              TKM
                                                                       795
                Turks and Caicos Islands
228
       228
                                                    TC
                                                              TCA
                                                                       796
229
       229
                                     Tuvalu
                                                    TV
                                                              TUV
                                                                       798
230
       230
                                     Uganda
                                                    UG
                                                              UGA
                                                                       800
231
       231
                                    Ukraine
                                                    UA
                                                              UKR
                                                                       804
232
       232
                      United Arab Emirates
                                                    AE
                                                              ARE
                                                                       784
233
       233
                            United Kingdom
                                                    GB
                                                              GBR
                                                                       826
234
       234
                   United States of America
                                                    US
                                                              USA
235
       235
                  US Minor Outlying Islands
                                                    UM
                                                                       581
                                    Uruguay
237
       237
                                 Uzbekistan
                                                    UΖ
                                                              UZB
                                                                       860
238
       238
                                    Vanuatu
                                                    VU
                                                              VUT
                                                                       548
239
       239 Venezuelaÿ(Bolivarian Republic)
                                                    VE
                                                              VEN
                                                                       862
240
       240
                                   Viet Nam
                                                    VN
                                                              VNM
                                                                       704
                        Virgin Islands, US
                                                              VIR
                                                                       850
241
       241
                                                    VT
                  Wallis and Futuna Islands
242
       242
                                                                       876
                                                    WF
                                                              WLF
243
                                                    EΗ
                                                              ESH
       243
                            Western Sahara
                                                                       732
244
                                                              YEM
       244
                                      Yemen
                                                    YΕ
                                                                       887
245
       245
                                     Zambia
                                                              ZMB
                                                                       894
                                                    ZM
                                   Zimbabwe
                                                                       716
[247 rows x 5 columns]
                      _____
```

14

Process Data \	/alues =				
	index	CountryName			
CountryNumber			70	27	Bosnia and Herzegovina
716	246	Zimbabwe	68	26	Bolivia
894	245	Zambia	64	25	Bhutan
887	244	Yemen	60	24	Bermuda
732	243	Western Sahara	204	23	Benin
876	242	Wallis and Futuna Islands	84	22	Belize
850	241	Virgin Islands, US	56	21	Belgium
704	240	Viet Nam	112	20	Belarus
862	239	Venezuelaÿ(Bolivarian Republic)	52	19	Barbados
548	238	Vanuatu	50	18	Bangladesh
860 858	237 236	Uzbekistan	48	17	Bahrain
840	234	Uruguay United States of America	44	16	Bahamas
826	234	United Kingdom	31	15	Azerbaijan
784	232	United Arab Emirates	40	14	Austria
804	232	Ukraine	36	13	Australia
800	230	Uganda	533	12	Aruba
581	235	US Minor Outlying Islands			
798	229	Tuvalu	51	11	Armenia
796	228	Turks and Caicos Islands	32	10	Argentina
795	227	Turkmenistan	28	9	Antigua and Barbuda
792	226	Turkey	10	8	Antarctica
788	225	Tunisia	660	7	Anguilla
780	224	Trinidad and Tobago	24	6	Angola
776	223	Tonga	20	5	Andorra
772	222	Tokelau	16	4	American Samoa
768	221	Togo	12	3	Algeria
626	220	Timor-Leste	8	2	Albania
764	219	Thailand	248	1	Aland Islands
834	218	Tanzania, United Republic of	4	0	Afghanistan
762	217	Tajikistan			
2	• • • • • • • • • • • • • • • • • • • •		[247 row	vs x 2 columns]	
74	29	Bouvet Island	=======		
72	28	Botswana	Database	to HORUS - Done	
70	27	Bosnia and Herzegovina			

# e) Picture (JPEG) to HORUS Format

# Utility Start Picture to HORUS ====================================
from scipy.misc import imread
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
# Input Agreement ====================================
sInputFileName='C:/VKHCG/05-DS/9999-Data/Angus.jpg'
InputData = imread(sInputFileName, flatten=False, mode='RGBA')
print('Input Data Values =========')
<pre>print('X: ',InputData.shape[0])</pre>
<pre>print('Y: ',InputData.shape[1])</pre>
<pre>print('RGBA: ', InputData.shape[2])</pre>
print('======')
# Processing Rules ====================================
ProcessRawData=InputData.flatten()
y=InputData.shape[2] + 2
x=int(ProcessRawData.shape[0]/y)
ProcessData=pd.DataFrame(np.reshape(ProcessRawData, (x, y)))
sColumns= ['XAxis','YAxis','Red', 'Green', 'Blue','Alpha']
ProcessData.columns=sColumns
ProcessData.index.names =['ID']
<pre>print('Rows: ',ProcessData.shape[0])</pre>
print('Columns:',ProcessData.shape[1])
print('======')
print('Process Data Values =======')
print('======')
plt.imshow(InputData)
plt.show()
print('======')
# Output Agreement ====================================
OutputData=ProcessData
print('Storing File')
sOutputFileName='C:/VKHCG/05-DS/9999-Data/HORUS-Picture.csv'
OutputData.to_csv(sOutputFileName, index = False)
print('======')
print('Picture to HORUS - Done')
print('======')

### f) i. Video to HORUS Format

```
Movies to Frames
import os
import shutil
import cv2
#======
sInputFileName='C:/VKHCG/05-DS/9999-Data/dog.mp4'
sDataBaseDir='C:/VKHCG/05-DS/9999-Data/temp'
if os.path.exists(sDataBaseDir):
shutil.rmtree(sDataBaseDir)
if not os.path.exists(sDataBaseDir):
os.makedirs(sDataBaseDir)
print('========')
print('Start Movie to Frames')
print('=======')
vidcap = cv2.VideoCapture(sInputFileName)
success,image = vidcap.read()
count = 0
while success:
success,image = vidcap.read()
sFrame=sDataBaseDir + str('/dog-frame-' + str(format(count, '04d'))+ '.jpg')
print('Extracted: ', sFrame)
cv2.imwrite(sFrame, image)
if os.path.getsize(sFrame) == 0:
count += -1
os.remove(sFrame)
print('Removed: ', sFrame)
if cv2.waitKey(10) == 27: # exit if Escape is hit
break
count += 1
print('=======')
print('Generated : ', count, ' Frames')
print('=======')
print('Movie to Frames HORUS - Done')
print('=======')
```

# **Output:**

```
05-DS/9999-Data')
Start Movie to Frames
______
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0000.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0001.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0002.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0003.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0004.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0005.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0006.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0007.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0008.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0009.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0010.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0011.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0012.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0013.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0014.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0015.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0016.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0017.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0018.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0019.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0020.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0021.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0022.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0023.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0024.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0025.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0026.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0027.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0028.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0079.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0080.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0081.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0082.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0083.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0084.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0085.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0086.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0087.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0088.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0089.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0090.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0091.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0092.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0093.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0094.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0095.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0096.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0097.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0098.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0099.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0100.jpg
Extracted: C:/VKHCG/05-DS/9999-Data/temp/dog-frame-0101.jpg
_____
Generated: 101 Frames
_____
Movie to Frames HORUS - Done
```

In [72]: runfile('C:/VKHCG/05-DS/9999-Data/MOVIE2HORUSFrame.py', wdir='C:/VKHCG/

#### ii. Frames to Horus

```
from scipy.misc import imread
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import os
sDataBaseDir='C:/VKHCG/05-DS/9999-Data/temp'
for file in os.listdir(sDataBaseDir):
if file.endswith(".jpg"):
f += 1
sInputFileName=os.path.join(sDataBaseDir, file)
print('Process : ', sInputFileName)
InputData = imread(sInputFileName, flatten=False, mode='RGBA')
print('Input Data Values =========')
print('X: ',InputData.shape[0])
print('Y: ',InputData.shape[1])
print('RGBA: ', InputData.shape[2])
print('=======')
ProcessRawData=InputData.flatten()
y=InputData.shape[2] + 2
x=int(ProcessRawData.shape[0]/y)
ProcessFrameData=pd.DataFrame(np.reshape(ProcessRawData, (x, y)))
ProcessFrameData['Frame']=file
print('=======')
print('Process Data Values =======')
print('=======')
plt.imshow(InputData)
plt.show()
if f == 1:
ProcessData=ProcessFrameData
ProcessData=ProcessData.append(ProcessFrameData)
if f > 0:
sColumns= ['XAxis', 'YAxis', 'Red', 'Green', 'Blue', 'Alpha', 'FrameName']
ProcessData.columns=sColumns print('===========')
ProcessFrameData.index.names =['ID']
print('Rows: ',ProcessData.shape[0])
print('Columns :',ProcessData.shape[1])
print('=======')
```

## g) Audio to HORUS Format

```
Code:
# Standard Tools
#_____
from scipy.io import wavfile
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
def show_info(aname, a,r):
print ('----')
print ("Audio:", aname)
print ('----')
print ("Rate:", r)
print ('----')
print ("shape:", a.shape)
print ("dtype:", a.dtype)
print ("min, max:", a.min(), a.max())
print ('----')
plot_info(aname, a,r)
def plot info(aname, a,r):
sTitle= 'Signal Wave - '+ aname + ' at ' + str(r) + 'hz'
plt.title(sTitle)
sLegend=[]
for c in range(a.shape[1]):
sLabel = 'Ch' + str(c+1)
sLegend=sLegend+[str(c+1)]
plt.plot(a[:,c], label=sLabel)
plt.legend(sLegend)
plt.show()
sInputFileName='C:/VKHCG/05-DS/9999-Data/2ch-sound.wav'
print('=======')
print('Processing : ', sInputFileName)
print('=======')
InputRate, InputData = wavfile.read(sInputFileName)
show_info("2 channel", InputData,InputRate)
ProcessData=pd.DataFrame(InputData)
sColumns=['Ch1','Ch2']
ProcessData.columns=sColumns
OutputData=ProcessData
sOutputFileName='C:/VKHCG/05-DS/9999-Data/HORUS-Audio-2ch.csv'
OutputData.to_csv(sOutputFileName, index = False)
```



## **Practical 3**

# **Utilities and Auditing**

#### A. Fixers Utilities:

print('>',cleandata,'<')</pre>

# 

### # 2 Removing nonprintable characters from a data entry

```
print('#2 Removing nonprintable characters from a data entry')
printable = set(string.printable)
baddata = "Data\x00Science with\x02 funny characters is \x10bad!!!"
cleandata=".join(filter(lambda x: x in string.printable,baddata))
print('Bad Data: ',baddata);
print('Clean Data: ',cleandata)
```

### #3 Reformatting data entry to match specific formatting criteria.

```
# Convert YYYY/MM/DD to DD Month YYYY
print('# 3 Reformatting data entry to match specific formatting criteria.')
baddate = dt.date(2019, 10, 31)
baddata=format(baddate,'%Y-%m-%d')
gooddate = dt.datetime.strptime(baddata,'%Y-%m-%d')
gooddata=format(gooddate,'%d %B %Y')
print('Bad Data: ',baddata)
print('Good Data: ',gooddata)
```

```
In [10]: runfile('C:/Users/MSCIT/.spyder-py3/temp.py', wdir='C:/Users/MSCIT/.spyder-py3')

#1 Removing leading or lagging spaces from a data entry
> Data Science with too many spaces is bad!!! <
> Data Science with too many spaces is bad!!! <

#2 Removing nonprintable characters from a data entry
Bad Data: Data Science with funny characters is †bad!!!
Clean Data: DataScience with funny characters is bad!!!

# 3 Reformatting data entry to match specific formatting criteria.
Bad Data: 2019-10-31
Good Data: 31 October 2019
```

# **B.** Averaging of Data

#### Code:

import pandas as pd InputFileName='IP DATA CORE.csv' OutputFileName='Retrieve\_Router\_Location.csv' Base='C:/VKHCG' print('##############") print('Working Base:',Base, 'using') print('#############") sFileName=Base + '/01-Vermeulen/00-RawData/' + InputFileName print('Loading :',sFileName) IP\_DATA\_ALL=pd.read\_csv(sFileName,header=0,low\_memory=False, usecols=['Country','Place Name','Latitude','Longitude'], encoding="latin-1") IP DATA ALL.rename(columns={'Place Name': 'Place Name'}, inplace=True) AllData=IP\_DATA\_ALL[['Country', 'Place\_Name','Latitude']] print(AllData) MeanData=AllData.groupby(['Country', 'Place\_Name'])['Latitude'].mean() print(MeanData)

3561	DE	Munich	1 48.1480
[3562	rows x 3 (	olumns]	
Count	ry Place_M	lame	
DE	Munich		48.143223
GB	London		51.509406
US	New Yor	rk	40.747044
Name:	Latitude,	dtype:	float64

### **D. Outlier Detection**

```
Code:
```

```
# -*- coding: utf-8 -*-
import pandas as pd
InputFileName='IP_DATA_CORE.csv'
OutputFileName='Retrieve Router Location.csv'
Base='C:/VKHCG'
print('##############")
print('Working Base :',Base)
print('##############")
sFileName=Base + '/01-Vermeulen/00-RawData/' + InputFileName
print('Loading :',sFileName)
IP DATA ALL=pd.read csv(sFileName,header=0,low memory=False,
usecols=['Country','Place Name','Latitude','Longitude'], encoding="latin-1")
IP DATA ALL.rename(columns={'Place Name': 'Place Name'}, inplace=True)
LondonData=IP_DATA_ALL.loc[IP_DATA_ALL['Place_Name']=='London']
AllData=LondonData[['Country', 'Place_Name','Latitude']]
print('All Data')
print(AllData)
MeanData=AllData.groupby(['Country', 'Place_Name'])['Latitude'].mean()
StdData=AllData.groupby(['Country', 'Place Name'])['Latitude'].std()
print('Outliers')
UpperBound=float(MeanData+StdData)
print('Higher than ', UpperBound)
OutliersHigher=AllData[AllData.Latitude>UpperBound]
print(OutliersHigher)
LowerBound=float(MeanData-StdData)
print('Lower than ', LowerBound)
OutliersLower=AllData[AllData.Latitude<LowerBound]
print(OutliersLower)
print('Not Outliers')
OutliersNot=AllData[(AllData.Latitude>=LowerBound) & (AllData.Latitude<=UpperBound)]
print(OutliersNot)
```

<pre>In [35]: runfile('C:/Users/MSCIT/.spyder-py3/temp.py',</pre>	[1502 rows x 3 columns] Outliers
wdir='C:/Users/MSCIT/.spyder-py3')	Higher than 51.51263550786781
	Country Place Name Latitude
	1910 GB London 51.5130
	1911 GB London 51.5508
***************************************	1912 GB London 51.5649
Working Base : C:/VKHCG	1913 GB London 51.5895
***************************************	1914 GB London 51.5232
Loading : C:/VKHCG/01-Vermeulen/00-RawData/IP DATA CORE.csv	1916 GB London 51.5491
All Data	1919 GB London 51.5161
	1920 GB London 51.5198
Country Place_Name Latitude	1921 GB London 51.5198
1910 GB London 51.5130	1923 GB London 51.5237
1911 GB London 51.5508	1924 GB London 51.5237
1912 GB London 51,5649	1925 GB London 51.5237
	1926 GB London 51.5237
1913 GB London 51.5895	1927 GB London 51.5232
1914 GB London 51.5232	3436 GB London 51.5163
1915 GB London 51.4739	3438 GB London 51.5136
1916 GB London 51.5491	Lower than 51.50617687562166
	Country Place_Name Latitude
1917 GB London 51.5085	1915 GB London 51.4739

Not Outliers								
	Country	Place_Name	Latitude					
1917	GB	London	51.5085					
1918	GB	London	51.5085					
1922	GB	London	51.5085					
1928	GB	London	51.5085					
1929	GB	London	51.5085					
1957	GB	London	51.5115					
1958	GB	London	51.5092					
1959	GB	London	51.5092					
1960	GB	London	51.5092					
1961	GB	London	51.5092					
1962	GB	London	51.5092					

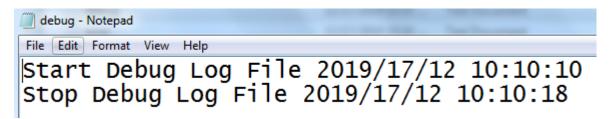
# E. Logging

```
Code:
import sys
import os
import logging
import uuid
import shutil
   import time
if sys.platform == 'linux':
Base=os.path.expanduser('~') + '/VKHCG'
else:
   Base='C:/VKHCG'
sCompanies=['01-Vermeulen','02-Krennwallner','03-Hillman','04-Clark']
sLayers=['01-Retrieve','02-Assess','03-Process','04-Transform','05-Organise','
06-Report']
   sLevels=['debug','info','warning','error']
for sCompany in sCompanies:
sFileDir=Base + '/' + sCompany
if not os.path.exists(sFileDir):
os.makedirs(sFileDir)
for sLayer in sLayers:
log = logging.getLogger() # root logger
for hdlr in log.handlers[:]: # remove all old handlers
log.removeHandler(hdlr)
sFileDir=Base + '/' + sCompany + '/' + sLayer + '/Logging'
if os.path.exists(sFileDir):
shutil.rmtree(sFileDir)
time.sleep(2)
if not os.path.exists(sFileDir):
os.makedirs(sFileDir)
skey=str(uuid.uuid4())
sLogFile=Base + '/' + sCompany + '/' + sLayer + '/Logging/
Logging_'+skey+'.log'
print('Set up:',sLogFile)
logging.basicConfig(level=logging.DEBUG,
format='%(asctime)s %(name)-12s %(levelname)-8s %(message)s',
datefmt='%m-%d %H:%M'.
filename=sLogFile,
filemode='w')
console = logging.StreamHandler()
console.setLevel(logging.INFO)
formatter = logging.Formatter('%(name)-12s: %(levelname)-8s %(message)s')
console.setFormatter(formatter)
logging.getLogger(").addHandler(console)
```

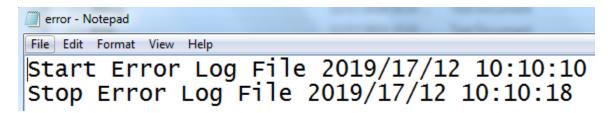
```
logging.info('Practical Data Science is fun!.')
for sLevel in sLevels:
sApp='Application-'+ sCompany + '-' + sLayer + '-' + sLevel
logger = logging.getLogger(sApp)
if sLevel == 'debug':
logger.debug('Practical Data Science logged a debugging message.')
if sLevel == 'info':
logger.info('Practical Data Science logged information message.')
if sLevel == 'warning':
logger.warning('Practical Data Science logged a warning message.')
if sLevel == 'error':
logger.error('Practical Data Science logged an error message.')
```

0	Retrieve-IP_D	_	× IP_D	ATA_ALL_with_ID ×			
<b>(</b>		Filter		<u> </u>		Q,	
_	RowID	X1 =	Country	Place.Name	Post.Code	Latitude	Longitude
1	1	1	BW	Gaborone	NA	-24.6464	25.9119 💣
2	2	2	BW	Gaborone	NA	-24.6464	25.9119
3	3	3	BW	Gaborone	NA	-24.6464	25.9119
4	4	4	BW	Gaborone	NA	-24.6464	25.9119
5	5	5	BW	Gaborone	NA	-24.6464	25.9119
6	6	6	BW	Gaborone	NA	-24.6464	25.9119
7	7	7	BW	Gaborone	NA	-24.6464	25.9119
8	8	8	BW	Gaborone	NA	-24.6464	25.9119
9	9	9	BW	Gaborone	NA	-24.6464	25.9119
10	10	10	BW	Gaborone	NA	-24.6464	25.9119
11	11	44	DIAI	Cabarana	NIA	24 6464	25 0110
Show Con	_	f 1,247,50	2 entries, 9 tota	l columns			-0
+ + + + + + + + + + + + + + + + + + +	fil rite(pas fil rite(pas fil	fo e=debu teO('S fo e=info teO('S fo e=erro ######	ormat(Stop gLog,appe stop Infor ormat(Stop Log,appen stop Error ormat(Stop orLog,appe	Log File ' Time, "%Y/% nd = TRUE) ###########	File ', d/%m %H:%M: , d/%m %H:%M:	%s")),	<i>∆</i>

#### Debug.txt



#### Error.txt



#### Info.txt

```
info-Notepad

File Edit Format View Help

Start Information Log File 2019/17/12 10:10:10

Practical Data Scientist Load library: readr

Practical Data Scientist Load library: data.table

Practical Data Scientist Load library: tibble

Practical Data Scientist Retrieve data file: C:/VKHCG/01-Vermeulen/00

-RawData/IP_DATA_ALL.csv

Practical Data Scientist Stores Retrieve data file: C:/VKHCG/01-Vermeulen/01-Retrieve/01-EDS/01-R/Retrieve_IP_DATA.csv

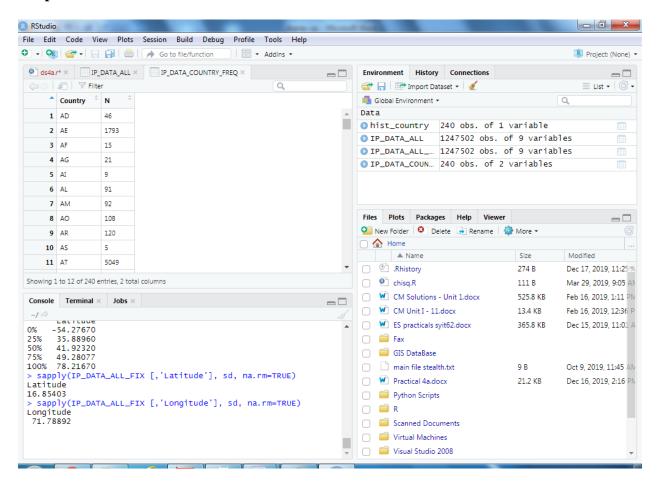
Stop Information Log File 2019/17/12 10:10:18
```

# **Practical 4**

# **Retrieve Superstep**

# A. Perform the following data processing using R.

```
library(readr)
IP DATA ALL <- read csv("C:/VKHCG/01-Vermeulen/00-RawData/IP DATA ALL.csv")
View(IP DATA ALL)
spec(IP DATA ALL)
IP DATA ALL FIX=set tidy names(IP DATA ALL, syntactic = TRUE, quiet = TRUE)
library(tibble)
set tidy names(IP DATA ALL, syntactic = TRUE, quiet = FALSE)
sapply(IP_DATA_ALL_FIX, typeof)
library(data.table)
hist_country=data.table(Country=unique(IP_DATA_ALL_FIX[is.na(IP_DATA_ALL_FIX
['Country'] = 0, ]$Country
IP_DATA_COUNTRY_FREQ=data.table(with(IP_DATA_ALL_FIX, table(Country)))
View(IP_DATA_COUNTRY_FREQ)
sapply(IP DATA ALL FIX[,'Latitude'], min, na.rm=TRUE)
sapply(IP_DATA_ALL_FIX[,'Country'], min, na.rm=TRUE)
sapply(IP DATA ALL FIX[,'Latitude'], max, na.rm=TRUE)
sapply(IP_DATA_ALL_FIX[,'Country'], max, na.rm=TRUE)
sapply(IP DATA ALL FIX [,'Latitude'], mean, na.rm=TRUE)
sapply(IP_DATA_ALL_FIX [,'Latitude'], median, na.rm=TRUE)
sapply(IP_DATA_ALL_FIX [,'Latitude'], range, na.rm=TRUE)
sapply(IP DATA ALL FIX [,'Latitude'], quantile, na.rm=TRUE)
sapply(IP_DATA_ALL_FIX [,'Latitude'], sd, na.rm=TRUE)
sapply(IP DATA ALL FIX [,'Longitude'], sd, na.rm=TRUE)
```



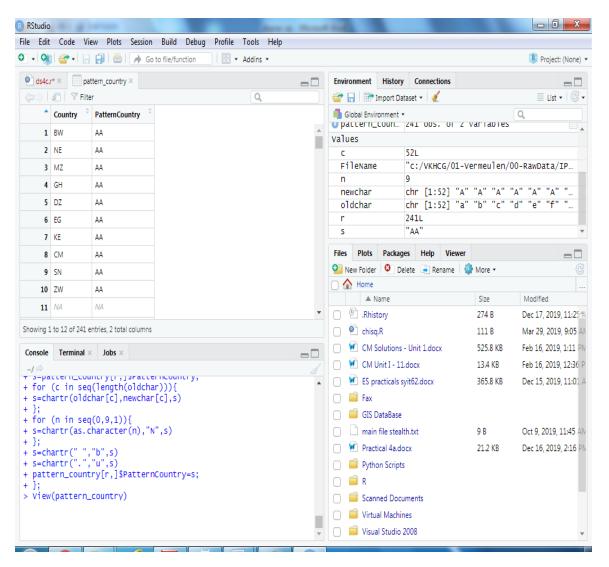
# B. Retrive\_IP\_DATA\_ALL.py

```
Code:
import sys
import os
import pandas as pd
Base='C:/VKHCG'
sFileName=Base + '/01-Vermeulen/00-RawData/IP DATA ALL.csv'
print('Loading :',sFileName)
IP_DATA_ALL=pd.read_csv(sFileName,header=0,low_memory=False, encoding="latin-1")
sFileDir=Base + '/01-Vermeulen/01-Retrieve/01-EDS/02-Python'
if not os.path.exists(sFileDir):
os.makedirs(sFileDir)
print('Rows:', IP_DATA_ALL.shape[0])
print('Columns:', IP DATA ALL.shape[1])
print('### Raw Data Set ###########################")
for i in range(0.len(IP DATA ALL.columns)):
print(IP DATA ALL.columns[i].type(IP DATA ALL.columns[i]))
print('### Fixed Data Set #######################")
IP DATA ALL FIX=IP DATA ALL
for i in range(0,len(IP DATA ALL.columns)):
cNameOld=IP DATA ALL FIX.columns[i] + ' '
cNameNew=cNameOld.strip().replace(" ", ".")
IP_DATA_ALL_FIX.columns.values[i] = cNameNew
print(IP DATA ALL.columns[i],type(IP DATA ALL.columns[i]))
#print(IP DATA ALL FIX.head())
print('Fixed Data Set with ID')
IP DATA ALL with ID=IP DATA ALL FIX
IP DATA ALL with ID.index.names = ['RowID']
#print(IP DATA ALL with ID.head())
sFileName2=sFileDir + '/Retrieve IP DATA.csv'
IP_DATA_ALL_with_ID.to_csv(sFileName2, index = True, encoding="latin-1")
print('### Done!! ####################")
```

```
In [42]: runfile('C:/Users/MSCIT/.spyder-py3/temp.py',
wdir='C:/Users/MSCIT/.spyder-py3')
Loading : C:/VKHCG/01-Vermeulen/00-RawData/IP_DATA_ALL.csv
Unnamed: 0 <class 'str'>
ID <class 'str'>
Country <class 'str'>
Place.Name <class 'str'>
Post.Code <class 'str'>
Latitude <class 'str'>
Longitude <class 'str'>
First.IP.Number <class 'str'>
Last.IP.Number <class 'str'>
Unnamed:.0 <class 'str'>
ID <class 'str'>
Country <class 'str'>
Place.Name <class 'str'>
Post.Code <class 'str'>
Latitude <class 'str'>
Longitude <class 'str'>
First.IP.Number <class 'str'>
Last.IP.Number <class 'str'>
```

## C. Data Pattern

```
library(readr)
library(data.table)
FileName=paste0('c:/VKHCG/01-Vermeulen/00-RawData/IP_DATA_ALL.csv')
IP_DATA_ALL <- read_csv(FileName)</pre>
hist_country=data.table(Country=unique(IP_DATA_ALL$Country))
pattern_country=data.table(Country=hist_country$Country,
PatternCountry=hist_country$Country)
oldchar=c(letters,LETTERS)
newchar=replicate(length(oldchar),"A")
for (r in seq(nrow(pattern_country))){
s=pattern_country[r,]$PatternCountry;
for (c in seq(length(oldchar))){
s=chartr(oldchar[c],newchar[c],s)
};
for (n in seq(0,9,1))
s=chartr(as.character(n),"N",s)
};
s=chartr(" ","b",s)
s=chartr(".","u",s)
pattern_country[r,]$PatternCountry=s;
};
View(pattern_country)
```



# D. Loading IP\_DATA\_ALL

```
# -*- coding: utf-8 -*-
import sys
import os
import pandas as pd
Base='C:/VKHCG'
sFileName=Base + '/01-Vermeulen/00-RawData/IP DATA ALL.csv'
print('Loading :',sFileName)
IP DATA ALL=pd.read csv(sFileName,header=0,low memory=False, encoding="latin-1")
sFileDir=Base + '/01-Vermeulen/01-Retrieve/01-EDS/02-Python'
if not os.path.exists(sFileDir):
os.makedirs(sFileDir)
print('Rows:', IP DATA ALL.shape[0])
print('Columns:', IP DATA ALL.shape[1])
print('### Raw Data Set ############################")
for i in range(0,len(IP DATA ALL.columns)):
print(IP DATA ALL.columns[i].type(IP DATA ALL.columns[i]))
print('### Fixed Data Set ########################")
IP DATA ALL FIX=IP DATA ALL
for i in range(0,len(IP_DATA_ALL.columns)):
cNameOld=IP DATA ALL FIX.columns[i] + ' '
cNameNew=cNameOld.strip().replace(" ", ".")
IP DATA ALL FIX.columns.values[i] = cNameNew
print(IP_DATA_ALL.columns[i],type(IP_DATA_ALL.columns[i]))
#print(IP DATA ALL FIX.head())
print('Fixed Data Set with ID')
IP DATA ALL with ID=IP DATA ALL FIX
IP_DATA_ALL_with_ID.index.names = ['RowID']
#print(IP DATA_ALL_with_ID.head())
sFileName2=sFileDir + '/Retrieve_IP_DATA.csv'
IP DATA ALL with ID.to csv(sFileName2, index = True, encoding="latin-1")
print('### Done!! ####################")
```

#### E. Program to connect to different data sources.

```
SQLite: Code:
```

```
# -*- coding: utf-8 -*-
import sqlite3 as sq
import pandas as pd
Base='C:/VKHCG'
sDatabaseName=Base + '/01-Vermeulen/00-RawData/SQLite/vermeulen.db'
conn = sq.connect(sDatabaseName)
sFileName='C:/VKHCG/01-Vermeulen/01-Retrieve/01-EDS/02-Python/Retrieve_IP_DATA.csv'
print('Loading :',sFileName)
IP_DATA_ALL_FIX=pd.read_csv(sFileName,header=0,low_memory=False)
IP DATA ALL FIX.index.names = ['RowIDCSV']
sTable='IP DATA ALL'
print('Storing :',sDatabaseName,' Table:',sTable)
IP_DATA_ALL_FIX.to_sql(sTable, conn, if_exists="replace")
print('Loading :',sDatabaseName,' Table:',sTable)
TestData=pd.read_sql_query("select * from IP_DATA_ALL;", conn)
print('##########")
print('## Data Values')
print('##########")
print(TestData)
print('##########")
print('## Data Profile')
print('##########")
print('Rows:',TestData.shape[0])
print('Columns:',TestData.shape[1])
print('##########")
print('### Done!! #########################")
```

```
In [9]: runfile('C:/Users/MSCIT/.spyder-py3/temp.py', wdir='C:/Users/
MSCIT/, spyder-py3')
Loading : C:/VKHCG/01-Vermeulen/01-Retrieve/01-EDS/02-Python/Retrieve_IP_DATA.csv
Storing : C:/VKHCG/01-Vermeulen/00-RawData/SQLite/vermeulen.db Table: IP_DATA_ALL
Loading : C:/VKHCG/01-Vermeulen/00-RawData/SQLite/vermeulen.db Table: IP_DATA_ALL
Data Values
*************
                  RowID
       ROWIDCSV
                            ID ... Longitude First.IP.Number Last.IP.Number
                     0
                              1 ...
                                      25.9119
                                                692781056
                                       25.9119
1
                      1
                              2 ...
                                                   692781824
                                                                 692783103
              1
             2
                                                   692909056
                                                                 692909311
2
                                      25.9119
                      2
                              3 ...
                                                   692909568
                                                                 692910079
3
             3
                      3
                              4 ...
                                       25.9119
             4
                     4
                             5 ... 25.9119
                                                   693051392
                                                                 693052415
             5
                      5
                              6 ...
                                       25,9119
                                                   693078272
                                                                 693078527
                             7 ...
                                      25.9119
                                                   693608448
6
             6
                     6
                                                                 693616639
             7
                     7
                                                   696929792
                                                                 696930047
                              8 ...
                                      25,9119
                    8
                                                                 700439039
8
             8
                              9 ...
                                       25.9119
                                                   700438784
                     9
                            10 ...
             9
                                       25.9119
                                                   702075904
                                                                 782876927
                            11 ...
                    10
             10
10
                                       25.9119
                                                   702498816
                                                                 702499839
            11
                   11
                                      25.9119
                                                   702516224
                                                                 702517247
11
                            13 ...
14 ...
                    12
12
             12
                                       25,9119
                                                   774162663
                                                                 774162667
                                       25.9119
                                                1401887232
                                                                1401887743
13
             13
                    14
                            15 ... 25.9119
16 ... 2.1167
                                                1754209024
14
             14
                                                                1754209279
15
             15
                     15
                                                   696918528
                                                                 696919839
                            17 ...
            16
                    16
                                        2.1167
                                                   696922112
                                                                 696924159
                            18 ...
19 ...
17
             17
                     17
                                        2.1167
                                                   701203456
                                                                 701203711
18
             18
                    18
                                        2.1167
                                                   758886912
                                                                 758887167
                            20 ...
19
             19
                     19
                                        2.1167
                                                  1347294153
                                                                1347294168
20
             20
                     20
                                        2.1167
                                                  1755108096
                                                                1755108351
                            22 ...
23 ...
24 ...
21
             21
                     21
                                       2.1167
                                                  1755828480
                                                                1755828735
22
             22
                     22
                                       32.5892
                                                   692883456
                                                                 692883967
                     23
                                       32.5892
                                                    692944896
23
             23
                                                                 692946943
                            25 ...
26 ...
             24
                                                    696967168
                     24
                                       32,5892
                                                                 696971263
24
25
             25
                     25
                                       32,5892
                                                    700358656
                                                                 700360959
            26
26
                     26
                              27 ...
                                       32.5892
                                                    702292992
                                                                 702294015
```

#### **Practical 5**

#### **Assessing Data**

#### 5A1. Drop the Columns Where All Elements Are Missing Values

#### Code:

```
# -*- coding: utf-8 -*-
import sys
import os
import pandas as pd
Base='C:/VKHCG'
print('##############")
print('Working Base :',Base, 'using ', sys.platform)
print('##############")
sInputFileName='Good-or-Bad.csv'
sOutputFileName='Good-or-Bad-01.csv'
Company='01-Vermeulen'
Base='C:/VKHCG'
sFileDir=Base + '/' + Company + '/02-Assess/01-EDS/02-Python'
if not os.path.exists(sFileDir):
os.makedirs(sFileDir)
### Import Warehouse
sFileName=Base + '/' + Company + '/00-RawData/' + sInputFileName
print('Loading :',sFileName)
RawData=pd.read_csv(sFileName,header=0)
print('##############")
print('## Raw Data Values')
print('##############")
print(RawData)
print('##############")
print('## Data Profile')
print('#############")
print('Rows:',RawData.shape[0])
print('Columns:',RawData.shape[1])
print('##############")
```

```
sFileName=sFileDir + '/' + sInputFileName
RawData.to_csv(sFileName, index = False)
TestData=RawData.dropna(axis=1, how='all')
print('##############")
print('## Test Data Values')
print('#############")
print(TestData)
print('##############")
print('## Data Profile')
print('##############")
print('Rows:',TestData.shape[0])
print('Columns :',TestData.shape[1])
print('##############")
sFileName=sFileDir + '/' + sOutputFileName
TestData.to csv(sFileName, index = False)
print('##############")
print('### Done!! ############")
print('##############")
```

```
In [16]: runfile('C:/VKHCG/01-Vermeulen/02-Assess/Assess-Good-
Rad-01.py', wdir='C:/VKHCG/01-Vermeulen/02-Assess')
Working Base : C:/VKHCG using win32
*****************************
Loading : C:/VKHCG/01-Vermeulen/00-RawData/Good-or-Bad.csv
*****************************
## Raw Data Values
-------
                                             FieldF FieldG
     ID FieldA FieldB FieldC FieldD FieldE
         Good Better Best 1024.0
                                       NaN
                                            10241.0
    1.0
1
    2.0
         Good
                  NaN
                       Best
                             512.0
                                       NaN
                                             5121.0
                             256.0
         Good
               Better
                        NaN
                                       NaN
                                              256.0
2
    3.0
                                                         3
               Better
3
    4.0
         Good
                       Best
                               NaN
                                       NaN
                                              211.0
4
    5.0
          Good
               Better
                        NaN
                               64.0
                                       NaN
                                             6411.0
    6.0
         Good
                 NaN
                       Best
                              32.0
                                       NaN
                                               32.0
          NaN Better
                                       NaN
                                             1611.0
6
    7.0
                       Best
                              16.0
7
    8.0
          NaN
                  NaN
                       Best
                               8.0
                                       NaN
                                             8111.0
                                                         8
8
    9.0
          NaN
                  NaN
                        NaN
                               4.0
                                       NaN
                                               41.0
                                                         9
                               2.0
9
   10.0
                   В
                         C
                                       NaN
                                            21111.0
                                                        10
          NaN
                  NaN
                        NaN
                                       NaN
10
   NaN
                               NaN
                                                NaN
                                                        11
                                           102411.0
11
   10.0
         Good
               Better
                       Best 1024.0
                                       NaN
                                                        12
12
   10.0
         Good
                  NaN
                       Best
                             512.0
                                       NaN
                                              512.0
                                                        13
13
   10.0
         Good
               Better
                        NaN
                             256.0
                                       NaN
                                             1256.0
                                                        14
14
   10.0
               Better
                       Best
                               NaN
                                       NaN
                                                NaN
         Good
                                                        15
                                              164.0
15
   10.0
         Good Better
                        NaN
                               64.0
                                       NaN
                                                        16
16
   10.0
         Good
                  NaN
                       Best
                              32.0
                                       NaN
                                              322.0
                                                        17
17
          NaN Better
                                              163.0
                                                        18
   10.0
                       Best
                               16.0
                                       NaN
                                              844.0
18
  10.0
          NaN
                  NaN
                       Best
                               8.0
                                       NaN
                                                        19
          NaN
                                             4555.0
19 10.0
                  NaN
                        NaN
                               4.0
                                       NaN
                                                        20
20 10.0
           A
                  В
                         C
                               2.0
                                       NaN
                                              111.0
                                                        21
************
## Data Profile
******************************
Rows : 21
Columns: 8
***********
## Test Data Values
*************************
     ID FieldA FieldB FieldC FieldD
                                        FieldF FieldG
0
          Good
                Better
                        Best 1024.0
                                       10241.0
    1.0
          Good
                  NaN
                               512.0
                                        5121.0
1
    2.0
                         Best
                Better
                         NaN
                               256.0
                                         256.0
2
    3.0
          Good
                                                     3
    4.0
          Good
                Better
                                 NaN
                         Best
                                         211.0
    5.0
          Good
                Better
                          NaN
                                 64.0
                                        6411.0
    6.0
          Good
                   NaN
                         Best
                                 32.0
           NaN
                Better
                                        1611.0
                         Best
                                 16.0
    8.0
           NaN
                   NaN
                         Best
                                 8.0
                                        8111.0
                                                     8
8
    9.0
           NaN
                   NaN
                         NaN
                                 4.0
                                          41.0
                                                     9
9
   10.0
                   В
                           C
                                 2.0
                                       21111.0
                                                    10
10
   NaN
           NaN
                   NaN
                         NaN
                                 NaN
                                          NaN
                                                    11
                                      102411.0
11
   10.0
          Good
                Better
                         Best
                              1024.0
                                                    12
12
   10.0
          Good
                   NaN
                         Best
                               512.0
                                         512.0
                                                    13
                Better
13
   10.0
          Good
                         NaN
                               256.0
                                        1256.0
                                                    14
14
   10.0
          Good
                Better
                         Best
                                 NaN
                                          NaN
                                                    15
                                         164.0
15
                         NaN
                                 64.0
   10.0
          Good
                Better
                                                    16
                                 32.0
                                         322.0
                                                    17
16
   10.0
          Good
                   NaN
                         Best
17
   10.0
           NaN
                Better
                         Best
                                 16.0
                                         163.0
                                                    18
                                         844.0
18
   10.0
           NaN
                   NaN
                         Best
                                 8.0
                                                    19
19
   10.0
           NaN
                   NaN
                          NaN
                                 4.0
                                        4555.0
                                                    20
   10.0
            Α
                   В
                                 2.0
                                         111.0
                                                    21
************
## Data Profile
******************************
Rows: 21
Columns : 7
*******************************
Done!!
```

#### 5A2. Drop the Columns Where Any of the Elements Is Missing Values

#### Code:

```
# -*- coding: utf-8 -*-
import sys
import os
import pandas as pd
Base='C:/VKHCG'
sInputFileName='Good-or-Bad.csv'
sOutputFileName='Good-or-Bad-02.csv'
Company='01-Vermeulen'
Base='C:/VKHCG'
print('##############")
print('Working Base :',Base, 'using ', sys.platform)
print('##############")
sFileDir=Base + '/' + Company + '/02-Assess/01-EDS/02-Python'
if not os.path.exists(sFileDir):
os.makedirs(sFileDir)
### Import Warehouse
sFileName=Base + '/' + Company + '/00-RawData/' + sInputFileName
print('Loading :',sFileName)
RawData=pd.read_csv(sFileName,header=0)
print('#############")
print('## Raw Data Values')
print('##############")
print(RawData)
print('#############")
print('## Data Profile')
print('##############")
print('Rows:',RawData.shape[0])
print('Columns :',RawData.shape[1])
print('#############")
sFileName=sFileDir + '/' + sInputFileName
RawData.to csv(sFileName, index = False)
TestData=RawData.dropna(axis=1, how='any')
```

```
print('##############")
print('## Test Data Values')
print('##############")
print(TestData)
print('##############")
print('## Data Profile')
print('##############")
print('Rows:',TestData.shape[0])
print('Columns :',TestData.shape[1])
print('##############")
sFileName=sFileDir + '/' + sOutputFileName
TestData.to csv(sFileName, index = False)
print('##############")
print('### Done!! #############")
print('##############")
```

#### **Output:**

<pre>In [22]: runfile('C:/VKHCG/01-Vermeulen/02-Assess/Assess-Good- Bad-02.py', wdir='C:/VKHCG/01-Vermeulen/02-Assess')</pre>									## Data Profile ####################################		
											WARRAND WARRAND WARRAND WARRAND WING WING WARRAND WING WING WING WING WING WING WING WING
######################################											
***************************************									FieldG		
Loading : C:/VKHCG/01-Vermeulen/00-RawData/Good-or-Bad.csv									0	1	
***************************************									1	2	
## Raw Data Values									2	3	
***************************************									3	4	
		FieldA			FieldD	FieldE	FieldF	FieldG	4	5	
0	1.0	Good	Better	Best	1024.0	NaN	10241.0	1	5	6	
1	2.0	Good	NaN	Best	512.0	NaN	5121.0	2	6	7	
2	3.0	Good	Better	NaN	256.0	NaN	256.0	3	7	8	
3	4.0	Good	Better	Best	NaN	NaN	211.0	4	8 9	9 10	
4	5.0	Good	Better	NaN	64.0	NaN	6411.0	5	10	10	
5	6.0	Good	NaN	Best	32.0	NaN	32.0	6	11	12	
6	7.0	NaN	Better	Best	16.0	NaN	1611.0	7	12	13	
7	8.0	NaN	NaN	Best	8.0	NaN	8111.0	8	13	14	
8	9.0	NaN	NaN	NaN	4.0	NaN	41.0	9	14	15	
9	10.0	A	В	C	2.0	NaN	21111.0	10	15	16	
10	NaN	NaN	NaN	NaN	NaN	NaN	NaN	11	16	17	
11	10.0	Good	Better	Best	1024.0	NaN	102411.0	12	17	18	
12	10.0	Good	NaN	Best	512.0	NaN	512.0	13	18	19	
13	10.0	Good	Better	NaN	256.0	NaN	1256.0	14	19	20	
14	10.0	Good	Better	Best	NaN	NaN	NaN	15	20	21	
15	10.0	Good	Better	NaN	64.0	NaN	164.0	16		######################################	
16	10.0	Good	NaN	Best	32.0	NaN	322.0	17	## Data Profile		
17	10.0	NaN	Better	Best	16.0	NaN	163.0	18	Rows : 21		
18	10.0	NaN	NaN	Best	8.0	NaN	844.0	19	Columns : 1		
19	10.0	NaN	NaN	NaN	4.0	NaN	4555.0	20	***************************************		
20	10.0	A	В	C	2.0	NaN	111.0	21	Done!	!	

### **5A3.** Keep Only the Rows That Contain a Maximum of Two Missing Values

#### Code:

```
# -*- coding: utf-8 -*-
import sys
import os
import pandas as pd
sInputFileName='Good-or-Bad.csv'
sOutputFileName='Good-or-Bad-03.csv'
Company='01-Vermeulen'
Base='C:/VKHCG'
print('##############")
print('Working Base:',Base, 'using Windows ~~~~')
print('##############")
sFileDir=Base + '/' + Company + '/02-Assess/01-EDS/02-Python'
if not os.path.exists(sFileDir):
os.makedirs(sFileDir)
### Import Warehouse
sFileName=Base + '/' + Company + '/00-RawData/' + sInputFileName
print('Loading :',sFileName)
RawData=pd.read_csv(sFileName,header=0)
print('#############")
print('## Raw Data Values')
print('##############")
print(RawData)
print('##############")
print('## Data Profile')
print('#############")
print('Rows:',RawData.shape[0])
print('Columns:',RawData.shape[1])
print('##############")
sFileName=sFileDir + '/' + sInputFileName
RawData.to_csv(sFileName, index = False)
TestData=RawData.dropna(thresh=2)
print('#############")
```

```
print('## Test Data Values')
print('##############")
print(TestData)
print('##############")
print('## Data Profile')
print('#############")
print('Rows:',TestData.shape[0])
print('Columns :',TestData.shape[1])
print('#############")
sFileName=sFileDir + '/' + sOutputFileName
TestData.to csv(sFileName, index = False)
print('##############")
print('### Done!! #############")
print('##############")
```

#### **Output:**

```
## Test Data Values
In [35]: runfile('C:/VKHCG/01-Vermeulen/02-Assess/Assess-Good-
                                                           Bad-03.py', wdir='C:/VKHCG/01-Vermeulen/02-Assess')
                                                                                              FieldF FieldG
                                                               ID FieldA FieldB FieldC FieldD FieldE
                                                              1.0
                                                                   Good Better Best 1024.0
                                                                                         NaN
                                                                                             10241.0
                                                                                                       1
1 2.0
                                                                   Good
                                                                         NaN
                                                                             Best 512.0
                                                                                         NaN
                                                                                              5121.0
Working Base : C:/VKHCG using win32
                                                           2 3.0
                                                                   Good Better
                                                                              NaN
                                                                                  256.0
                                                                                         NaN
                                                                                               256.0
Loading : C:/VKHCG/01-Vermeulen/00-RawData/Good-or-Bad.csv
                                                           3
                                                              4.0
                                                                   Good Better
                                                                             Best
                                                                                   NaN
                                                                                         NaN
                                                                                              211.0
------
                                                           4
                                                              5.0
                                                                   Good Better
                                                                                   64.0
                                                                                         NaN
                                                                                              6411.0
                                                                              NaN
## Raw Data Values
                                                                             Best
                                                           5 6.0
                                                                   Good
                                                                         NaN
                                                                                   32.0
                                                                                         NaN
                                                                                               32.0
                                                                                                       6
ID FieldA FieldB FieldC FieldD FieldE
                                       FieldF FieldG
                                                           6 7.0
                                                                   NaN Better
                                                                                   16.0
                                                                                              1611.0
                                                                                                       7
   1.0 Good Better Best 1024.0
                                 NaN 10241.0
                                                  1
                                                           7 8.0
                                                                                              8111.0
                                                                   NaN
                                                                         NaN
                                                                             Best
                                                                                   8.0
                                                                                         NaN
                                                                                                       8
        Good
                                  NaN
                                       5121.0
   2.0
               NaN Best 512.0
                                                           8 9.0
                                                                   NaN
                                                                         NaN
                                                                              NaN
                                                                                   4.0
                                                                                         NaN
                                                                                               41.0
                                                                                                       9
   3.0
        Good Better
                    NaN 256.0
                                NaN
                                        256.0
    4.8
        Good Better
                    Best
                           NaN
                                  NaN
                                        211.0
                                                           9 10.0
                                                                    Α
                                                                          В
                                                                               C
                                                                                   2.0
                                                                                         NaN
                                                                                             21111.0
                                                                                                      10
4
    5.0
        Good Better
                     NaN
                           64.0
                                  NaN
                                        6411.0
                                                  5
                                                           11 10.0
                                                                   Good Better Best 1024.0
                                                                                         NaN 102411.0
                                                                                                      12
                          32.0
                                  NaN
                                        32.0
        Good
               NaN
                    Best
    6.0
                                                           12 10.0
                                                                   Good
                                                                         NaN
                                                                             Best
                                                                                  512.0
                                                                                         NaN
                                                                                               512.0
                                                                                                      13
   7.0
         NaN Better
                    Best
                          16.0
                                  NaN
                                        1611.0
                                                 7
                                                                                              1256.0
                                                           13 10.0
                                                                   Good Better
                                                                              NaN
                                                                                  256.0
                                                                                         NaN
                                                                                                      14
                           8.0
                                  NaN
                                        8111.0
    8.8
         NaN
               NaN
                    Best
                                                  8
8
    9.0
         NaN
               NaN
                     NaN
                           4.0
                                  NaN
                                         41.0
                                                  9
                                                           14 10.0
                                                                   Good Better
                                                                             Best
                                                                                   NaN
                                                                                         NaN
                                                                                                      15
                                  NaN
                                       21111.0
                                               10
   10.0
         A
               B
                     C
                           2.0
                                                           15 10.0
                                                                   Good Better
                                                                             NaN
                                                                                   64.0
                                                                                         NaN
                                                                                               164.0
                                                                                                      16
10 NaN
         NaN
               NaN
                    NaN
                            NaN
                                  NaN
                                          NaN
                                               11
                                                           16 10.0
                                                                   Good
                                                                             Best
                                                                                   32.0
                                                                                         NaN
                                                                                               322.0
                                                                                                      17
                    Best 1024.0
                                  NaN 102411.0
11 10.0
        Good Better
                                                 12
                                                                                              163.0
                                                           17 10.0
                                                                   NaN Better
                                                                             Best
                                                                                   16.0
                                                                                         NaN
                                                                                                      18
12 10.0
        Good
               NaN
                    Best
                          512.0
                                  NaN
                                        512.0
                                                 13
                                  NaN
                                        1256.0
13 10.0
        Good Better
                     NaN
                          256.0
                                                14
                                                           18 10.0
                                                                   NaN
                                                                         NaN
                                                                             Best
                                                                                   8.0
                                                                                         NaN
                                                                                               844.0
                                                                                                      19
14 10.0
        Good Better
                           NaN
                                  NaN
                                         NaN
                    Best
                                                           19 10.0
                                                                   NaN
                                                                         NaN
                                                                              NaN
                                                                                   4.0
                                                                                         NaN
                                                                                              4555.0
                                                                                                      20
15 10.0
                     NaN
                           64.0
                                  NaN
                                         164.0
        Good Better
                                                 16
                                                                                   2.0
                                                           20 10.0
                                                                    Α
                                                                          В
                                                                               C
                                                                                         NaN
                                                                                              111.0
                                                                                                      21
16
  10.0
        Good
               NaN
                    Best
                           32.0
                                  NaN
                                         322.0
                                                 17
                                                           17 10.0
                          16.0
                                  NaN
                                         163.0
         NaN Better
                    Best
                                                 18
18 10.0
         NaN
                          8.0
                                  NaN
                                         844.0
             NaN
                    Best
                                                           ## Data Profile
19 18.8
         NaN
               NaN
                     NaN
                           4.9
                                  NaN
                                        4555.0
                                                 28
                                                           В
                      C
                            2.0
                                  NaN
                                        111.0
                                                 21
20 10.0
          A
                                                           Rows : 20
## Data Profile
                                                           Columns: 8
Rows : 21
```

Done!!

3

4

5

Columns : 8

### 5A4. Fill All Missing Values with the Mean, Median, Mode, Minimum, and Maximum of the Particular Numeric Column

#### Code:

```
# -*- coding: utf-8 -
import os
import pandas as pd
Base='C:/VKHCG'
sInputFileName='Good-or-Bad.csv'
sOutputFileNameA='Good-or-Bad-04-A.csv'
sOutputFileNameB='Good-or-Bad-04-B.csv'
sOutputFileNameC='Good-or-Bad-04-C.csv'
sOutputFileNameD='Good-or-Bad-04-D.csv'
sOutputFileNameE='Good-or-Bad-04-E.csv'
Company='01-Vermeulen'
if sys.platform == 'linux':
Base=os.path.expanduser('~') + '/VKHCG'
else:
Base='C:/VKHCG'
print('###############")
print('Working Base:',Base,'using',sys.platform)
print('############")
sFileDir=Base + '/' + Company + '/02-Assess/01-EDS/02-Python'
if not os.path.exists(sFileDir):
    os.makedirs(sFileDir)
### Import Warehouse
sFileName=Base + '/' + Company + '/00-RawData/' + sInputFileName
print('Loading:',sFileName)
RawData=pd.read_csv(sFileName,header=0)
print('##############")
print('## Raw Data Values')
print('#############")
print(RawData) print('#######################")
print('## Data Profile')
print('#############")
print('Rows:',RawData.shape[0])
print('Columns:',RawData.shape[1])
print('##############")
sFileName=sFileDir + '/' + sInputFileName
```

```
RawData.to csv(sFileName,index=False)
TestData=RawData.fillna(RawData.mean())
print('#############")
print('## Test Data Values- Mean')
print('##############")
print(TestData)
print('#############")
print('## Data Profile')
print('#############")
print('Rows:',TestData.shape[0])
print('Columns:',TestData.shape[1])
print('##############")
sFileName=sFileDir + '/' + sOutputFileNameA
TestData.to csv(sFileName,index=False)
## This is the important action! The rest of this code snippet
## Only supports this action.
TestData=RawData.fillna(RawData.median())
print('##############")
print('## Test Data Values - Median')
print('##############")
print(TestData)
print('#############")
print('## Data Profile')
print('#############")
print('Rows:',TestData.shape[0])
print('Columns:',TestData.shape[1])
print('##############")
sFileName=sFileDir + '/' + sOutputFileNameB
TestData.to_csv(sFileName,index=False)
TestData=RawData.fillna(RawData.mode())
print('##############")
print('## Test Data Values - Mode')
print('#############")
print(TestData)
print('#############")
```

```
print('## Data Profile')
print('##############")
print('Rows:',TestData.shape[0])
print('Columns:',TestData.shape[1])
print('##############")
sFileName=sFileDir + '/' + sOutputFileNameC
TestData.to_csv(sFileName,index=False)
TestData=RawData.fillna(RawData.min())
print('#############")
print('## Test Data Values - Minumum')
print('##############")
print(TestData) print('#####################")
print('## Data Profile')
print('##############")
print('Rows:',TestData.shape[0])
print('Columns:',TestData.shape[1])
print('##############")
sFileName=sFileDir + '/' + sOutputFileNameD
TestData.to csv(sFileName,index=False)
TestData=RawData.fillna(RawData.max())
print('##############")
print('## Test Data Values - Maximum')
print('##############")
print(TestData)
print('#############")
print('##DataProfile')
print('##############")
print('Rows:',TestData,shape[0])
print('Columns:',TestData.shape[1])
print('##############")
sFileName=sFileDir + '/' + sOutputFileNameE
TestData.to csv(sFileName,index=False)
print('##############")
print('### Done!! ############")
```

```
In [7]: runfile('C:/VKHCG/01-Vermeulen/02-Assess/Assess-Good-Bad-05.py',
wdir='C:/VKHCG/01-Vermeulen/02-Assess')
*******************************
Working Base : C:/VKHCG using win32
************
Loading : C:/VKHCG/01-Vermeulen/00-RawData/Good-or-Bad.csv
*************************
## Raw Data Values
************************
     ID FieldA FieldB FieldC FieldD FieldE FieldF FieldG
     1.0 Good Better Best 1024.0 NaN 10241.0
0
  2.0 Good NaN Best 512.0 NaN 5121.0
1
  3.0 Good Better NaN 256.0 NaN 256.0
                                                                3
2
   4.0 Good Better Best NaN NaN 211.0
5.0 Good Better NaN 64.0 NaN 6411.0
6.0 Good NaN Best 32.0 NaN 32.0
7.0 NaN Better Best 16.0 NaN 1611.0
8.0 NaN NaN Best 8.0 NaN 8111.0
3
                                                               4
                                                                5
4
5
                                                                6
                                                                7
6
7
                                                               8
8 9.0 NaN
                  NaN NaN 4.0 NaN
                                                  41.0
                                                               9
9 10.0 A B C 2.0 NaN 21111.0
10 NaN NaN NaN NaN NaN NaN
                                                             10
                                                             11
11 10.0 Good Better Best 1024.0 NaN 102411.0
12 10.0 Good NaN Best 512.0 NaN 512.0
13 10.0 Good Better NaN 256.0 NaN 1256.0
14 10.0 Good Better Best NaN NaN NaN
                                                             12
                                                             13
                                                              14
                                                              15
15 10.0 Good Better NaN 64.0 NaN 164.0
                                                             16
16 10.0 Good NaN Best 32.0 NaN 322.0
                                                             17
17 10.0 NaN Better Best 16.0 NaN 163.0
                                                             18
18 10.0 NaN NaN Best 8.0 NaN 844.0
19 10.0 NaN NaN NaN 4.0 NaN 4555.0
20 10.0 A B C 2.0 NaN 111.0
                                                             19
                                                             20
                                                              21
## Data Profile
```

# Practical 6 Processing Data Hubs, Links, Satellite

```
# -*- coding: utf-8 -*-
import sys
import os
from datetime import datetime
from datetime import timedelta
from pytz import timezone, all_timezones
import pandas as pd
import sqlite3 as sq
from pandas.io import sql
import uuid
pd.options.mode.chained assignment = None
if sys.platform == 'linux':
 Base=os.path.expanduser('~') + '/VKHCG'
else:
 Base='C:/VKHCG'
print('##############")
print('Working Base:',Base, 'using', sys.platform)
print('##############")
Company='01-Vermeulen'
InputDir='00-RawData'
InputFileName='VehicleData.csv'
sDataBaseDir=Base + '/' + Company + '/03-Process/SQLite'
if not os.path.exists(sDataBaseDir):
 os.makedirs(sDataBaseDir)
sDatabaseName=sDataBaseDir + '/Hillman.db'
conn1 = sq.connect(sDatabaseName)
sDataVaultDir=Base + '/88-DV'
if not os.path.exists(sDataBaseDir):
 os.makedirs(sDataBaseDir)
sDatabaseName=sDataVaultDir + '/datavault.db'
conn2 = sq.connect(sDatabaseName)
```

```
base = datetime(2018,1,1,0,0,0)
numUnits=10*365*24
date list = [base - timedelta(hours=x) for x in range(0, numUnits)]
t=0
for i in date list:
 now utc=i.replace(tzinfo=timezone('UTC'))
 sDateTime=now_utc.strftime("%Y-%m-%d %H:%M:%S")
 print(sDateTime)
 sDateTimeKey=sDateTime.replace('','-').replace(':','-')
 t+=1
 IDNumber=str(uuid.uuid4())
 TimeLine=[('ZoneBaseKey', ['UTC']),
     ('IDNumber', [IDNumber]),
     ('nDateTimeValue', [now_utc]),
     ('DateTimeValue', [sDateTime]),
     ('DateTimeKey', [sDateTimeKey])]
   TimeFrame = pd.DataFrame.from items(TimeLine)
 else:
   TimeRow = pd.DataFrame.from items(TimeLine)
   TimeFrame = TimeFrame.append(TimeRow)
TimeHub=TimeFrame[['IDNumber', 'ZoneBaseKey', 'DateTimeKey', 'DateTimeValue']]
TimeHubIndex=TimeHub.set index(['IDNumber'].inplace=False)
TimeFrame.set_index(['IDNumber'],inplace=True)
sTable = 'Process-Time'
print('Storing :',sDatabaseName,' Table:',sTable)
TimeHubIndex.to sql(sTable, conn1, if exists="replace")
sTable = 'Hub-Time'
print('Storing :',sDatabaseName,' Table:',sTable)
TimeHubIndex.to sql(sTable, conn2, if exists="replace")
active_timezones=all_timezones
z=0
for zone in active timezones:
 t=0
 for j in range(TimeFrame.shape[0]):
   now date=TimeFrame['nDateTimeValue'][i]
   DateTimeKey=TimeFrame['DateTimeKey'][i]
   now utc=now date.replace(tzinfo=timezone('UTC'))
   sDateTime=now_utc.strftime("%Y-%m-%d %H:%M:%S")
   now zone = now utc.astimezone(timezone(zone))
```

```
sZoneDateTime=now zone.strftime("%Y-%m-%d %H:%M:%S")
   print(sZoneDateTime)
   t+=1
   z+=1
   IDZoneNumber=str(uuid.uuid4())
   TimeZoneLine=[('ZoneBaseKey', ['UTC']),
        ('IDZoneNumber', [IDZoneNumber]),
        ('DateTimeKey', [DateTimeKey]),
        ('UTCDateTimeValue', [sDateTime]),
        ('Zone', [zone]),
        ('DateTimeValue', [sZoneDateTime])]
   if t==1:
    TimeZoneFrame = pd.DataFrame.from items(TimeZoneLine)
   else:
    TimeZoneRow = pd.DataFrame.from_items(TimeZoneLine)
    TimeZoneFrame = TimeZoneFrame.append(TimeZoneRow)
 TimeZoneFrameIndex=TimeZoneFrame.set_index(['IDZoneNumber'],inplace=False)
 sZone=zone.replace('/','-').replace(' ',")
 sTable = 'Process-Time-'+sZone
 print('Storing :',sDatabaseName,' Table:',sTable)
 TimeZoneFrameIndex.to_sql(sTable, conn1, if_exists="replace")
sTable = 'Satellite-Time-'+sZone
 print('Storing :',sDatabaseName,' Table:',sTable)
 TimeZoneFrameIndex.to sql(sTable, conn2, if exists="replace")
print('##########")
print('Vacuum Databases')
sSQL="VACUUM;"
sql.execute(sSQL,conn1)
sql.execute(sSQL,conn2)
print('##########")
print('### Done!! ###############################)
```

```
In [33]: runfile('C:/Users/MSCIT/.spyder-py3/temp.py',
wdir='C:/Users/MSCIT/.spyder-py3')
************
Working Base : C:/VKHCG using win32
2018-01-01 00:00:00
2017-12-31 23:00:00
2017-12-31 22:00:00
2017-12-31 21:00:00
2017-12-31 20:00:00
2017-12-31 19:00:00
2017-12-31 18:00:00
2017-12-31 17:00:00
2017-12-31 16:00:00
2017-12-31 15:00:00
Storing: C:/VKHCG/88-DV/datavault.db Table: Process-Time
Storing: C:/VKHCG/88-DV/datavault.db Table: Hub-Time
2018-01-01 00:00:00
2017-12-31 23:00:00
2017-12-31 22:00:00
2017-12-31 21:00:00
2017-12-31 20:00:00
2017-12-31 19:00:00
2017-12-31 18:00:00
2017-12-31 17:00:00
2017-12-31 16:00:00
2017-12-31 15:00:00
Storing: C:/VKHCG/88-DV/datavault.db Table: Process-Time-
Africa-Abidjan
Storing : C:/VKHCG/88-DV/datavault.db Table: Satellite-
Time-Africa-Abidjan
2018-01-01 00:00:00
2017-12-31 23:00:00
2017-12-31 22:00:00
2017-12-31 21:00:00
2017-12-31 20:00:00
2017-12-31 19:00:00
2017-12-31 18:00:00
2017-12-31 17:00:00
2017-12-31 16:00:00
2017-12-31 15:00:00
```

# Practical 7 Transforming Data

Sun Model

```
# -*- coding: utf-8 -*-
import sys
import os
from datetime import datetime
from pytz import timezone
import pandas as pd
import sqlite3 as sq
import uuid
pd.options.mode.chained assignment = None
if sys.platform == 'linux':
 Base=os.path.expanduser('~') + '/VKHCG'
else:
 Base='C:/VKHCG'
print('##############")
print('Working Base :',Base, 'using ', sys.platform)
print('##############")
Company='01-Vermeulen'
sDataBaseDir=Base + '/' + Company + '/04-Transform/SQLite'
if not os.path.exists(sDataBaseDir):
 os.makedirs(sDataBaseDir)
sDatabaseName=sDataBaseDir + '/Vermeulen.db'
conn1 = sq.connect(sDatabaseName)
sDataWarehousetDir=Base + '/99-DW'
if not os.path.exists(sDataWarehousetDir):
 os.makedirs(sDataWarehousetDir)
sDatabaseName=sDataWarehousetDir + '/datawarehouse.db'
conn2 = sq.connect(sDatabaseName)
print('\n#############")
print('Time Dimension')
BirthZone = 'Atlantic/Reykjavik'
BirthDateUTC = datetime(1960,12,20,10,15,0)
BirthDateZoneUTC=BirthDateUTC.replace(tzinfo=timezone('UTC'))
BirthDateZoneStr=BirthDateZoneUTC.strftime("%Y-%m-%d %H:%M:%S")
BirthDateZoneUTCStr=BirthDateZoneUTC.strftime("%Y-%m-%d %H:%M:%S (%Z) (%z)")
```

```
BirthDate = BirthDateZoneUTC.astimezone(timezone(BirthZone))
BirthDateStr=BirthDate.strftime("%Y-%m-%d %H:%M:%S (%Z) (%z)")
BirthDateLocal=BirthDate.strftime("%Y-%m-%d %H:%M:%S")
IDTimeNumber=str(uuid.uuid4())
TimeLine=[('TimeID', [IDTimeNumber]),
   ('UTCDate', [BirthDateZoneStr]),
   ('LocalTime', [BirthDateLocal]),
   ('TimeZone', [BirthZone])]
TimeFrame = pd.DataFrame.from_items(TimeLine)
DimTime=TimeFrame
DimTimeIndex=DimTime.set index(['TimeID'],inplace=False)
sTable = 'Dim-Time'
print('\n##############")
print('Storing :',sDatabaseName,'\n Table:',sTable)
print('\n##############")
DimTimeIndex.to sql(sTable, conn1, if exists="replace")
DimTimeIndex.to_sql(sTable, conn2, if_exists="replace")
print('\n##############")
print('Dimension Person')
print('\n##############")
FirstName = 'Guðmundur'
LastName = 'Gunnarsson'
IDPersonNumber=str(uuid.uuid4())
PersonLine=[('PersonID', [IDPersonNumber]),
     ('FirstName', [FirstName]),
     ('LastName', [LastName]),
     ('Zone', ['UTC']),
     ('DateTimeValue', [BirthDateZoneStr])]
PersonFrame = pd.DataFrame.from items(PersonLine)
DimPerson=PersonFrame
DimPersonIndex=DimPerson.set_index(['PersonID'],inplace=False)
sTable = 'Dim-Person'
print('\n#############")
print('Storing :',sDatabaseName,'\n Table:',sTable)
print('\n#############")
DimPersonIndex.to sql(sTable, conn1, if exists="replace")
DimPersonIndex.to sql(sTable, conn2, if exists="replace")
print('\n##############")
```

```
print('Fact - Person - time')
print('\n##############")
IDFactNumber=str(uuid.uuid4())
PersonTimeLine=[('IDNumber', [IDFactNumber]),
      ('IDPersonNumber', [IDPersonNumber]),
      ('IDTimeNumber', [IDTimeNumber])]
PersonTimeFrame = pd.DataFrame.from_items(PersonTimeLine)
FctPersonTime=PersonTimeFrame
FctPersonTimeIndex=FctPersonTime.set_index(['IDNumber'],inplace=False)
sTable = 'Fact-Person-Time'
print('\n##############")
print('Storing :',sDatabaseName,'\n Table:',sTable)
print('\n##############")
FctPersonTimeIndex.to sql(sTable, conn1, if exists="replace")
FctPersonTimeIndex.to_sql(sTable, conn2, if_exists="replace")
```

```
In [24]: runfile('C:/VKHCG/01-Vermeulen/04-Transform/Transform-
Gunnarsson-Sun-Model.py', wdir='C:/VKHCG/01-Vermeulen/04-
Transform')
***********
Working Base : C:/VKHCG using win32
*******************************
******************************
Time Dimension
****************************
Storing: C:/VKHCG/99-DW/datawarehouse.db
Table: Dim-Time
*******************************
C:/VKHCG/01-Vermeulen/04-Transform/Transform-Gunnarsson-Sun-
Model.py:55: FutureWarning: from_items is deprecated. Please use
DataFrame.from_dict(dict(items), ...) instead.
DataFrame.from_dict(OrderedDict(items)) may be used to preserve
the key order.
 DimTime=TimeFrame
*******************************
Dimension Person
_____
Storing: C:/VKHCG/99-DW/datawarehouse.db
Table: Dim-Person
******************************
C:/VKHCG/01-Vermeulen/04-Transform/Transform-Gunnarsson-Sun-
Model.py:79: FutureWarning: from_items is deprecated. Please use
DataFrame.from_dict(dict(items), ...) instead.
DataFrame.from_dict(OrderedDict(items)) may be used to preserve
the key order.
  DimPerson=PersonFrame
*****************************
Fact - Person - time
******************************
************************
Storing : C:/VKHCG/99-DW/datawarehouse.db
 Table: Fact-Person-Time
******************************
C:/VKHCG/01-Vermeulen/04-Transform/Transform-Gunnarsson-Sun-
Model.py:98: FutureWarning: from items is deprecated. Please use
DataFrame.from dict(dict(items), ...) instead.
DataFrame.from dict(OrderedDict(items)) may be used to preserve
the key order.
  FctPersonTime=PersonTimeFrame
```

## Practical 8A Organizing Data

#### A. Horizontal Style

```
# -*- coding: utf-8 -*-
import sys
import os
import pandas as pd
import sqlite3 as sq
if sys.platform == 'linux':
 Base=os.path.expanduser('~') + '/VKHCG'
else:
 Base='C:/VKHCG'
print('#############")
print('Working Base :',Base, ' using ', sys.platform)
print('##############")
Company='01-Vermeulen'
sDataWarehouseDir=Base + '/99-DW'
if not os.path.exists(sDataWarehouseDir):
 os.makedirs(sDataWarehouseDir)
sDatabaseName=sDataWarehouseDir + '/datawarehouse.db'
conn1 = sq.connect(sDatabaseName)
sDatabaseName=sDataWarehouseDir + '/datamart.db'
conn2 = sq.connect(sDatabaseName)
print('##########")
sTable = 'Dim-BMI'
print('Loading :',sDatabaseName,' Table:',sTable)
sSOL="SELECT * FROM [Dim-BMI];"
PersonFrame0=pd.read sql query(sSQL, conn1)
print('#############")
sTable = 'Dim-BMI'
print('Loading :',sDatabaseName,' Table:',sTable)
print('##############")
sSQL="SELECT PersonID,\
  Height,\
  Weight,\
```

```
bmi.\
  Indicator\
FROM [Dim-BMI]\
WHERE \
Height > 1.5 \setminus
and Indicator = 1
ORDER BY \
  Height,\
  Weight;"
PersonFrame1=pd.read_sql_query(sSQL, conn1)
DimPerson=PersonFrame1
DimPersonIndex=DimPerson.set index(['PersonID'],inplace=False)
sTable = 'Dim-BMI-Horizontal'
print('\n##############")
print('Storing :',sDatabaseName,'\n Table:',sTable)
print('\n###############")
DimPersonIndex.to sql(sTable, conn2, if exists="replace")
print('##############")
sTable = 'Dim-BMI-Horizontal'
print('Loading :',sDatabaseName,' Table:',sTable)
print('#############")
sSQL="SELECT * FROM [Dim-BMI];"
PersonFrame2=pd.read sql query(sSQL, conn2)
print('##############")
print('Full Data Set (Rows):', PersonFrame0.shape[0])
print('Full Data Set (Columns):', PersonFrame0.shape[1])
print('##############")
print('Horizontal Data Set (Rows):', PersonFrame2.shape[0])
print('Horizontal Data Set (Columns):', PersonFrame2.shape[1])
print('##############")
```

#### **Output:**

\*

```
In [20]: runfile('C:/VKHCG/01-Vermeulen/05-Organise/Organize-Horizontal.py', wdir='C:/VKHCG/
01-Vermeulen/05-Organise')
****************************
Working Base : C:/VKHCG using win32
**************
Loading : C:/VKHCG/99-DW/datamart.db Table: Dim-BMI
******************************
Loading : C:/VKHCG/99-DW/datamart.db Table: Dim-BMI
*****************************
*******************************
Storing : C:/VKHCG/99-DW/datamart.db
Table: Dim-BMI-Horizontal
*****************************
******************************
Loading : C:/VKHCG/99-DW/datamart.db Table: Dim-BMI-Horizontal
Full Data Set (Rows): 1080
Full Data Set (Columns): 5
*******************************
Horizontal Data Set (Rows): 194
Horizontal Data Set (Columns): 5
```

#### Practical 8B Vertical style

```
# -*- coding: utf-8 -*-
import sys
import os
import pandas as pd
import sqlite3 as sq
if sys.platform == 'linux':
 Base=os.path.expanduser('~') + '/VKHCG'
else:
 Base='C:/VKHCG'
print('###############")
print('Working Base :',Base, 'using ', sys.platform)
print('##############")
Company='01-Vermeulen'
sDataWarehouseDir=Base + '/99-DW'
if not os.path.exists(sDataWarehouseDir):
 os.makedirs(sDataWarehouseDir)
sDatabaseName=sDataWarehouseDir + '/datawarehouse.db'
conn1 = sq.connect(sDatabaseName)
sDatabaseName=sDataWarehouseDir + '/datamart.db'
conn2 = sq.connect(sDatabaseName)
print('##############")
sTable = 'Dim-BMI'
print('Loading :',sDatabaseName,' Table:',sTable)
sSQL="SELECT * FROM [Dim-BMI];"
PersonFrame0=pd.read sql query(sSOL, conn1)
print('###############")
sTable = 'Dim-BMI'
print('Loading :',sDatabaseName,' Table:',sTable)
print('#############")
sSQL="SELECT \
  Height,\
  Weight,\
  Indicator\
FROM [Dim-BMI];"
PersonFrame1=pd.read_sql_query(sSQL, conn1)
DimPerson=PersonFrame1
```

```
DimPersonIndex=DimPerson.set index(['Indicator'],inplace=False)
sTable = 'Dim-BMI-Vertical'
print('\n##############")
print('Storing :',sDatabaseName,'\n Table:',sTable)
print('\n##############")
DimPersonIndex.to_sql(sTable, conn2, if_exists="replace")
print('##########")
sTable = 'Dim-BMI-Vertical'
print('Loading :',sDatabaseName,' Table:',sTable)
sSQL="SELECT * FROM [Dim-BMI-Vertical];"
PersonFrame2=pd.read_sql_query(sSQL, conn2)
print('###############")
print('Full Data Set (Rows):', PersonFrame0.shape[0])
print('Full Data Set (Columns):', PersonFrame0.shape[1])
print('##############")
print('Horizontal Data Set (Rows):', PersonFrame2.shape[0])
print('Horizontal Data Set (Columns):', PersonFrame2.shape[1])
print('###############")
```

```
In [20]: runfile('C:/VKHCG/01-Vermeulen/05-Organise/Organize-Horizontal.py', wdir='C:/VKHCG/
01-Vermeulen/05-Organise')
Working Base : C:/VKHCG using win32
*************
Loading : C:/VKHCG/99-DW/datamart.db Table: Dim-BMI
Loading : C:/VKHCG/99-DW/datamart.db Table: Dim-BMI
------
Storing: C:/VKHCG/99-DW/datamart.db
Table: Dim-BMI-Horizontal
*******************************
Loading : C:/VKHCG/99-DW/datamart.db Table: Dim-BMI-Horizontal
*******************************
Full Data Set (Rows): 1080
Full Data Set (Columns): 5
******************************
Horizontal Data Set (Rows): 194
Horizontal Data Set (Columns): 5
```

#### Practical 8C Island Style

```
# -*- coding: utf-8 -*-
import sys
import os
import pandas as pd
import sqlite3 as sq
if sys.platform == 'linux':
 Base=os.path.expanduser('~') + '/VKHCG'
else:
 Base='C:/VKHCG'
print('##############")
print('Working Base :',Base, 'using ', sys.platform)
print('###############")
Company='01-Vermeulen'
sDataWarehouseDir=Base + '/99-DW'
if not os.path.exists(sDataWarehouseDir):
 os.makedirs(sDataWarehouseDir)
sDatabaseName=sDataWarehouseDir + '/datawarehouse.db'
conn1 = sq.connect(sDatabaseName)
sDatabaseName=sDataWarehouseDir + '/datamart.db'
conn2 = sq.connect(sDatabaseName)
print('##########")
sTable = 'Dim-BMI'
print('Loading :',sDatabaseName,' Table:',sTable)
sSOL="SELECT * FROM [Dim-BMI];"
PersonFrame0=pd.read_sql_query(sSQL, conn1)
print('##########")
sTable = 'Dim-BMI'
print('Loading :',sDatabaseName,' Table:',sTable)
sSOL="SELECT \
  Height,\
  Weight,\
```

```
Indicator\
FROM [Dim-BMI]\
WHERE Indicator > 2\
ORDER BY \
  Height,\
  Weight;"
PersonFrame1=pd.read sql query(sSQL, conn1)
DimPerson=PersonFrame1
DimPersonIndex=DimPerson.set_index(['Indicator'],inplace=False)
sTable = 'Dim-BMI-Vertical'
print('\n#############")
print('Storing :',sDatabaseName,'\n Table:',sTable)
print('\n##############")
DimPersonIndex.to_sql(sTable, conn2, if_exists="replace")
print('###############")
sTable = 'Dim-BMI-Vertical'
print('Loading :',sDatabaseName,' Table:',sTable)
print('##############")
sSQL="SELECT * FROM [Dim-BMI-Vertical];"
PersonFrame2=pd.read_sql_query(sSQL, conn2)
print('##############")
print('Full Data Set (Rows):', PersonFrame0.shape[0])
print('Full Data Set (Columns):', PersonFrame0.shape[1])
print('##############")
print('Horizontal Data Set (Rows):', PersonFrame2.shape[0])
print('Horizontal Data Set (Columns):', PersonFrame2.shape[1])
print('##############")
```

```
In [38]: runfile('C:/VKHCG/01-Vermeulen/05-Organise/Organize-
Island.py', wdir='C:/VKHCG/01-Vermeulen/05-Organise')
******************************
Working Base : C:/VKHCG using win32
********************************
**************
Loading : C:/VKHCG/99-DW/datamart.db Table: Dim-BMI
******
Loading : C:/VKHCG/99-DW/datamart.db Table: Dim-BMI
*****************************
Storing: C:/VKHCG/99-DW/datamart.db
Table: Dim-BMI-Vertical
***********
Loading: C:/VKHCG/99-DW/datamart.db Table: Dim-BMI-Vertical
********************************
***********
Full Data Set (Rows): 1080
Full Data Set (Columns): 5
Horizontal Data Set (Rows): 771
Horizontal Data Set (Columns): 3
```

#### Practical 8D Secure Vault Style

```
# -*- coding: utf-8 -*-
import sys
import os
import pandas as pd
import sqlite3 as sq
if sys.platform == 'linux':
 Base=os.path.expanduser('~') + '/VKHCG'
else:
 Base='C:/VKHCG'
print('##############")
print('Working Base :',Base, 'using ', sys.platform)
print('##############")
Company='01-Vermeulen'
sDataWarehouseDir=Base + '/99-DW'
if not os.path.exists(sDataWarehouseDir):
 os.makedirs(sDataWarehouseDir)
sDatabaseName=sDataWarehouseDir + '/datawarehouse.db'
conn1 = sq.connect(sDatabaseName)
sDatabaseName=sDataWarehouseDir + '/datamart.db'
conn2 = sq.connect(sDatabaseName)
print('###########")
sTable = 'Dim-BMI'
print('Loading :',sDatabaseName,' Table:',sTable)
sSQL="SELECT * FROM [Dim-BMI];"
PersonFrame0=pd.read sql query(sSQL, conn1)
print('##########")
sTable = 'Dim-BMI'
```

```
print('Loading :',sDatabaseName,' Table:',sTable)
sSQL="SELECT \
  Height,\
  Weight,\
  Indicator,\
  CASE Indicator\
  WHEN 1 THEN 'Pip'\
  WHEN 2 THEN 'Norman'\
  WHEN 3 THEN 'Grant'\
  ELSE 'Sam'\
  END AS Name
FROM [Dim-BMI]\
WHERE Indicator > 2\
ORDER BY \
  Height,\
  Weight:"
PersonFrame1=pd.read_sql_query(sSQL, conn1)
DimPerson=PersonFrame1
DimPersonIndex=DimPerson.set index(['Indicator'],inplace=False)
sTable = 'Dim-BMI-Secure'
print('\n##############")
print('Storing :',sDatabaseName,'\n Table:',sTable)
print('\n##############")
DimPersonIndex.to sql(sTable, conn2, if exists="replace")
print('#############")
sTable = 'Dim-BMI-Secure'
print('Loading :',sDatabaseName,' Table:',sTable)
print('##############")
sSQL="SELECT * FROM [Dim-BMI-Secure] WHERE Name = 'Sam';"
PersonFrame2=pd.read sql query(sSQL, conn2)
print('##############")
print('Full Data Set (Rows):', PersonFrame0.shape[0])
print('Full Data Set (Columns):', PersonFrame0.shape[1])
print('##############")
print('Horizontal Data Set (Rows):', PersonFrame2.shape[0])
print('Horizontal Data Set (Columns):', PersonFrame2.shape[1])
```

```
In [47]: runfile('C:/VKHCG/01-Vermeulen/05-Organise/Organize-Secure-
Vault.py', wdir='C:/VKHCG/01-Vermeulen/05-Organise')
*************************
Working Base : C:/VKHCG using win32
Loading: C:/VKHCG/99-DW/datamart.db Table: Dim-BMI
##################
Loading : C:/VKHCG/99-DW/datamart.db Table: Dim-BMI
*************************
Storing : C:/VKHCG/99-DW/datamart.db
Table: Dim-BMI-Secure
*******************************
*****************************
Loading: C:/VKHCG/99-DW/datamart.db Table: Dim-BMI-Secure
**************************
**************************
Full Data Set (Rows): 1080
Full Data Set (Columns): 5
************
Horizontal Data Set (Rows): 692
Horizontal Data Set (Columns): 4
Only Sam Data
  Indicator Height Weight Name
            1.0 35 Sam
       4
        4
             1.0
                    40 Sam
             1.0
        4
                    45 Sam
2
             1.0
                     50 Sam
3
        4
        4
             1.0
                     55 Sam
*************
```

# Practical 9 Reporting Data

#### A. Create a Network Routing Diagram

```
import sys
import os
import pandas as pd
import networkx as nx
import matplotlib.pyplot as plt
pd.options.mode.chained assignment = None
if sys.platform == 'linux':
 Base=os.path.expanduser('~') + 'VKHCG'
else:
 Base='C:/VKHCG'
print('##############")
print('Working Base:',Base, 'using', sys.platform)
print('#############")
sInputFileName='02-Assess/01-EDS/02-Python/Assess-Network-Routing-Company.csv'
sOutputFileName1='05-Organise/01-EDS/02-Python/Organise-Network-Routing-Company.gml'
sOutputFileName2='05-Organise/01-EDS/02-Python/Organise-Network-Routing-Company.png'
Company='01-Vermeulen'
### Import Country Data
sFileName=Base + '/' + Company + '/' + sInputFileName
print('#############")
print('Loading :',sFileName)
print('##############")
CompanyData=pd.read_csv(sFileName,header=0,low_memory=False, encoding="latin-1")
print('#############")
print(CompanyData.head())
print(CompanyData.shape)
```

```
G=nx.Graph()
for i in range(CompanyData.shape[0]):
 for i in range(CompanyData.shape[0]):
   Node0=CompanyData['Company_Country_Name'][i]
   Node1=CompanyData['Company_Country_Name'][j]
   if Node0 != Node1:
     G.add edge(Node0,Node1)
for i in range(CompanyData.shape[0]):
 Node0=CompanyData['Company Country Name'][i]
 Node1=CompanyData['Company_Place_Name'][i] + '('+
CompanyData['Company Country Name'][i] + ')'
 if Node0 != Node1:
   G.add_edge(Node0,Node1)
print('Nodes:', G.number_of_nodes())
print('Edges:', G.number_of_edges())
sFileName=Base + '/' + Company + '/' + sOutputFileName1
print('#############")
print('Storing :',sFileName)
print('##############")
nx.write_gml(G, sFileName)
sFileName=Base + '/' + Company + '/' + sOutputFileName2
print('#############")
print('Storing Graph Image:',sFileName)
print('##############")
plt.figure(figsize=(15, 15))
pos=nx.spectral_layout(G,dim=2)
nx.draw_networkx_nodes(G,pos, node_color='k', node_size=10, alpha=0.8)
nx.draw_networkx_edges(G, pos,edge_color='r', arrows=False, style='dashed')
nx.draw_networkx_labels(G,pos,font_size=12,font_family='sans-serif',font_color='b')
plt.axis('off')
plt.savefig(sFileName,dpi=600)
plt.show()
print('#############")
print('### Done!! #############")
```

```
In [63]: runfile('C:/VKHCG/01-Vermeulen/05-Organise/Organise-Network-Routing-Company.py', wdir='C:/VKHCG/01-
Vermeulen/05-Organise')
*******************************
Working Base : C:/VKHCG using win32
************
Loading: C:/VKHCG/01-Vermeulen/02-Assess/01-EDS/02-Python/Assess-Network-Routing-Company.csv
*******
Company_Country_Code ... Company_Country_Name
US ... United States of America
                US ... United States of America
1
2
                US ... United States of America
                US ... United States of America
                US ... United States of America
[5 rows x 5 columns]
(150, 5)
Nodes: 6
Edges: 6
Storing : C:/VKHCG/01-Vermeulen/05-Organise/01-EDS/02-Python/Organise-Network-Routing-Company.gml
******************************
******************************
Storing Graph Image: C:/VKHCG/01-Vermeulen/05-Organise/01-EDS/02-Python/Organise-Network-Routing-Company.png
************
```

# Practical 9B Directed Acyclic Graph

```
import networkx as nx
import matplotlib.pyplot as plt
import sys
import os
import pandas as pd
if sys.platform == 'linux':
 Base=os.path.expanduser('~') + 'VKHCG'
else:
 Base='C:/VKHCG'
print('################")
print('Working Base :',Base, 'using ', sys.platform)
print('##############")
sInputFileName='01-Retrieve/01-EDS/02-Python/Retrieve Router Location.csv'
sOutputFileName1='Assess-DAG-Company-Country.png'
sOutputFileName2='Assess-DAG-Company-Country-Place.png'
Company='01-Vermeulen'
### Import Company Data
sFileName=Base + '/' + Company + '/' + sInputFileName
print('#############")
print('Loading :',sFileName)
print('##############")
CompanyData=pd.read_csv(sFileName,header=0,low_memory=False, encoding="latin-1")
print('Loaded Company :',CompanyData.columns.values)
print('#############")
print(CompanyData)
print('##############")
print('Rows: ',CompanyData.shape[0])
print('##############")
G1=nx.DiGraph()
G2=nx.DiGraph()
```

```
for i in range(CompanyData.shape[0]):
 G1.add node(CompanyData['Country'][i])
 sPlaceName= CompanyData['Place_Name'][i] + '-' + CompanyData['Country'][i]
 G2.add_node(sPlaceName)
print('##############")
for n1 in G1.nodes():
 for n2 in G1.nodes():
   if n1 != n2:
     print('Link:',n1,' to ', n2)
     G1.add_edge(n1,n2)
print('#############")
print("Nodes of graph: ")
print(G1.nodes())
print("Edges of graph: ")
print(G1.edges())
print('##############")
sFileDir=Base + '/' + Company + '/02-Assess/01-EDS/02-Python'
if not os.path.exists(sFileDir):
 os.makedirs(sFileDir)
sFileName=sFileDir + '/' + sOutputFileName1
print('#############")
print('Storing :', sFileName)
print('#############")
nx.draw(G1,pos=nx.spectral layout(G1),
   nodecolor='r',edge_color='g',
   with_labels=True,node_size=8000,
   font_size=12)
plt.savefig(sFileName) # save as png
plt.show() # display
print('##############")
for n1 in G2.nodes():
 for n2 in G2.nodes():
   if n1 != n2:
     print('Link:',n1,' to ', n2)
     G2.add_edge(n1,n2)
```

```
print('###############")
print("Nodes of graph: ")
print(G2.nodes())
print("Edges of graph: ")
print(G2.edges())
print('##############")
sFileDir=Base + '/' + Company + '/02-Assess/01-EDS/02-Python'
if not os.path.exists(sFileDir):
 os.makedirs(sFileDir)
sFileName=sFileDir + '/' + sOutputFileName2
print('##############")
print('Storing :', sFileName)
print('#############")
nx.draw(G2,pos=nx.spectral_layout(G2),
   nodecolor='r',edge_color='b',
   with_labels=True,node_size=8000,
   font_size=12)
plt.savefig(sFileName) # save as png
plt.show() # display
```

```
Working Base : C:/VKHCG using
******************************
*************************
Loading : C:/VKHCG/01-Vermeulen/01-Retrieve/01-EDS/02-Python/
Retrieve_Router_Location.csv
Loaded Company: ['Country' 'Place_Name' 'Latitude' 'Longitude']
                                 Longitude
    Country Place_Name
                      Latitude
                                 -73.9725
-74.0052
         US
             New York
                         40.7528
         US
              New York
                         40.7214
         US
                        40.7662
                                   -73.9862
                        40.7449
         US
             New York
                                  -73.9782
         US
              New York
                        40.7588
                                  -73.9680
6
7
8
         US
                         40.7637
             New York
                                   -73.9727
         US
                         40.7553
                                   -73.9924
         US
              New York
                        40.7308
                                  -73.9975
         US
              New York
                         40.7694
                                   -73.9609
        US
US
                        40.7330
40.7505
              New York
                                   -74.0078
                                   -73.9931
11
              New York
              New York
13
14
         US
              New York
                         40.7082
                                   -74.0132
         US
             New York
                         40.7267
                                   -73.9981
         US
                         40.7168
                                   -73.9861
16
        US
             New York
                         40.7317
                                   -73.9885
        US
             New York
                        40.7584
                                   -73.9794
        US
US
             New York
New York
                        40.7592
40.7055
                                   -73.9778
19
                                   -74.0050
        US
US
              New York
                        40.7089
                                  -74.0012
             New York
                        40.7391
                                   -73.9826
```

#### Practical 9 C Graphics

#### **CODE**

```
import sys
import os
import pandas as pd
import matplotlib as ml
from matplotlib import pyplot as plt
if sys.platform == 'linux':
 Base=os.path.expanduser('~') + '/VKHCG'
else:
 Base='C:/VKHCG'
print('##############")
print('Working Base :',Base, 'using ', sys.platform)
print('#############")
GBase = Base+'/01-Vermeulen/06-Report/01-EDS/02-Python/'
ml.style.use('ggplot')
data=[
['London',
             29.2, 17.4],
['Glasgow',
             18.8, 11.3],
['Cape Town', 15.3, 9.0],
             22.0, 7.8],
['Houston',
['Perth', 18.0, 23.7],
['San Francisco',
                   11.4, 33.3]
os_new=pd.DataFrame(data)
pd.Index(['Item', 'Value', 'Value Percent', 'Conversions', 'Conversion Percent',
   'URL', 'Stats URL'],
   dtype='object')
os_new.rename(columns = {0 : "Warehouse Location"}, inplace=True)
os_new.rename(columns = {1 : "Profit 2016"}, inplace=True)
os_new.rename(columns = {2 : "Profit 2017"}, inplace=True)
explode = (0, 0, 0, 0, 0, 0.1)
labels=os_new['Warehouse Location']
```