RELATIONAL DATABASE & SQL

謝昇峯 Sheng-Feng Hsieh, Ph.D.

sfhsieh@ntu.edu.tw

Assistant Professor of Accounting

College of Management, National Taiwan University

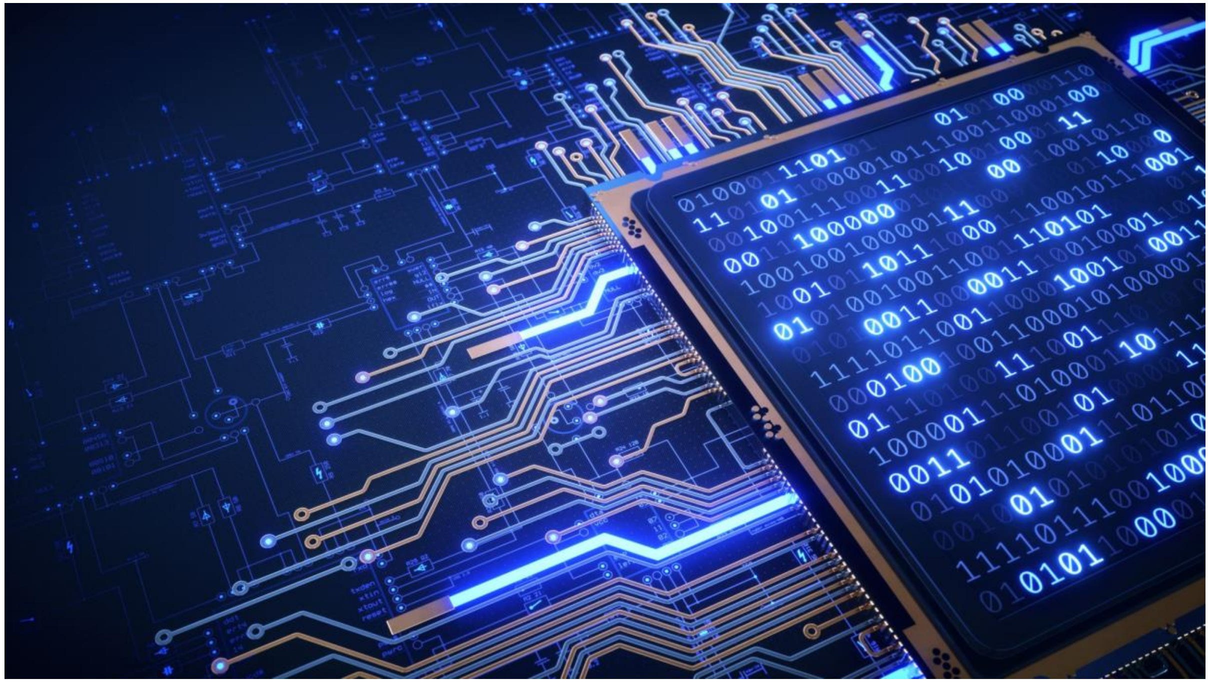
國立中山大學政治學研究所「資料分析方法入門」課程

December 6, 2023

Learning Objectives

- Describe what a **relational database** is, how it organizes data, and how to create a set of well-structured relational database tables.

- Query a relational database using **structured query language (SQL).**



Relational Database

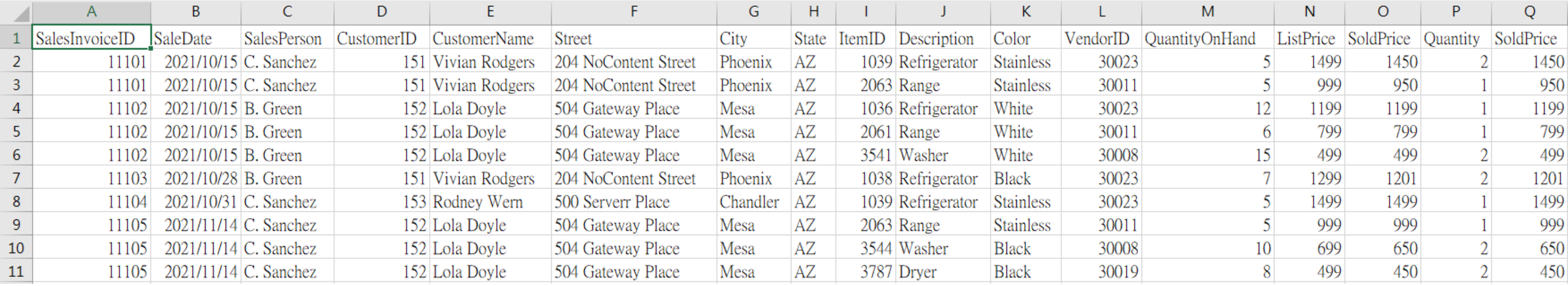
What Is a Database?

- A **database** is **organized collection of data** about a set of entities stored with as little data redundancy as possible.

- Stores all of the data needed to operate the business while **linking data** across various functions and **eliminating redundancy**.

- A **file** is a related group of records.
- A **record** is a related group of fields.
- A **field** is a specific attribute of interest for the entity (record).

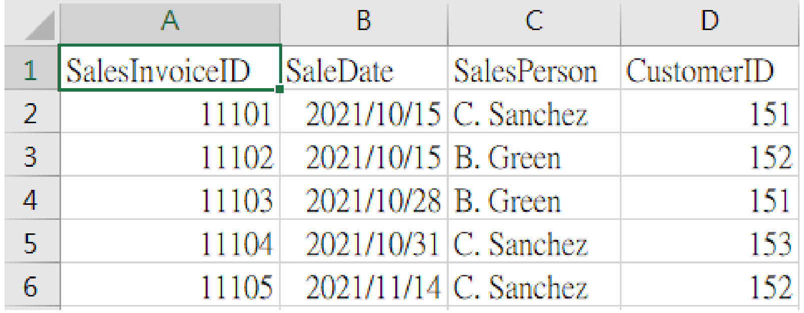
Example



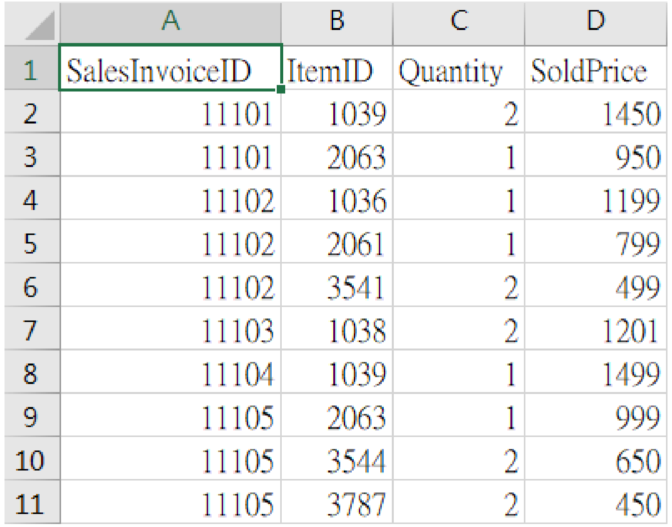
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	SalesInvoiceID	SaleDate	SalesPerson	CustomerID	CustomerName	Street	City	State	ItemID	Description	Color	VendorID	QuantityOnHand	ListPrice	SoldPrice	Quantity	SoldPrice
2	11101	2021/10/15	C. Sanchez	151	Vivian Rodgers	204 NoContent Street	Phoenix	AZ	1039	Refrigerator	Stainless	30023	5	1499	1450	2	1450
3	11101	2021/10/15	C. Sanchez	151	Vivian Rodgers	204 NoContent Street	Phoenix	AZ	2063	Range	Stainless	30011	5	999	950	1	950
4	11102	2021/10/15	B. Green	152	Lola Doyle	504 Gateway Place	Mesa	AZ	1036	Refrigerator	White	30023	12	1199	1199	1	1199
5	11102	2021/10/15	B. Green	152	Lola Doyle	504 Gateway Place	Mesa	AZ	2061	Range	White	30011	6	799	799	1	799
6	11102	2021/10/15	B. Green	152	Lola Doyle	504 Gateway Place	Mesa	AZ	3541	Washer	White	30008	15	499	499	2	499
7	11103	2021/10/28	B. Green	151	Vivian Rodgers	204 NoContent Street	Phoenix	AZ	1038	Refrigerator	Black	30023	7	1299	1201	2	1201
8	11104	2021/10/31	C. Sanchez	153	Rodney Wern	500 Serverr Place	Chandler	AZ	1039	Refrigerator	Stainless	30023	5	1499	1499	1	1499
9	11105	2021/11/14	C. Sanchez	152	Lola Doyle	504 Gateway Place	Mesa	AZ	2063	Range	Stainless	30011	5	999	999	1	999
10	11105	2021/11/14	C. Sanchez	152	Lola Doyle	504 Gateway Place	Mesa	AZ	3544	Washer	Black	30008	10	699	650	2	650
11	11105	2021/11/14	C. Sanchez	152	Lola Doyle	504 Gateway Place	Mesa	AZ	3787	Dryer	Black	30019	8	499	450	2	450

# of Cells = 10 (rows) * 17 (columns) = 170

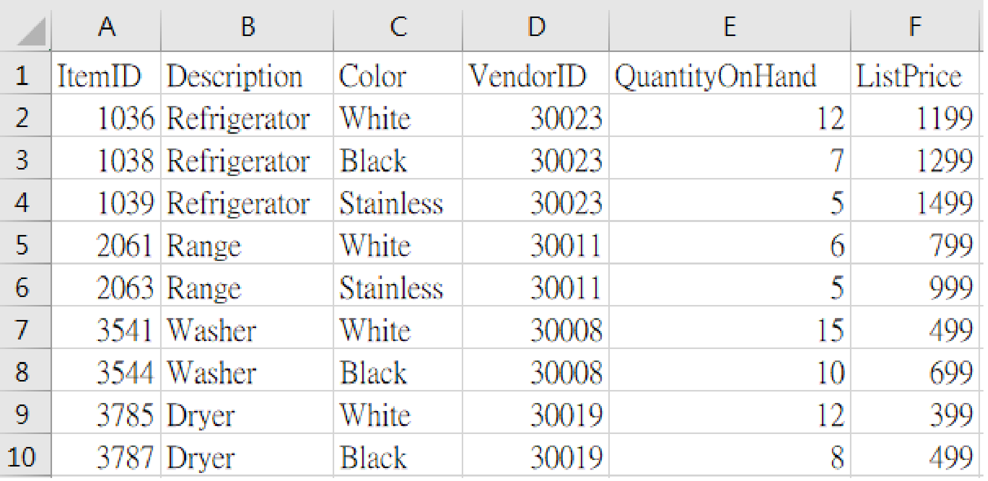
Example



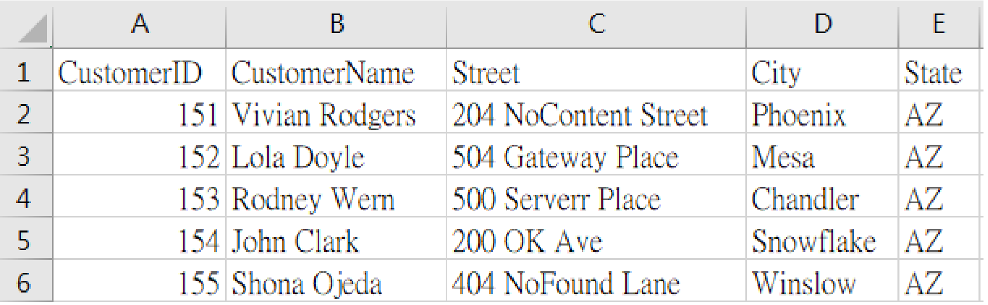
	A	B	C	D
1	SalesInvoiceID	SaleDate	SalesPerson	CustomerID
2	11101	2021/10/15	C. Sanchez	151
3	11102	2021/10/15	B. Green	152
4	11103	2021/10/28	B. Green	151
5	11104	2021/10/31	C. Sanchez	153
6	11105	2021/11/14	C. Sanchez	152



	A	B	C	D
1	SalesInvoiceID	ItemID	Quantity	SoldPrice
2	11101	1039	2	1450
3	11101	2063	1	950
4	11102	1036	1	1199
5	11102	2061	1	799
6	11102	3541	2	499
7	11103	1038	2	1201
8	11104	1039	1	1499
9	11105	2063	1	999
10	11105	3544	2	650
11	11105	3787	2	450



	A	B	C	D	E	F
1	ItemID	Description	Color	VendorID	QuantityOnHand	ListPrice
2	1036	Refrigerator	White	30023	12	1199
3	1038	Refrigerator	Black	30023	7	1299
4	1039	Refrigerator	Stainless	30023	5	1499
5	2061	Range	White	30011	6	799
6	2063	Range	Stainless	30011	5	999
7	3541	Washer	White	30008	15	499
8	3544	Washer	Black	30008	10	699
9	3785	Dryer	White	30019	12	399
10	3787	Dryer	Black	30019	8	499

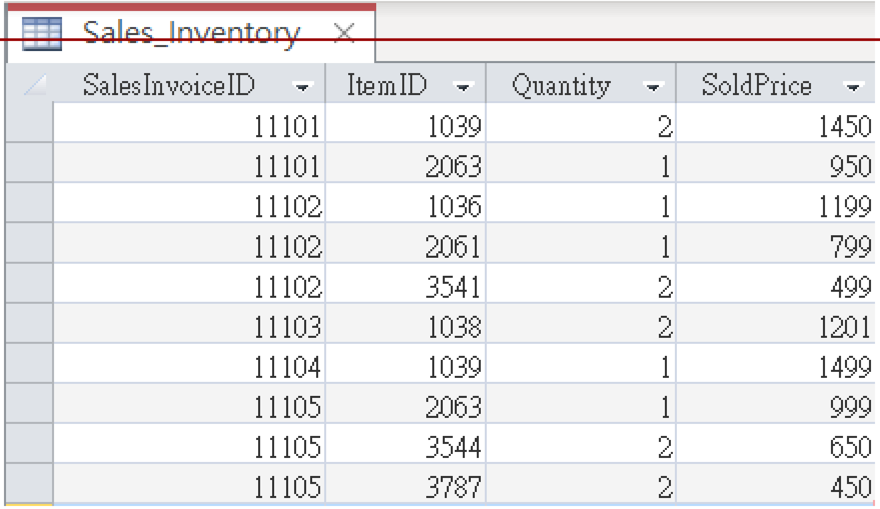


	A	B	C	D	E
1	CustomerID	CustomerName	Street	City	State
2	151	Vivian Rodgers	204 NoContent Street	Phoenix	AZ
3	152	Lola Doyle	504 Gateway Place	Mesa	AZ
4	153	Rodney Wern	500 Serverr Place	Chandler	AZ
5	154	John Clark	200 OK Ave	Snowflake	AZ
6	155	Shona Ojeda	404 NoFound Lane	Winslow	AZ

# of Cells = 20 + 40 + 54 + 25 = 139

Table, Column, and Row

Table	Column	Row
Relation	Attribute	Tuple
File	Field	Record



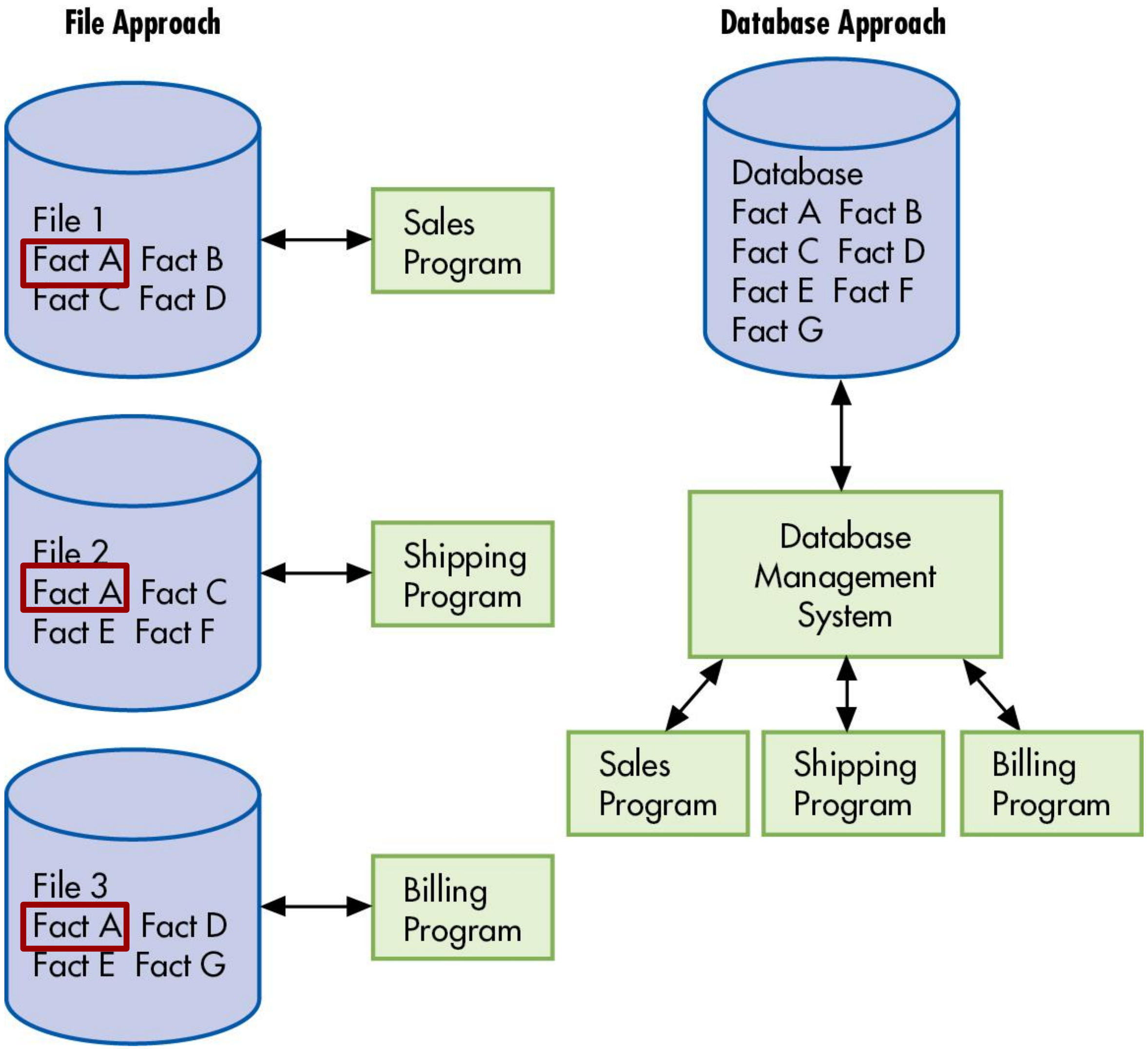
Sales_Inventory

SalesInvoiceID	ItemID	Quantity	SoldPrice
11101	1039	2	1450
11101	2063	1	950
11102	1036	1	1199
11102	2061	1	799
11102	3541	2	499
11103	1038	2	1201
11104	1039	1	1499
11105	2063	1	999
11105	3544	2	650
11105	3787	2	450

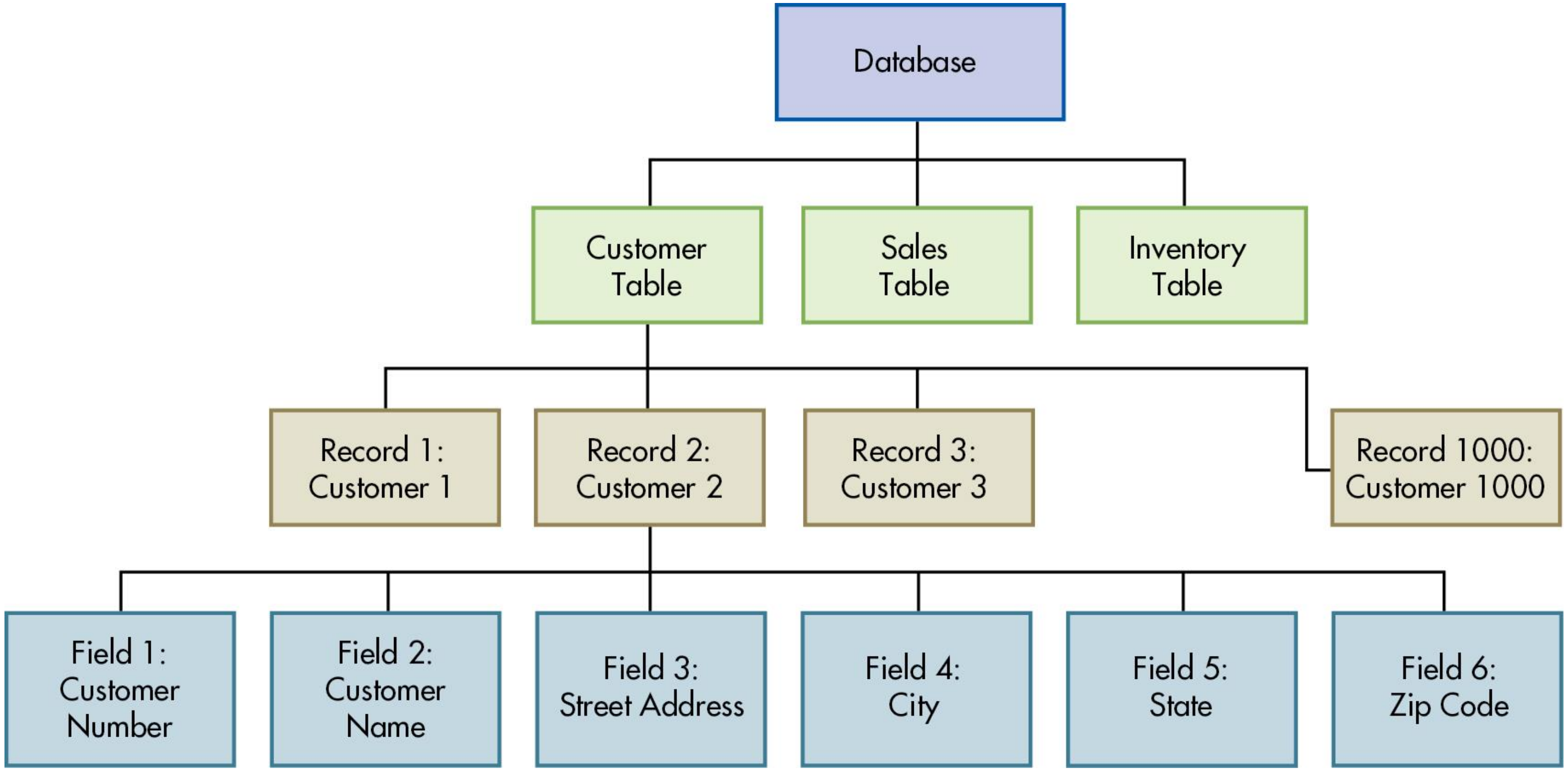
D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

File-Oriented Systems vs. Database Systems

Redundancy!



Basic Elements of Data Hierarchy



Tables

Rows

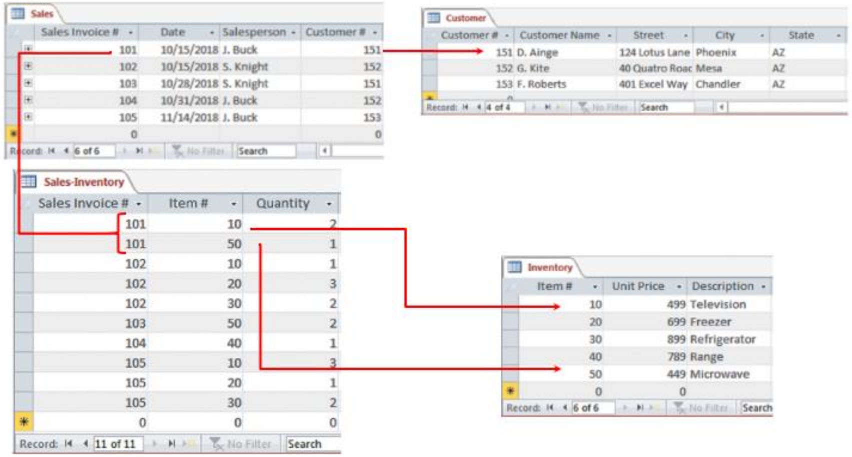
Columns

Advantages of Databases

- Data integration
- Data sharing
- Cross-functional analysis
- Minimal data redundancy and data inconsistencies

Relational Database

- A relational database is **a collection of 2D tables** with each table representing an object about which we wish to collect and store information.



Sales

Sales Invoice #	Date	Salesperson	Customer #
101	10/15/2018	J. Buck	151
102	10/15/2018	S. Knight	152
103	10/28/2018	S. Knight	151
104	10/31/2018	J. Buck	152
105	11/14/2018	J. Buck	153
0			0

Customer

Customer #	Customer Name	Street	City	State
151	D. Ainge	124 Lotus Lane	Phoenix	AZ
152	G. Kite	40 Quatro Roac	Mesa	AZ
153	F. Roberts	401 Excel Way	Chandler	AZ
0				

Sales-Inventory

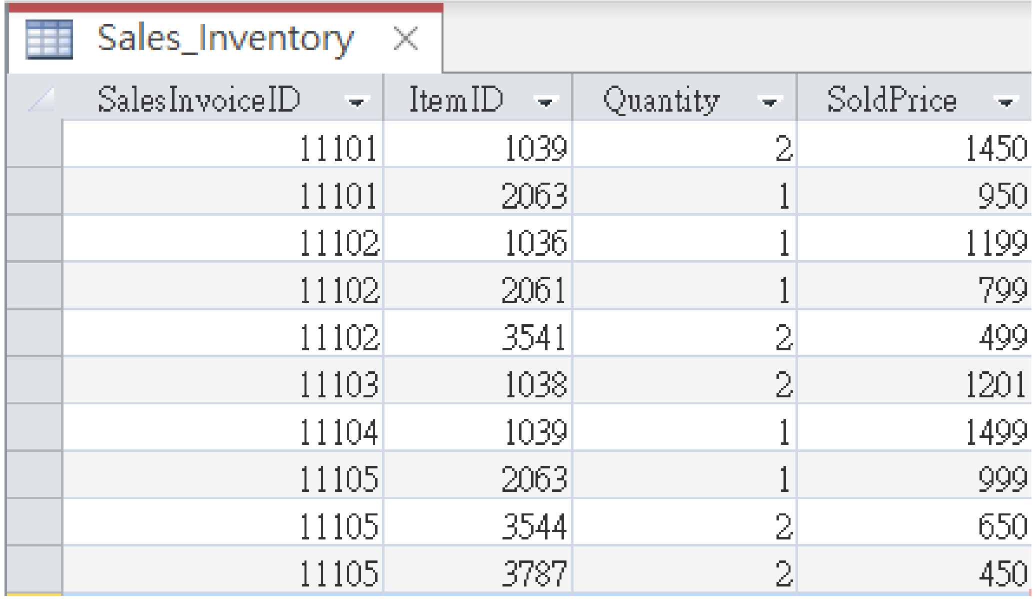
Sales Invoice #	Item #	Quantity
101	10	2
101	50	1
102	10	1
102	20	3
102	30	2
103	50	2
104	40	1
105	10	3
105	20	1
105	30	2
0	0	0

Inventory

Item #	Unit Price	Description
10	499	Television
20	699	Freezer
30	899	Refrigerator
40	789	Range
50	449	Microwave
0	0	

- Although the conceptual view appears to the user that this information is in one big table, it really is **a set of tables that relate to one another**.

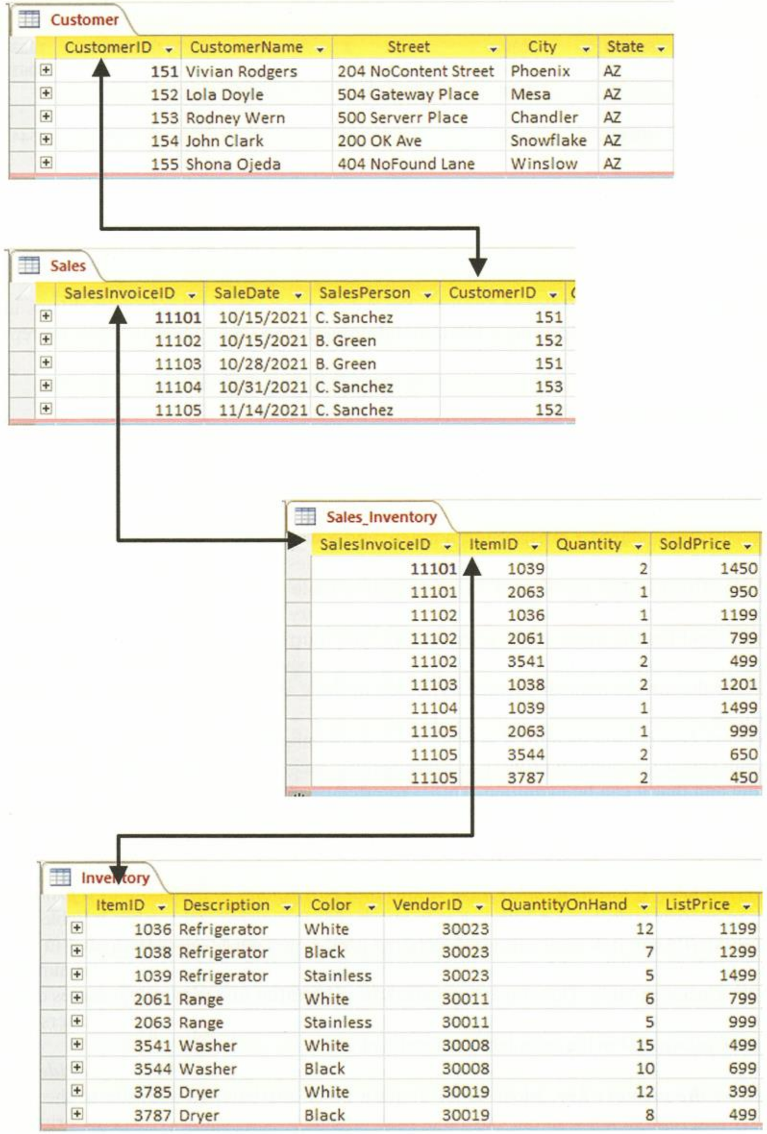
Relational Database



Sales_Inventory

SalesInvoiceID	ItemID	Quantity	SoldPrice
11101	1039	2	1450
11101	2063	1	950
11102	1036	1	1199
11102	2061	1	799
11102	3541	2	499
11103	1038	2	1201
11104	1039	1	1499
11105	2063	1	999
11105	3544	2	650
11105	3787	2	450

Set of Relational Tables



Customer

CustomerID	CustomerName	Street	City	State
151	Vivian Rodgers	204 NoContent Street	Phoenix	AZ
152	Lola Doyle	504 Gateway Place	Mesa	AZ
153	Rodney Wern	500 Serverr Place	Chandler	AZ
154	John Clark	200 OK Ave	Snowflake	AZ
155	Shona Ojeda	404 NoFound Lane	Winslow	AZ

Sales

SalesInvoiceID	SaleDate	SalesPerson	CustomerID
11101	10/15/2021	C. Sanchez	151
11102	10/15/2021	B. Green	152
11103	10/28/2021	B. Green	151
11104	10/31/2021	C. Sanchez	153
11105	11/14/2021	C. Sanchez	152

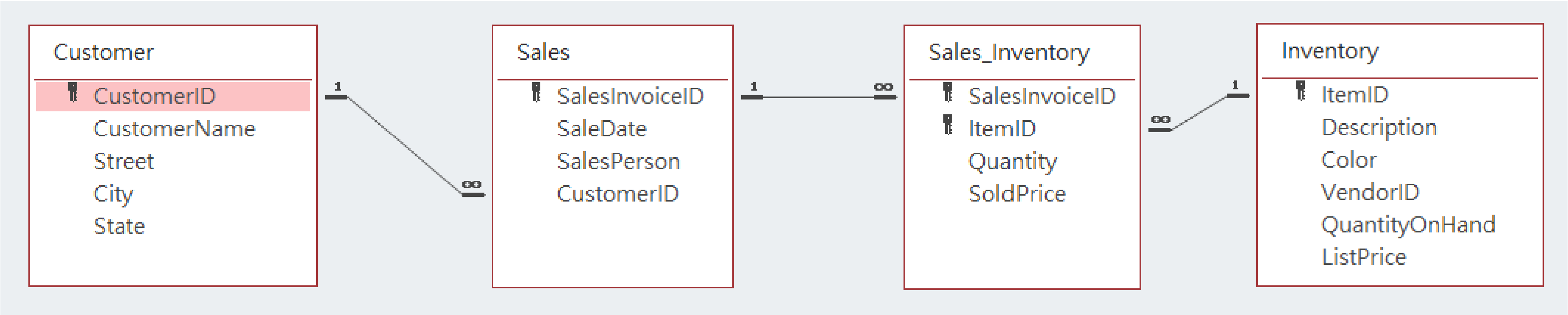
Sales_Inventory

SalesInvoiceID	ItemID	Quantity	SoldPrice
11101	1039	2	1450
11101	2063	1	950
11102	1036	1	1199
11102	2061	1	799
11102	3541	2	499
11103	1038	2	1201
11104	1039	1	1499
11105	2063	1	999
11105	3544	2	650
11105	3787	2	450

Inventory

ItemID	Description	Color	VendorID	QuantityOnHand	ListPrice
1036	Refrigerator	White	30023	12	1199
1038	Refrigerator	Black	30023	7	1299
1039	Refrigerator	Stainless	30023	5	1499
2061	Range	White	30011	6	799
2063	Range	Stainless	30011	5	999
3541	Washer	White	30008	15	499
3544	Washer	Black	30008	10	699
3785	Dryer	White	30019	12	399
3787	Dryer	Black	30019	8	499

Example

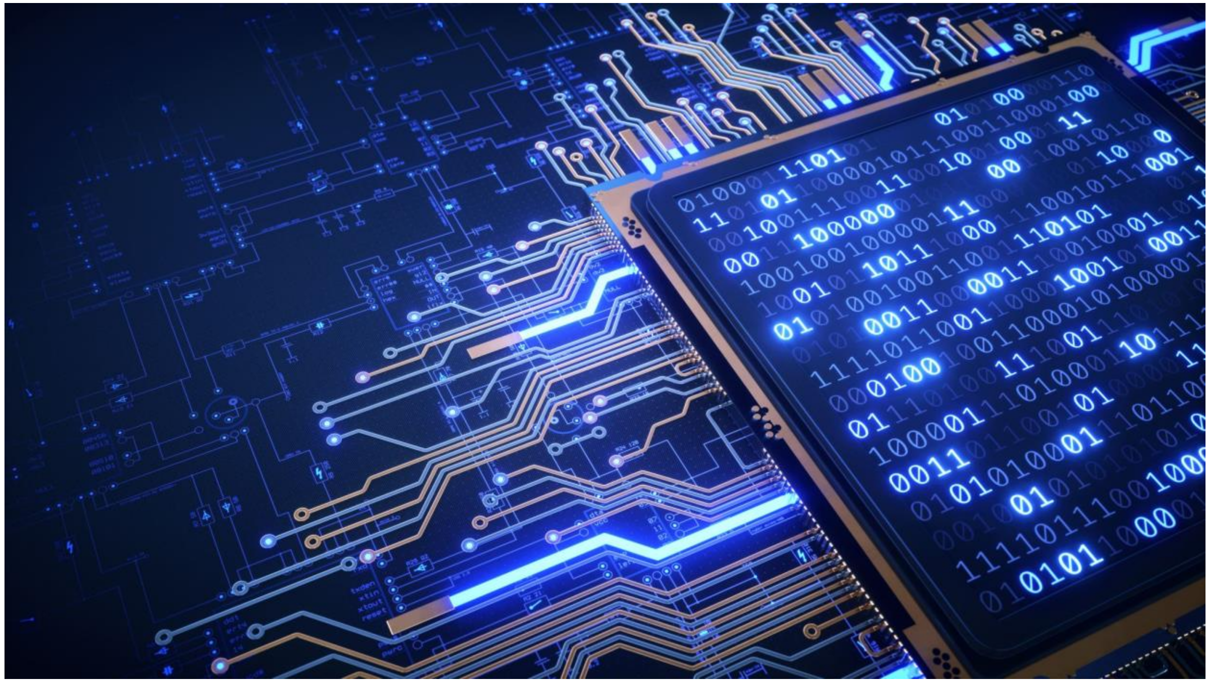


Customer
- CustomerID
- CustomerName
- Street
- City
- State

Sales
- SalesInvoiceID
- SaleDate
- SalesPerson
- CustomerID

Sales_Inventory
- SalesInvoiceID
- ItemID
- Quantity
- SoldPrice

Inventory
- ItemID
- Description
- Color
- VendorID
- QuantityOnHand
- ListPrice



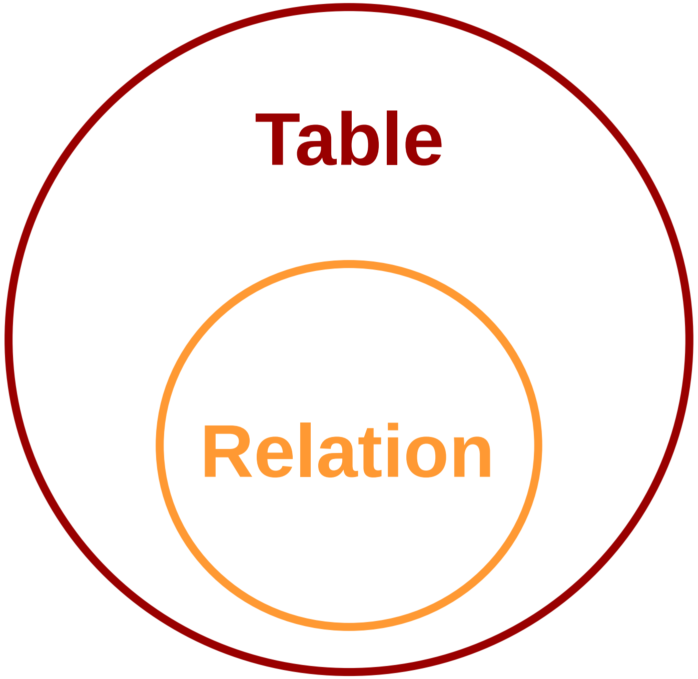
More Details about the Relational Database Design

Important Terms

- Relation
- Determinant
- Functional Dependency
- Candidate Key
- Composite Key
- Primary Key
- Foreign Key
- Referential Integrity Constraint

“Relation”

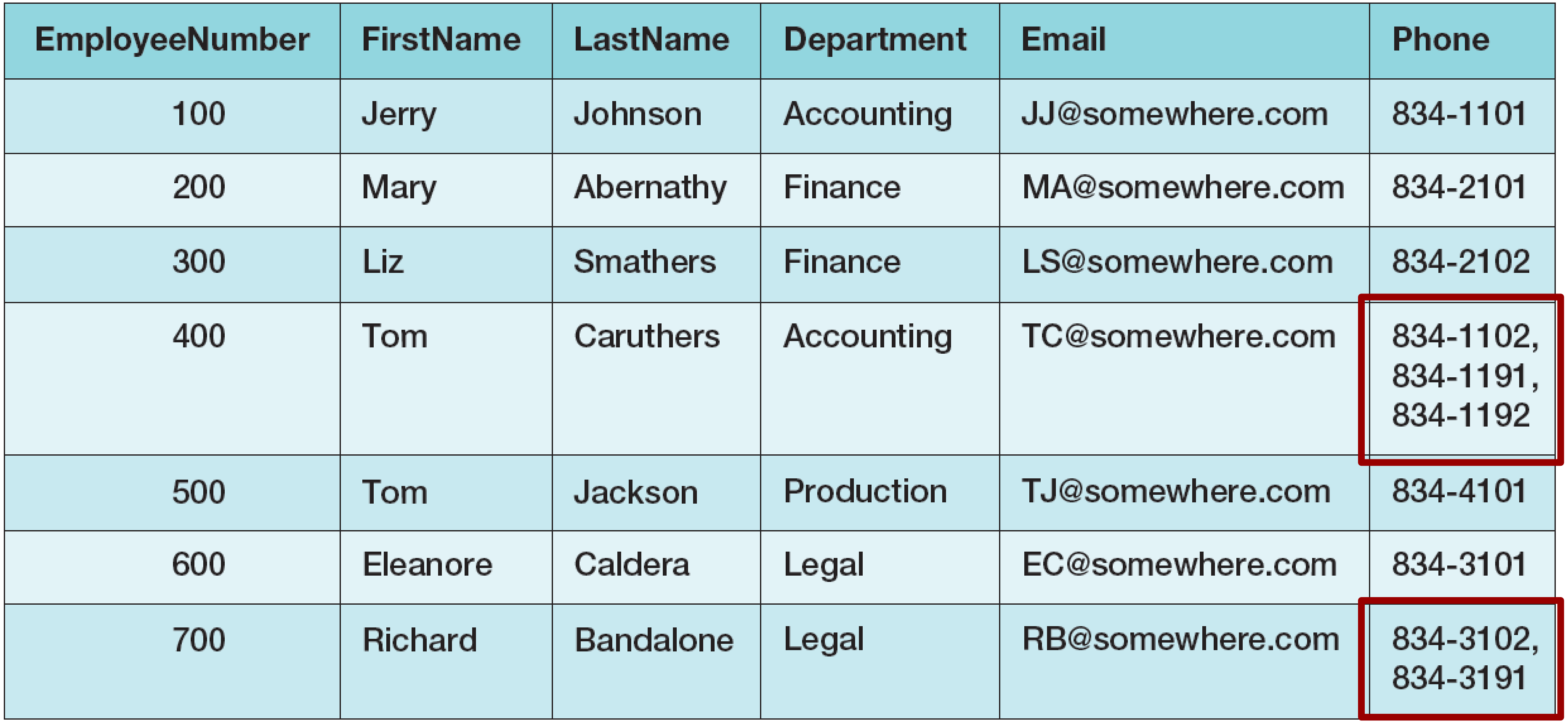
- Sometimes, people use the term ***table*** and ***relation*** interchangeably.
- However, a ***relation*** is a special case of a ***table***.



Characteristics of Relations

- **No two rows may be identical.**
- The order of the rows is **irrelevant**.
- The order of the columns is **irrelevant**.
- Cells of the table hold a **single** value.
- No two columns in the same relation may have the same name; columns in different relations may have the same name.
- All entries in a column are of the same kind. (**Domain Integrity Constraint**)
- Rows contain data about an entity.
- Columns contain data about attributes of the entities.

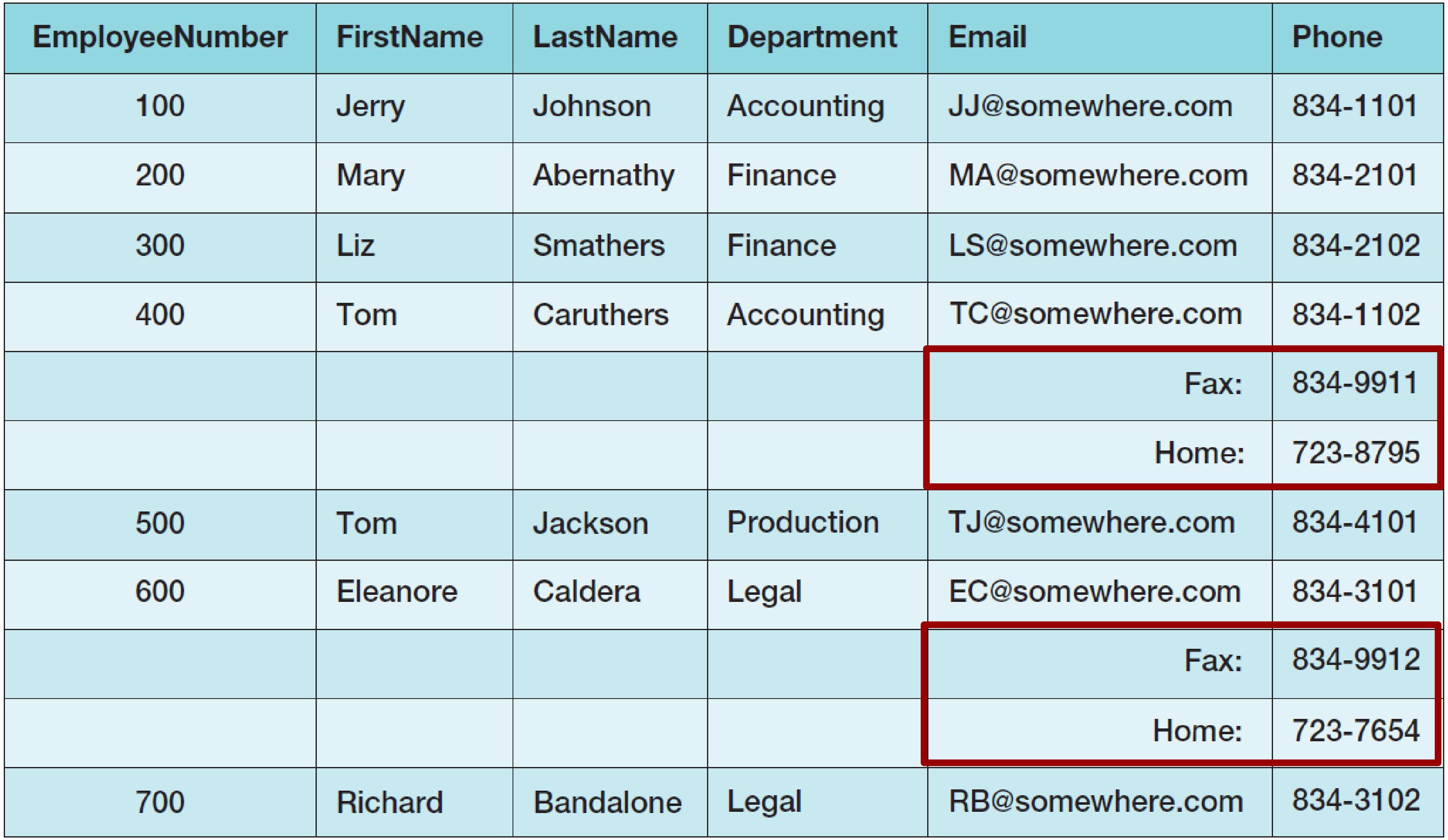
Why is it NOT a relation?



EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102, 834-1191, 834-1192
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102, 834-3191

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Why is it NOT a relation?



EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102
				Fax:	834-9911
				Home:	723-8795
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101
				Fax:	834-9912
				Home:	723-7654
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Functional Dependency

CokeCost = NumberOfCans X $20

- CokeCost **depends on** NumberOfCans.
- CokeCost **is functionally dependent on** NumberOfCans.

NumberOfCans → CokeCost

NumberOfCans: **determinant**

Functional Dependency

TotalRevenue = Quantity X UnitPrice

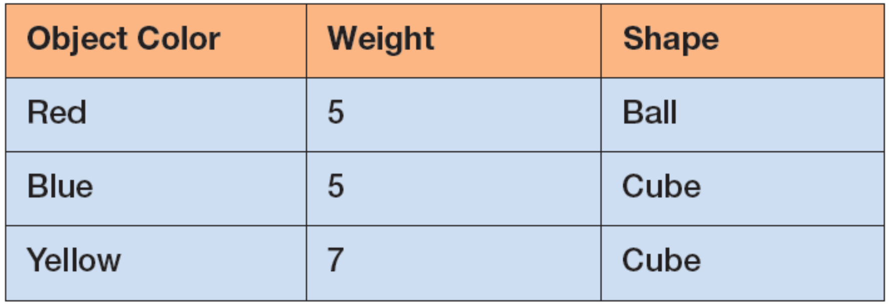
- TotalRevenue **depends on** Quantity & UnitPrice.
- TotalRevenue **is functionally dependent on** Quantity & UnitPrice.

(Quantity, UnitPrice) → TotalRevenue

Composite determinant

Functional Dependency

- Functional dependencies do not necessarily involve equations.



Object Color	Weight	Shape
Red	5	Ball
Blue	5	Cube
Yellow	7	Cube

Object → Weight

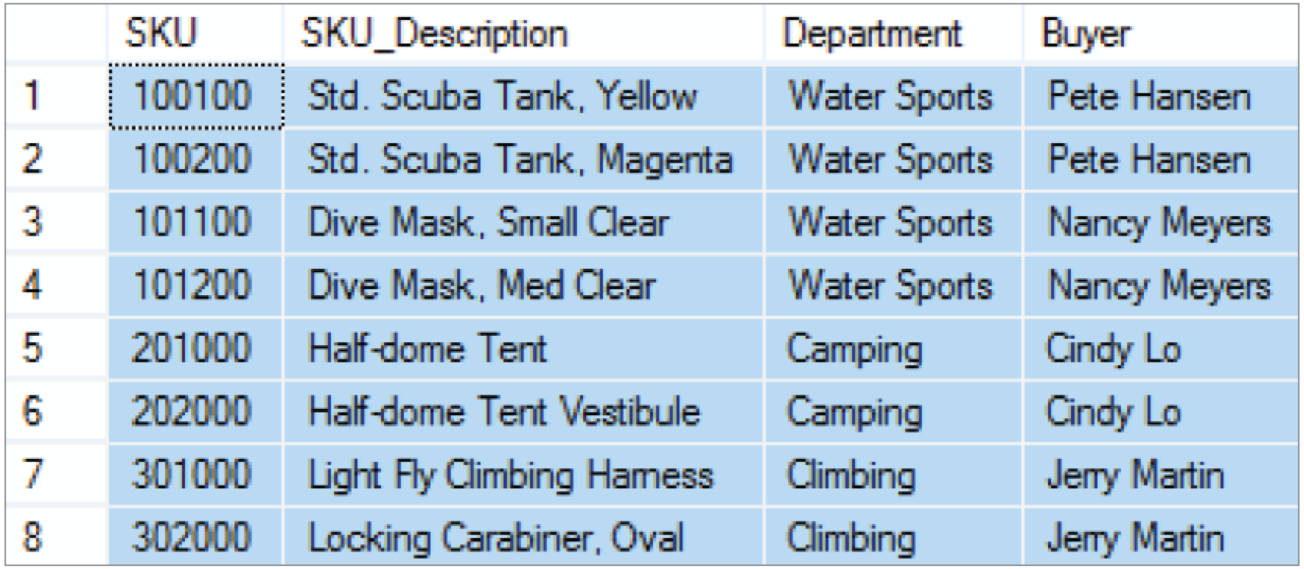
Object → Shape

Object → (Weight, Shape) (**Union Rule & Decomposition Rule**)

- The only reason for having **relations** is **to store instances of functional dependencies**.

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Finding Functional Dependency

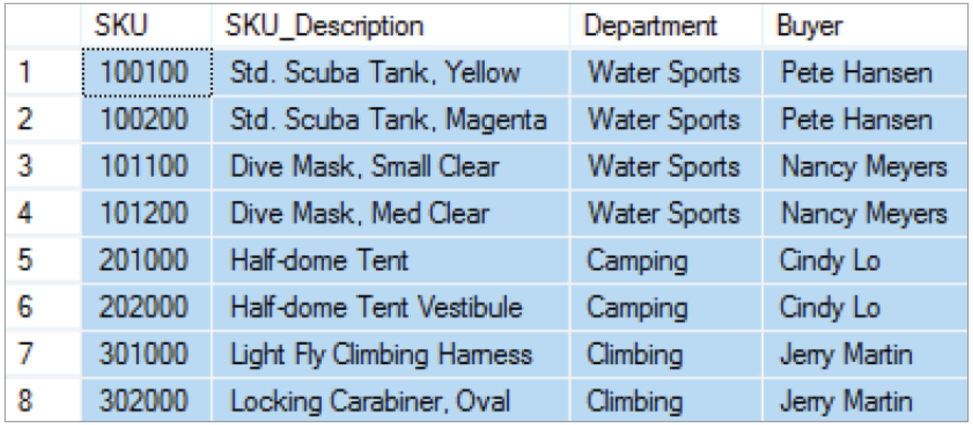


	SKU	SKU_Description	Department	Buyer
1	100100	Std. Scuba Tank, Yellow	Water Sports	Pete Hansen
2	100200	Std. Scuba Tank, Magenta	Water Sports	Pete Hansen
3	101100	Dive Mask, Small Clear	Water Sports	Nancy Meyers
4	101200	Dive Mask, Med Clear	Water Sports	Nancy Meyers
5	201000	Half-dome Tent	Camping	Cindy Lo
6	202000	Half-dome Tent Vestibule	Camping	Cindy Lo
7	301000	Light Fly Climbing Harness	Climbing	Jerry Martin
8	302000	Locking Carabiner, Oval	Climbing	Jerry Martin

SKU_DATA

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Finding Functional Dependency



	SKU	SKU_Description	Department	Buyer
1	100100	Std. Scuba Tank, Yellow	Water Sports	Pete Hansen
2	100200	Std. Scuba Tank, Magenta	Water Sports	Pete Hansen
3	101100	Dive Mask, Small Clear	Water Sports	Nancy Meyers
4	101200	Dive Mask, Med Clear	Water Sports	Nancy Meyers
5	201000	Half-dome Tent	Camping	Cindy Lo
6	202000	Half-dome Tent Vestibule	Camping	Cindy Lo
7	301000	Light Fly Climbing Harness	Climbing	Jerry Martin
8	302000	Locking Carabiner, Oval	Climbing	Jerry Martin

SKU → SKU_Description

SKU → Department

SKU → Buyer

SKU → (SKU_Description, Department, Buyer)

SKU_Description → SKU

SKU_Description → Department

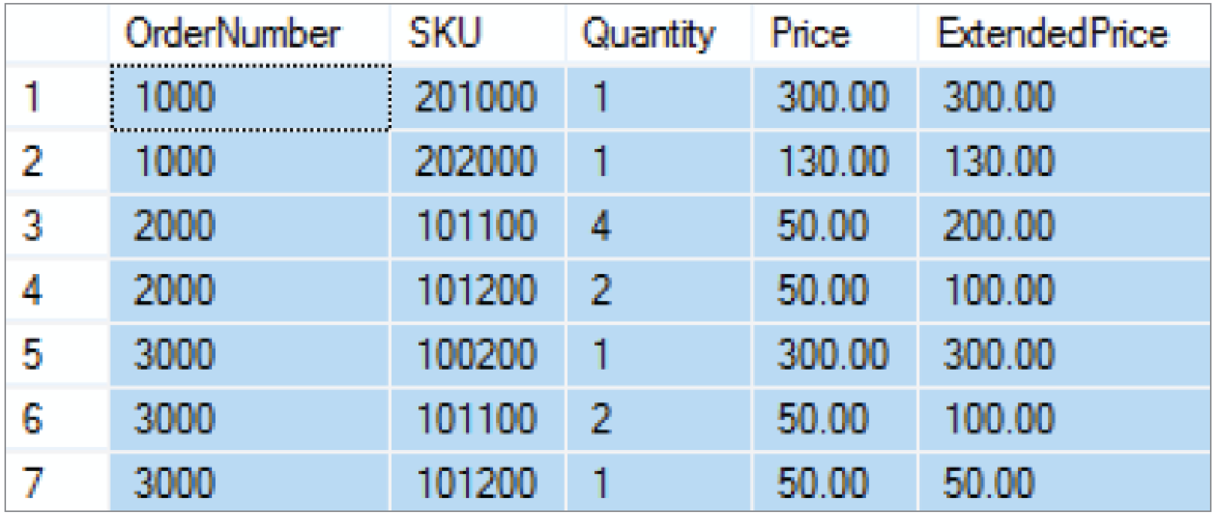
SKU_Description → Buyer

SKU_Description → (SKU, Department, Buyer)

Buyer → Department

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Finding Functional Dependency

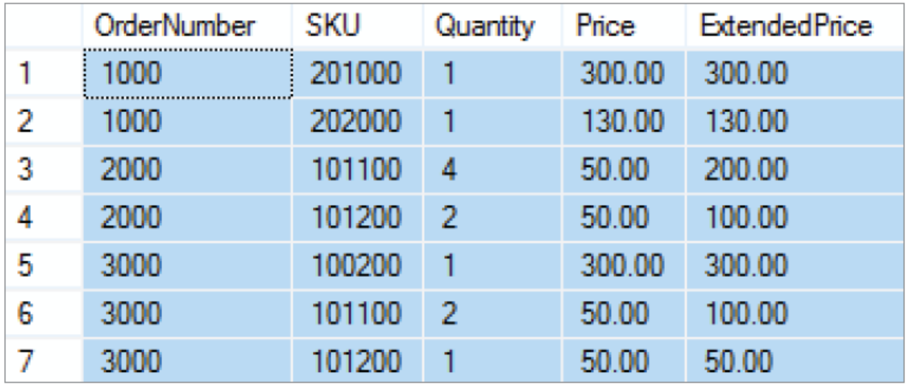


	OrderNumber	SKU	Quantity	Price	ExtendedPrice
1	1000	201000	1	300.00	300.00
2	1000	202000	1	130.00	130.00
3	2000	101100	4	50.00	200.00
4	2000	101200	2	50.00	100.00
5	3000	100200	1	300.00	300.00
6	3000	101100	2	50.00	100.00
7	3000	101200	1	50.00	50.00

ORDER_ITEM

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Finding Functional Dependency



	OrderNumber	SKU	Quantity	Price	ExtendedPrice
1	1000	201000	1	300.00	300.00
2	1000	202000	1	130.00	130.00
3	2000	101100	4	50.00	200.00
4	2000	101200	2	50.00	100.00
5	3000	100200	1	300.00	300.00
6	3000	101100	2	50.00	100.00
7	3000	101200	1	50.00	50.00

SKU → Price

(OrderNumber, SKU) → Price

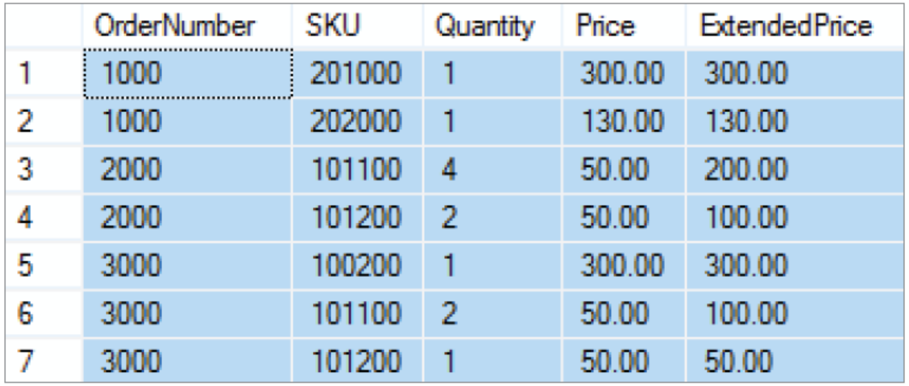
(OrderNumber, SKU) → (Quantity, Price, ExtendedPrice)

(Quantity, Price) → ExtendedPrice

(Quantity, Price) → OrderNumber? SKU? ✖

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

When are Determinant Values Unique?



	OrderNumber	SKU	Quantity	Price	ExtendedPrice
1	1000	201000	1	300.00	300.00
2	1000	202000	1	130.00	130.00
3	2000	101100	4	50.00	200.00
4	2000	101200	2	50.00	100.00
5	3000	100200	1	300.00	300.00
6	3000	101100	2	50.00	100.00
7	3000	101200	1	50.00	50.00

- A determinant is **unique** in a relation **only if** it determines **every other column** in the relation.

(OrderNumber, SKU) → (Quantity, Price, ExtendedPrice)

Unique!

(Quantity, Price) → ExtendedPrice

Not Unique!

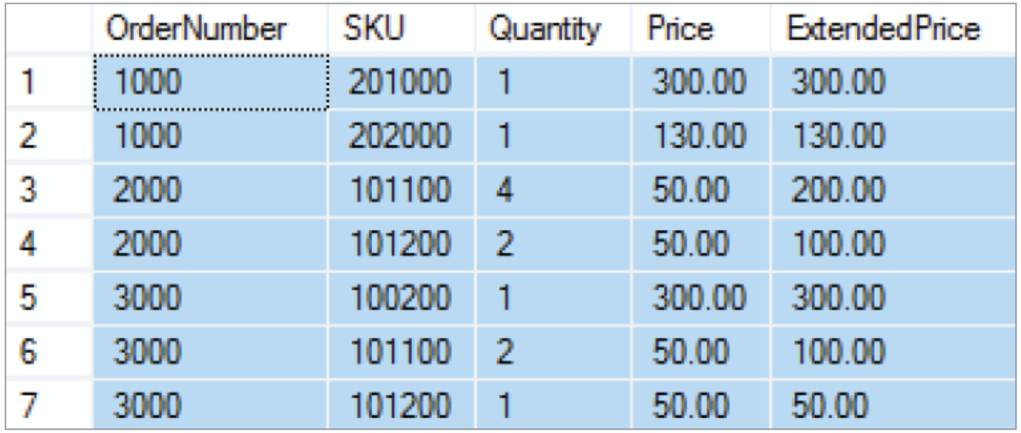
D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Various “Keys”

- Composite Key
- Candidate Key
- Primary Key
- Foreign Key

Composite Key

- **Composite keys** are keys that have **two or more columns**.



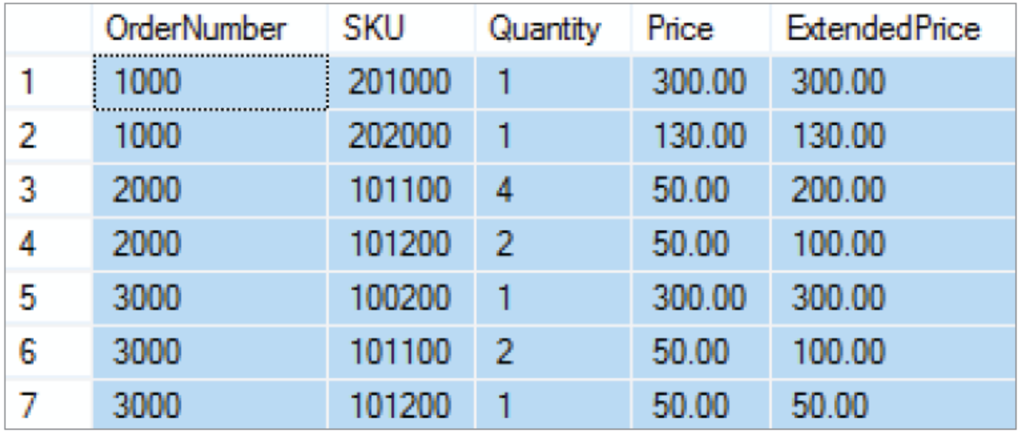
	OrderNumber	SKU	Quantity	Price	ExtendedPrice
1	1000	201000	1	300.00	300.00
2	1000	202000	1	130.00	130.00
3	2000	101100	4	50.00	200.00
4	2000	101200	2	50.00	100.00
5	3000	100200	1	300.00	300.00
6	3000	101100	2	50.00	100.00
7	3000	101200	1	50.00	50.00

(OrderNumber, SKU) → (Quantity, Price, ExtendedPrice)
(Quantity, Price) → ExtendedPrice

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Candidate Key

- A **candidate key** is a determinant that determines **all** of the other columns in a relation.
- Candidate keys can **identify a unique row** in a relation.



	OrderNumber	SKU	Quantity	Price	ExtendedPrice
1	1000	201000	1	300.00	300.00
2	1000	202000	1	130.00	130.00
3	2000	101100	4	50.00	200.00
4	2000	101200	2	50.00	100.00
5	3000	100200	1	300.00	300.00
6	3000	101100	2	50.00	100.00
7	3000	101200	1	50.00	50.00

(OrderNumber, SKU) → (Quantity, Price, ExtendedPrice)

Unique!

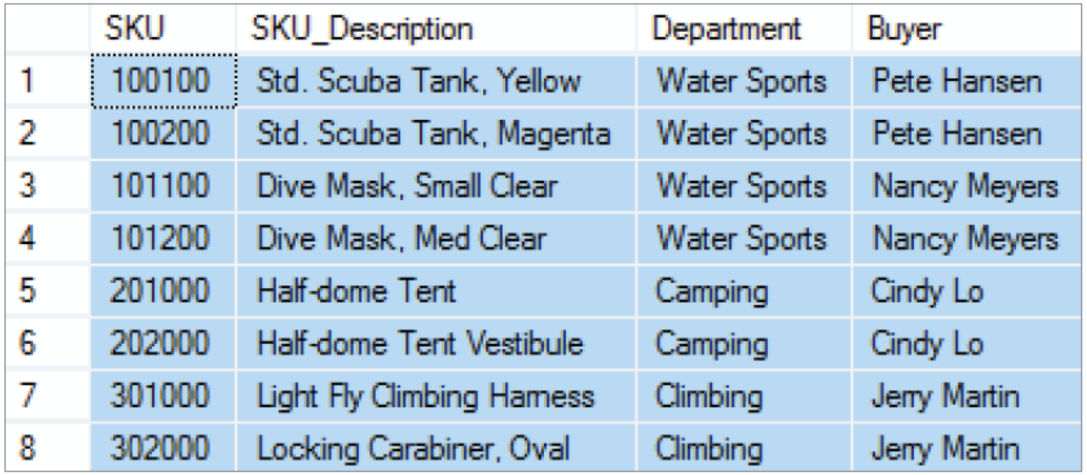
(Quantity, Price) → ExtendedPrice

Not Unique!

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Primary Key

- **One** of the candidate keys is selected as the **primary key** when we design a database.



	SKU	SKU_Description	Department	Buyer
1	100100	Std. Scuba Tank, Yellow	Water Sports	Pete Hansen
2	100200	Std. Scuba Tank, Magenta	Water Sports	Pete Hansen
3	101100	Dive Mask, Small Clear	Water Sports	Nancy Meyers
4	101200	Dive Mask, Med Clear	Water Sports	Nancy Meyers
5	201000	Half-dome Tent	Camping	Cindy Lo
6	202000	Half-dome Tent Vestibule	Camping	Cindy Lo
7	301000	Light Fly Climbing Harness	Climbing	Jerry Martin
8	302000	Locking Carabiner, Oval	Climbing	Jerry Martin

Buyer → Department

SKU → SKU_Description

SKU → Department

SKU → Buyer

SKU → (SKU_Description, Department, Buyer)

SKU_Description → SKU

SKU_Description → Department

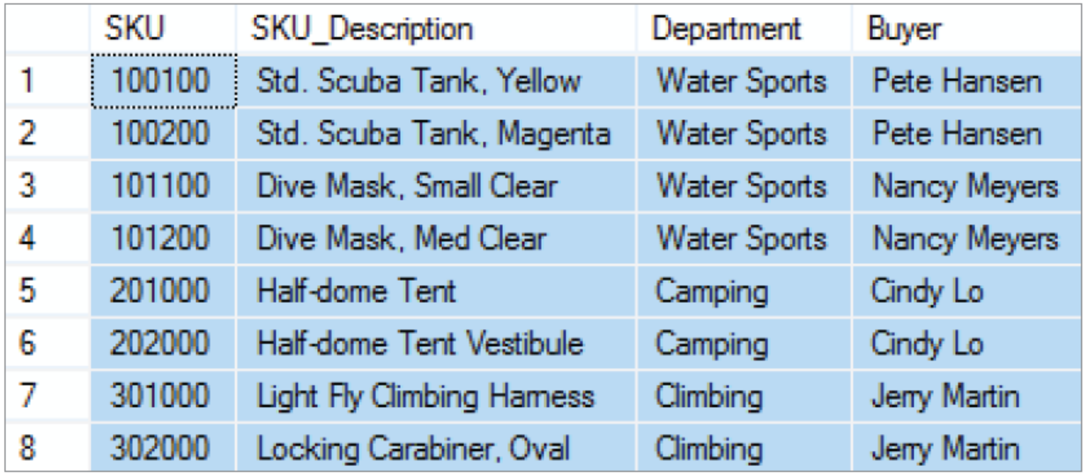
SKU_Description → Buyer

SKU_Description → (SKU, Department, Buyer)

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Primary Key

- **One** of the candidate keys is selected as the **primary key** when we design a database.



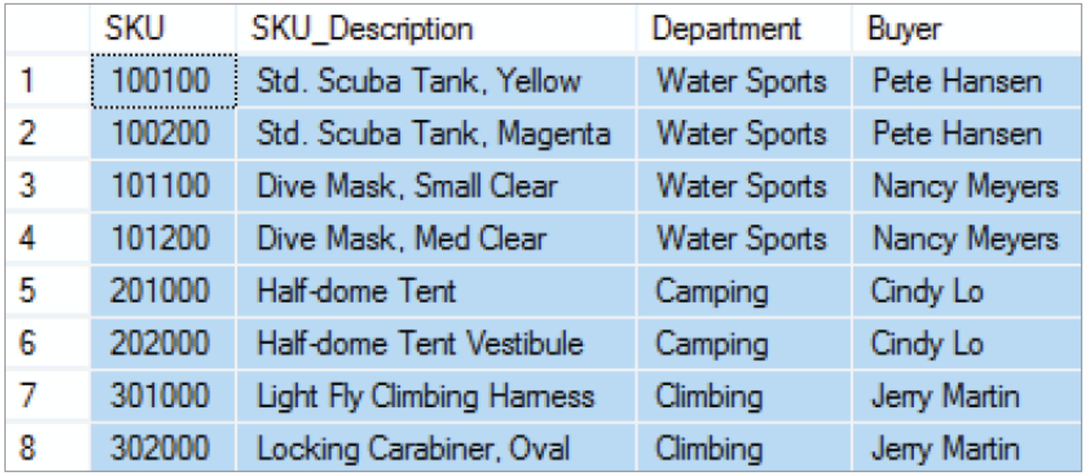
	SKU	SKU_Description	Department	Buyer
1	100100	Std. Scuba Tank, Yellow	Water Sports	Pete Hansen
2	100200	Std. Scuba Tank, Magenta	Water Sports	Pete Hansen
3	101100	Dive Mask, Small Clear	Water Sports	Nancy Meyers
4	101200	Dive Mask, Med Clear	Water Sports	Nancy Meyers
5	201000	Half-dome Tent	Camping	Cindy Lo
6	202000	Half-dome Tent Vestibule	Camping	Cindy Lo
7	301000	Light Fly Climbing Harness	Climbing	Jerry Martin
8	302000	Locking Carabiner, Oval	Climbing	Jerry Martin

Which one would you choose, SKU or SKU_Description?

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Primary Key

- **One** of the candidate keys is selected as the **primary key** when we design a database.



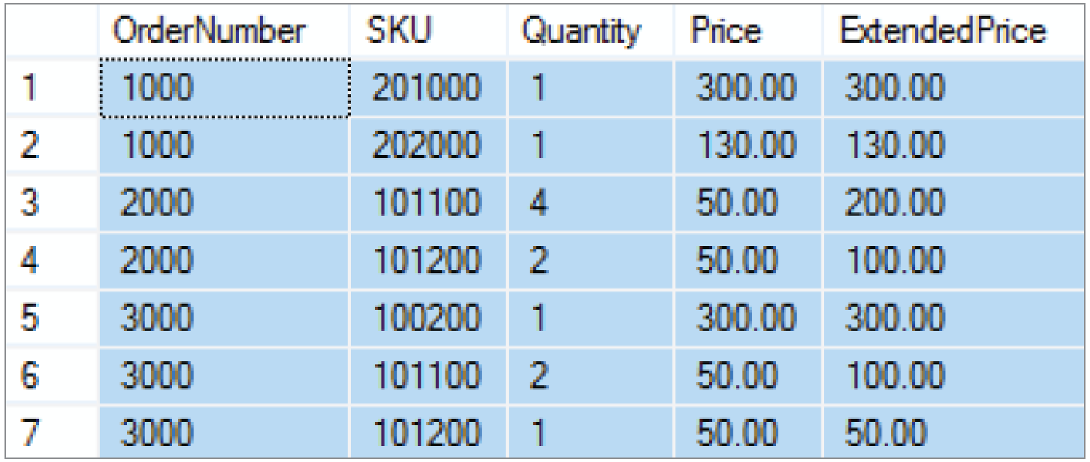
	SKU	SKU_Description	Department	Buyer
1	100100	Std. Scuba Tank, Yellow	Water Sports	Pete Hansen
2	100200	Std. Scuba Tank, Magenta	Water Sports	Pete Hansen
3	101100	Dive Mask, Small Clear	Water Sports	Nancy Meyers
4	101200	Dive Mask, Med Clear	Water Sports	Nancy Meyers
5	201000	Half-dome Tent	Camping	Cindy Lo
6	202000	Half-dome Tent Vestibule	Camping	Cindy Lo
7	301000	Light Fly Climbing Harness	Climbing	Jerry Martin
8	302000	Locking Carabiner, Oval	Climbing	Jerry Martin

SKU_DATA (SKU, SKU_Description, Department, Buyer)

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Primary Key

- **One** of the candidate keys is selected as the **primary key** when we design a database.



	OrderNumber	SKU	Quantity	Price	ExtendedPrice
1	1000	201000	1	300.00	300.00
2	1000	202000	1	130.00	130.00
3	2000	101100	4	50.00	200.00
4	2000	101200	2	50.00	100.00
5	3000	100200	1	300.00	300.00
6	3000	101100	2	50.00	100.00
7	3000	101200	1	50.00	50.00

(OrderNumber, SKU) → (Quantity, Price, ExtendedPrice)

Unique!

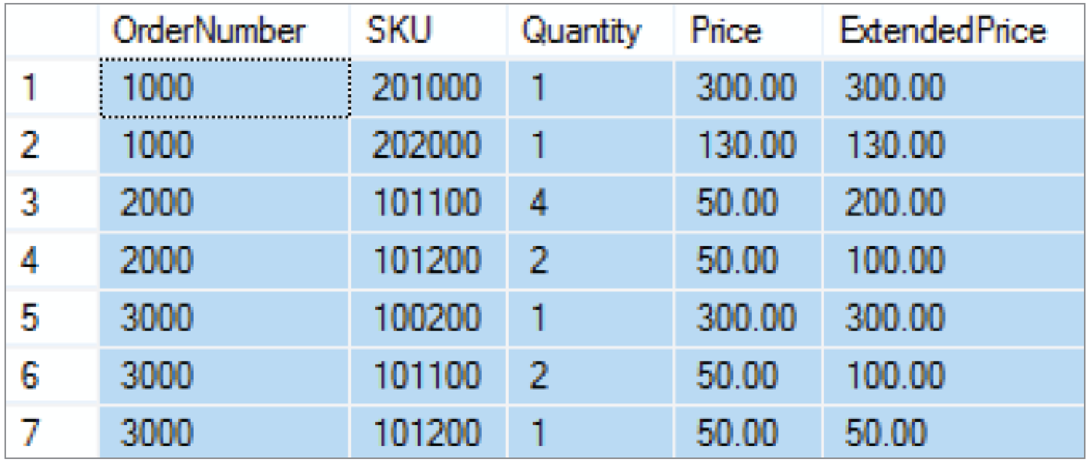
(Quantity, Price) → ExtendedPrice

Not Unique!

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

Primary Key

- **One** of the candidate keys is selected as the **primary key** when we design a database.



	OrderNumber	SKU	Quantity	Price	ExtendedPrice
1	1000	201000	1	300.00	300.00
2	1000	202000	1	130.00	130.00
3	2000	101100	4	50.00	200.00
4	2000	101200	2	50.00	100.00
5	3000	100200	1	300.00	300.00
6	3000	101100	2	50.00	100.00
7	3000	101200	1	50.00	50.00

ORDER_ITEM (OrderNumber, SKU, Quantity, Price, ExtendedPrice)

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

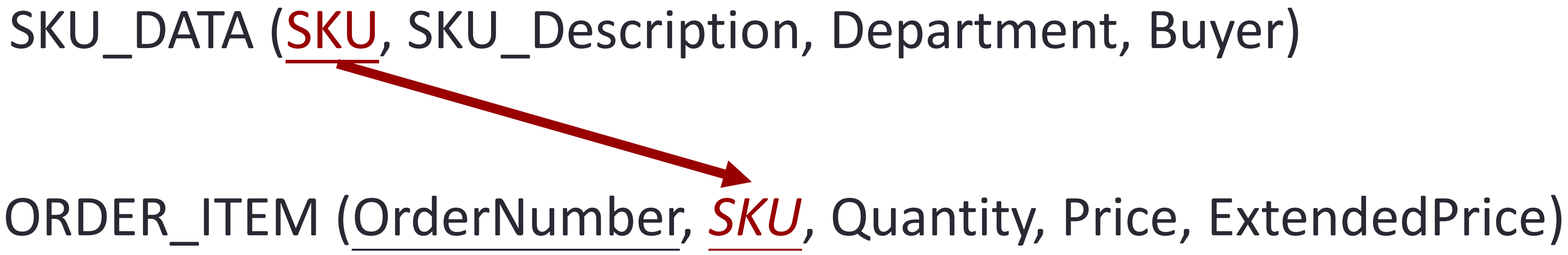
Primary Key

Entity Integrity Constraint

- A **Primary key**, whether it is a single column or a composite key, **must** have **unique** data values inserted into **every row** of the table.
- It is a fundamental requirement for the proper functioning of a relational database.

Foreign Key

- A **foreign key** is a column or composite of columns that is the **primary key of a table** other than the one in which it appears.



SKU_DATA (SKU, SKU_Description, Department, Buyer)

ORDER_ITEM (OrderNumber, *SKU*, Quantity, Price, ExtendedPrice)

- **ORDER_ITEM.SKU** is both a foreign key and part of the primary key of ORDER_ITEM.
  → Sometimes occurs but is not required.

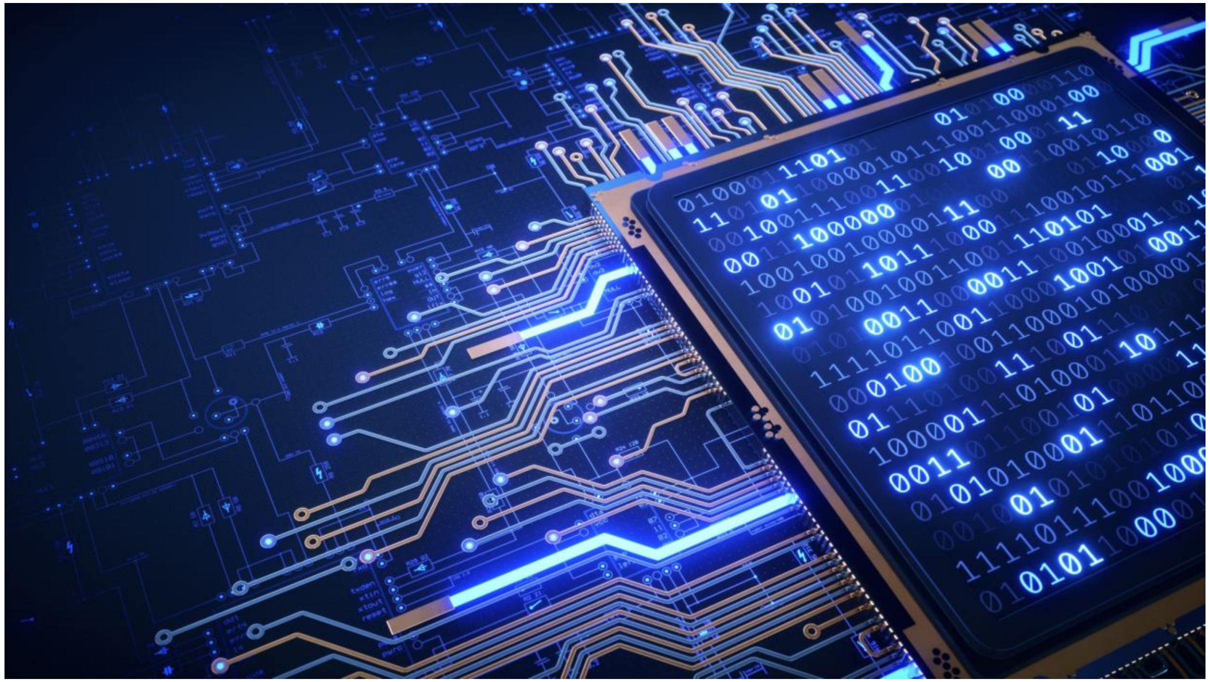
Foreign Key

- Also, we need to ensure that the values of a foreign key match a valid value of a primary key. It is called a **referential integrity constraint**.

SKU_DATA (SKU, SKU_Description, Department, Buyer)

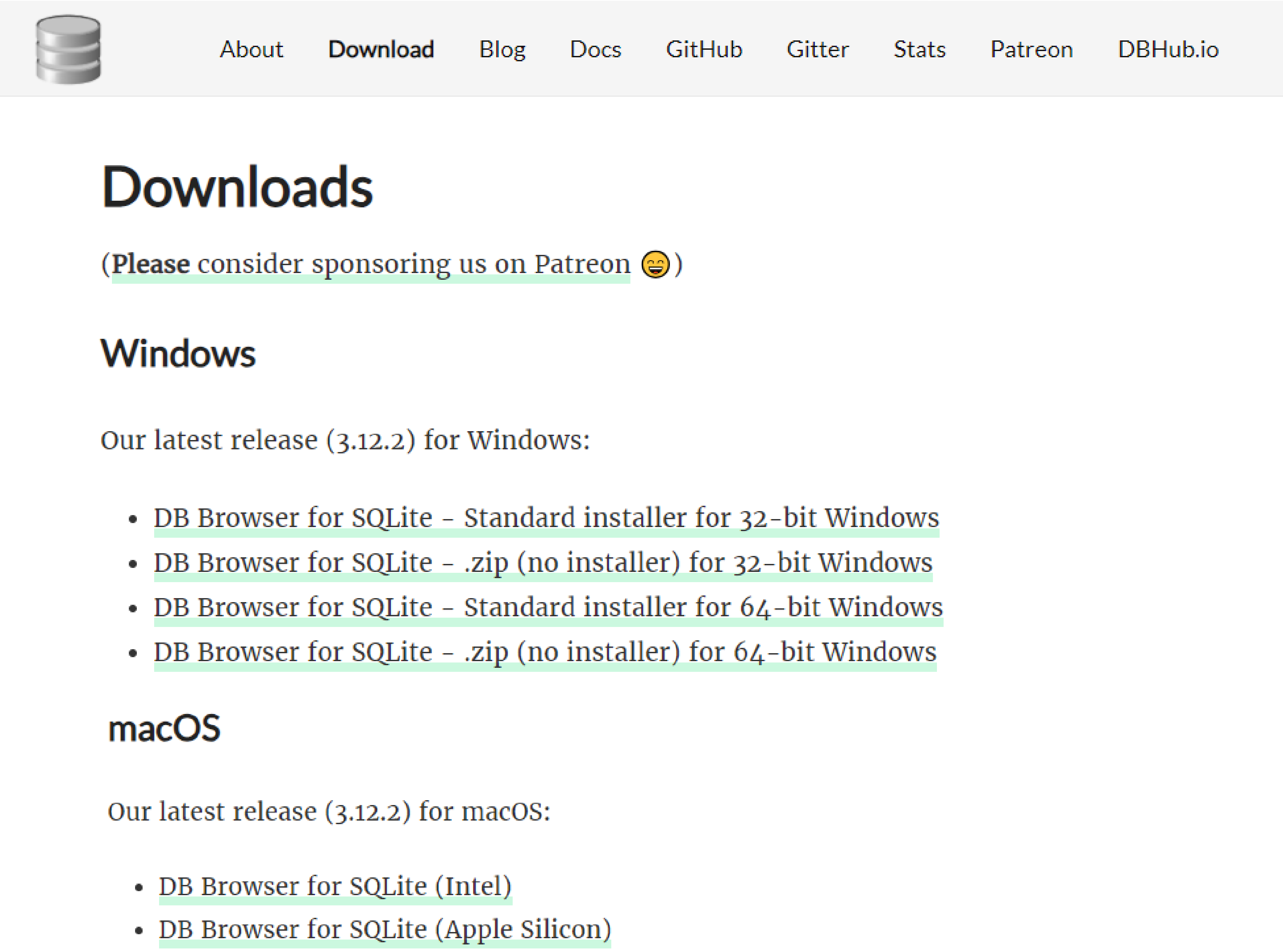
ORDER_ITEM (OrderNumber, *SKU*, Quantity, Price, ExtendedPrice)

- All of the values of ORDER_ITEM.SKU **must exist** in SKU in SKU_DATA.



Structured Query Language (SQL)

DB Browser Download



Downloads

(Please consider sponsoring us on Patreon 😄)

Windows

Our latest release (3.12.2) for Windows:

- DB Browser for SQLite - Standard installer for 32-bit Windows
- DB Browser for SQLite - .zip (no installer) for 32-bit Windows
- DB Browser for SQLite - Standard installer for 64-bit Windows
- DB Browser for SQLite - .zip (no installer) for 64-bit Windows

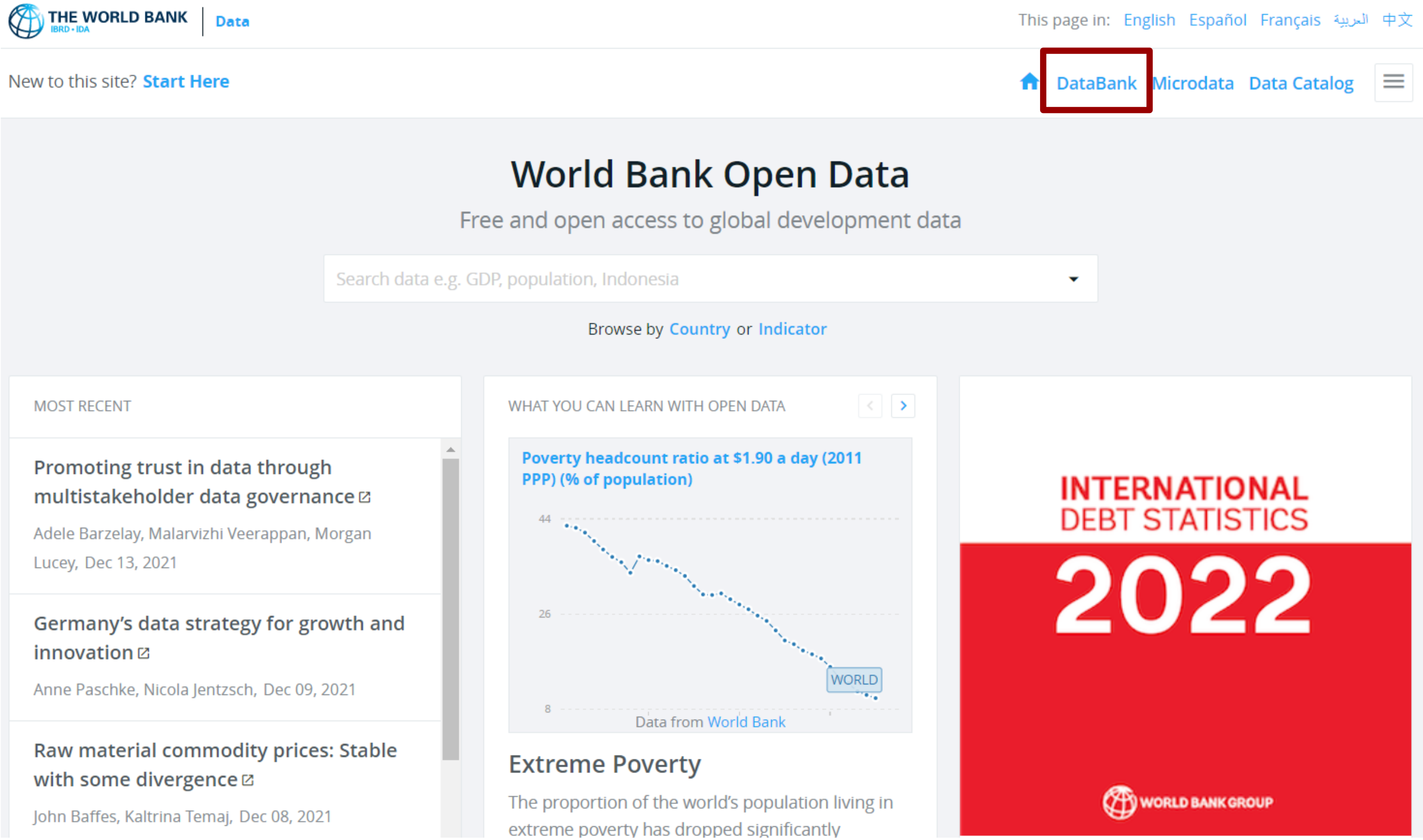
macOS

Our latest release (3.12.2) for macOS:

- DB Browser for SQLite (Intel)
- DB Browser for SQLite (Apple Silicon)

DB Browser for SQLite. Available at: https://sqlitebrowser.org/dl/

The Data We Will Use



THE WORLD BANK IBRD • IDA | Data

This page in: English Español Français العربية 中文

New to this site? Start Here

DataBank Microdata Data Catalog

World Bank Open Data

Free and open access to global development data

Search data e.g. GDP, population, Indonesia

Browse by Country or Indicator

MOST RECENT

Promoting trust in data through multistakeholder data governance

Adele Barzelay, Malarvizhi Veerappan, Morgan Lucey, Dec 13, 2021

Germany's data strategy for growth and innovation

Anne Paschke, Nicola Jentzsch, Dec 09, 2021

Raw material commodity prices: Stable with some divergence

John Baffes, Kaltrina Temaj, Dec 08, 2021

WHAT YOU CAN LEARN WITH OPEN DATA

Poverty headcount ratio at $1.90 a day (2011 PPP) (% of population)

Data from World Bank

Extreme Poverty

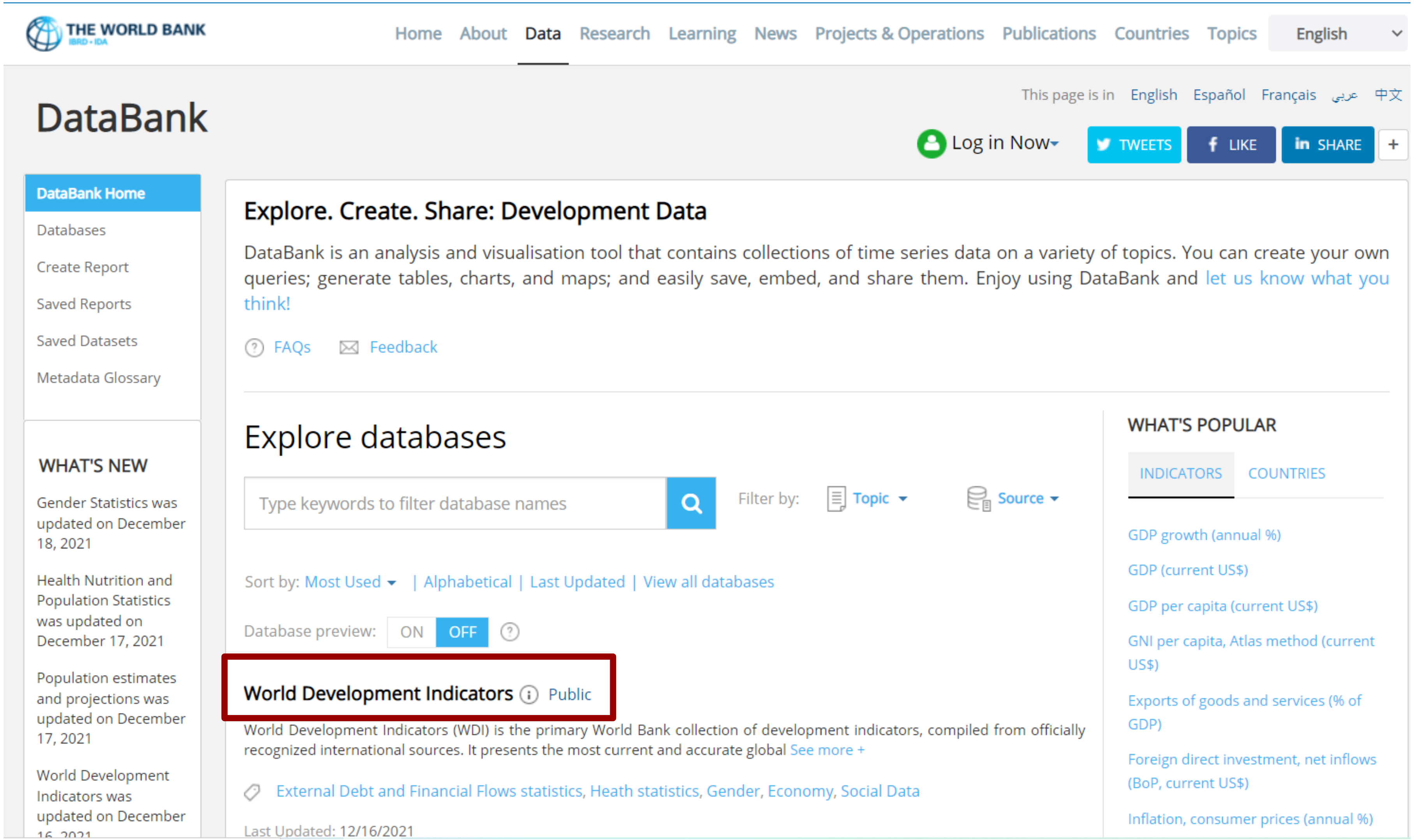
The proportion of the world's population living in extreme poverty has dropped significantly

INTERNATIONAL DEBT STATISTICS 2022

WORLD BANK GROUP

World Bank Open Data Website. Available at: https://data.worldbank.org/

The Data We Will Use



DataBank

WHAT'S NEW

Gender Statistics was updated on December 18, 2021

Health Nutrition and Population Statistics was updated on December 17, 2021

Population estimates and projections was updated on December 17, 2021

World Development Indicators was updated on December 16, 2021

Explore. Create. Share: Development Data

DataBank is an analysis and visualisation tool that contains collections of time series data on a variety of topics. You can create your own queries; generate tables, charts, and maps; and easily save, embed, and share them. Enjoy using DataBank and let us know what you think!

Explore databases

Type keywords to filter database names
Filter by:
Topic
Source
Sort by: Most Used | Alphabetical | Last Updated | View all databases
Database preview: ON OFF

World Development Indicators Public

World Development Indicators (WDI) is the primary World Bank collection of development indicators, compiled from officially recognized international sources. It presents the most current and accurate global See more +

External Debt and Financial Flows statistics, Heath statistics, Gender, Economy, Social Data

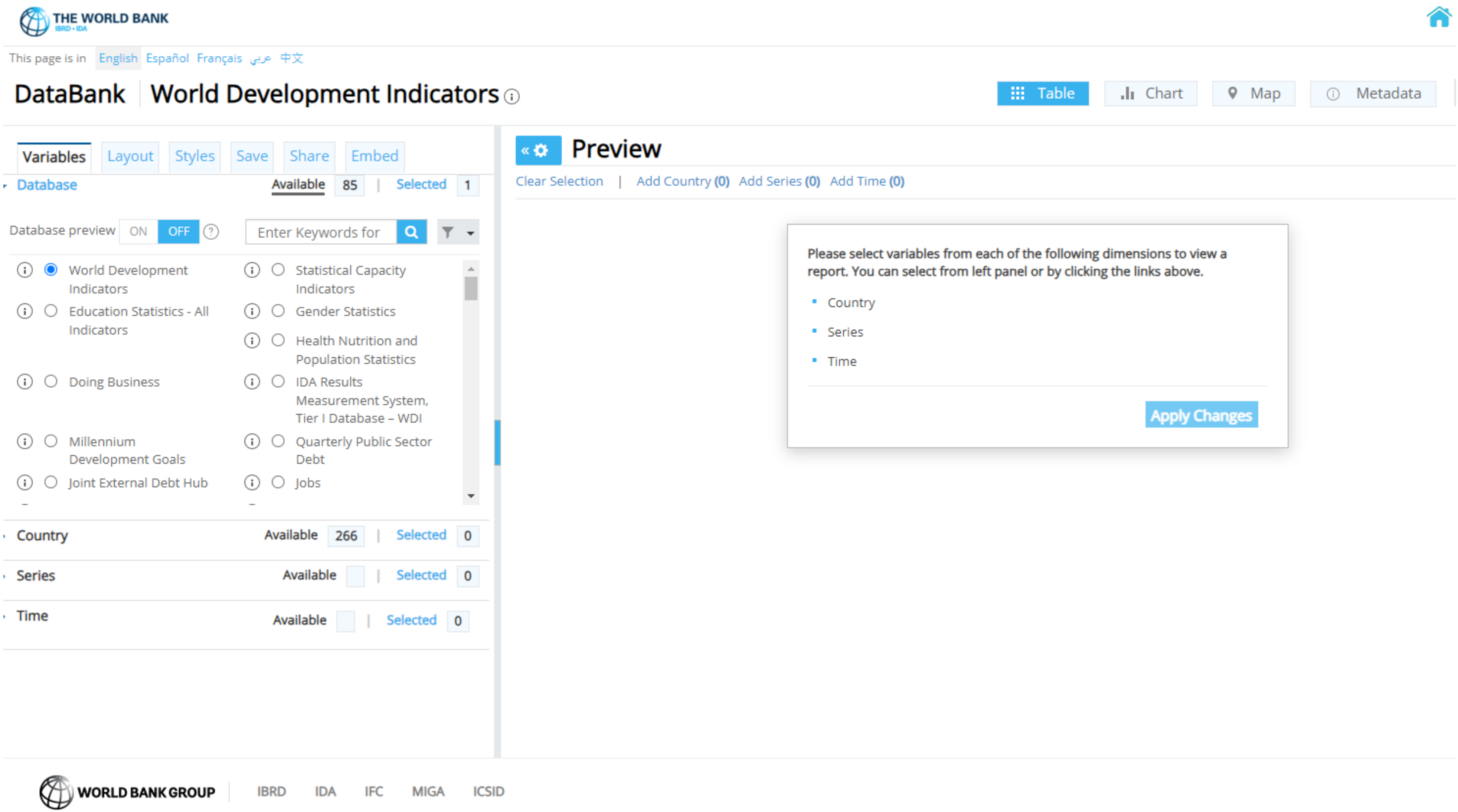
Last Updated: 12/16/2021

WHAT'S POPULAR

- GDP growth (annual %)
- GDP (current US$)
- GDP per capita (current US$)
- GNI per capita, Atlas method (current US$)
- Exports of goods and services (% of GDP)
- Foreign direct investment, net inflows (BoP, current US$)
- Inflation, consumer prices (annual %)

World Bank DataBank. Available at: https://databank.worldbank.org/home.aspx

The Data We Will Use



THE WORLD BANK
IBRD • IDA

This page is in English Español Français عربي 中文

DataBank | World Development Indicators ⓘ

Table | Chart | Map | ⓘ Metadata

Variables | Layout | Styles | Save | Share | Embed

	Available		Selected
Database	85		1

Database preview ON [✓ OFF] ?

Enter Keywords for

- ⓘ ● World Development Indicators
- ⓘ ○ Education Statistics - All Indicators
- ⓘ ○ Doing Business
- ⓘ ○ Millennium Development Goals
- ⓘ ○ Joint External Debt Hub
- ⓘ ○ Statistical Capacity Indicators
- ⓘ ○ Gender Statistics
- ⓘ ○ Health Nutrition and Population Statistics
- ⓘ ○ IDA Results Measurement System, Tier I Database – WDI
- ⓘ ○ Quarterly Public Sector Debt
- ⓘ ○ Jobs

	Available		Selected
Country	266		0
Series			0
Time			0

Preview

Clear Selection | Add Country (0) Add Series (0) Add Time (0)

Please select variables from each of the following dimensions to view a report. You can select from left panel or by clicking the links above.

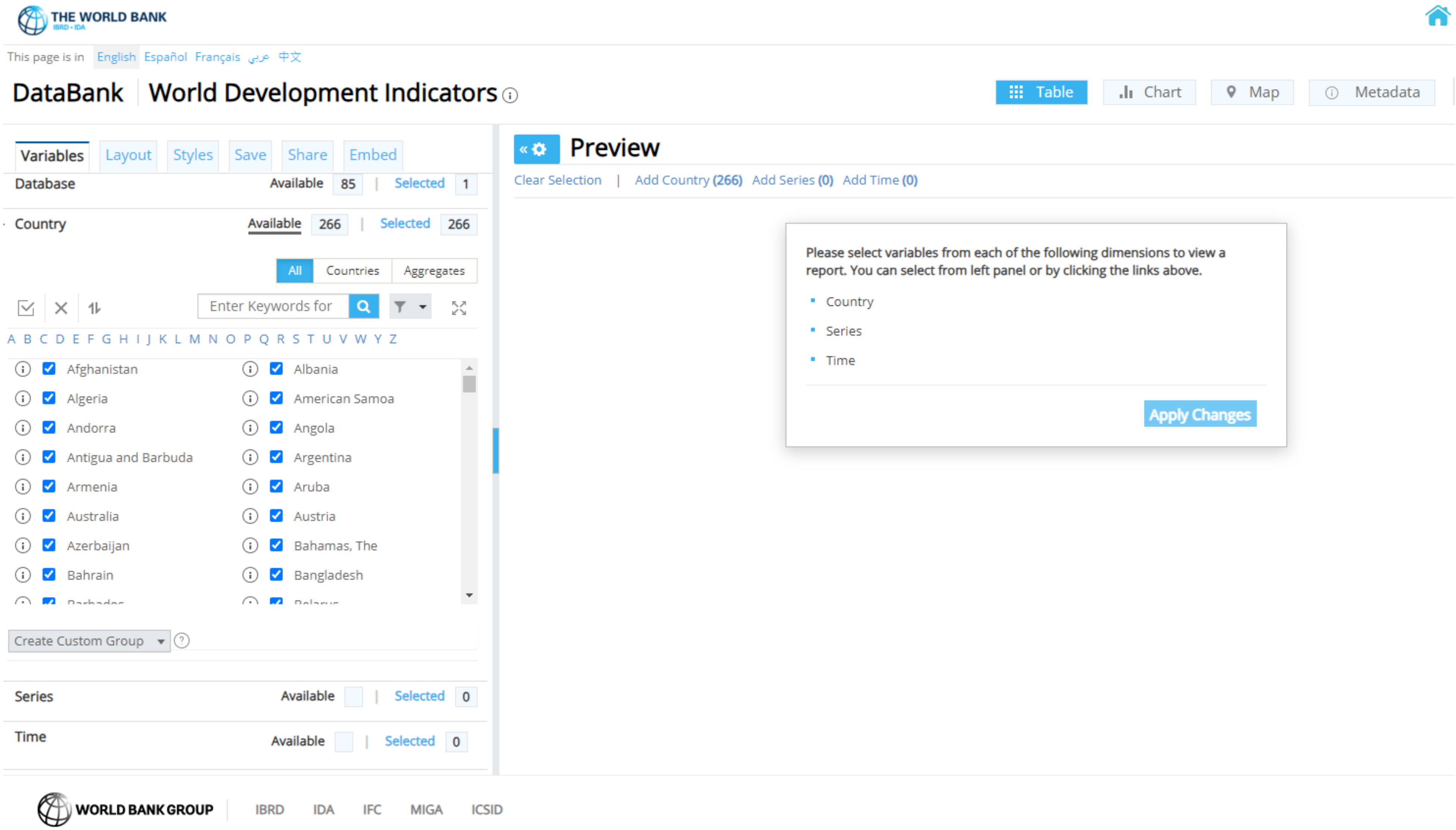
- Country
- Series
- Time

Apply Changes

WORLD BANK GROUP | IBRD IDA IFC MIGA ICSID

World Bank DataBank World Development Indicators. Available at: https://databank.worldbank.org/source/world-development-indicators

The Data We Will Use



THE WORLD BANK
IBRD • IDA

This page is in English Español Français عربي 中文

DataBank World Development Indicators ⓘ

Table | Chart | Map | ⓘ Metadata

Variables | Layout | Styles | Save | Share | Embed

Database — Available 85 | Selected 1

Country — Available 266 | Selected 266

All | Countries | Aggregates

Enter Keywords for

A B C D E F G H I J K L M N O P Q R S T U V W Y Z

ⓘ ☑ Afghanistan	ⓘ ☑ Albania
ⓘ ☑ Algeria	ⓘ ☑ American Samoa
ⓘ ☑ Andorra	ⓘ ☑ Angola
ⓘ ☑ Antigua and Barbuda	ⓘ ☑ Argentina
ⓘ ☑ Armenia	ⓘ ☑ Aruba
ⓘ ☑ Australia	ⓘ ☑ Austria
ⓘ ☑ Azerbaijan	ⓘ ☑ Bahamas, The
ⓘ ☑ Bahrain	ⓘ ☑ Bangladesh
ⓘ ☑ Barbados	ⓘ ☑ Belarus

Create Custom Group ⓘ

Series — Available | Selected 0

Time — Available | Selected 0

Preview

Clear Selection | Add Country (266) Add Series (0) Add Time (0)

Please select variables from each of the following dimensions to view a report. You can select from left panel or by clicking the links above.

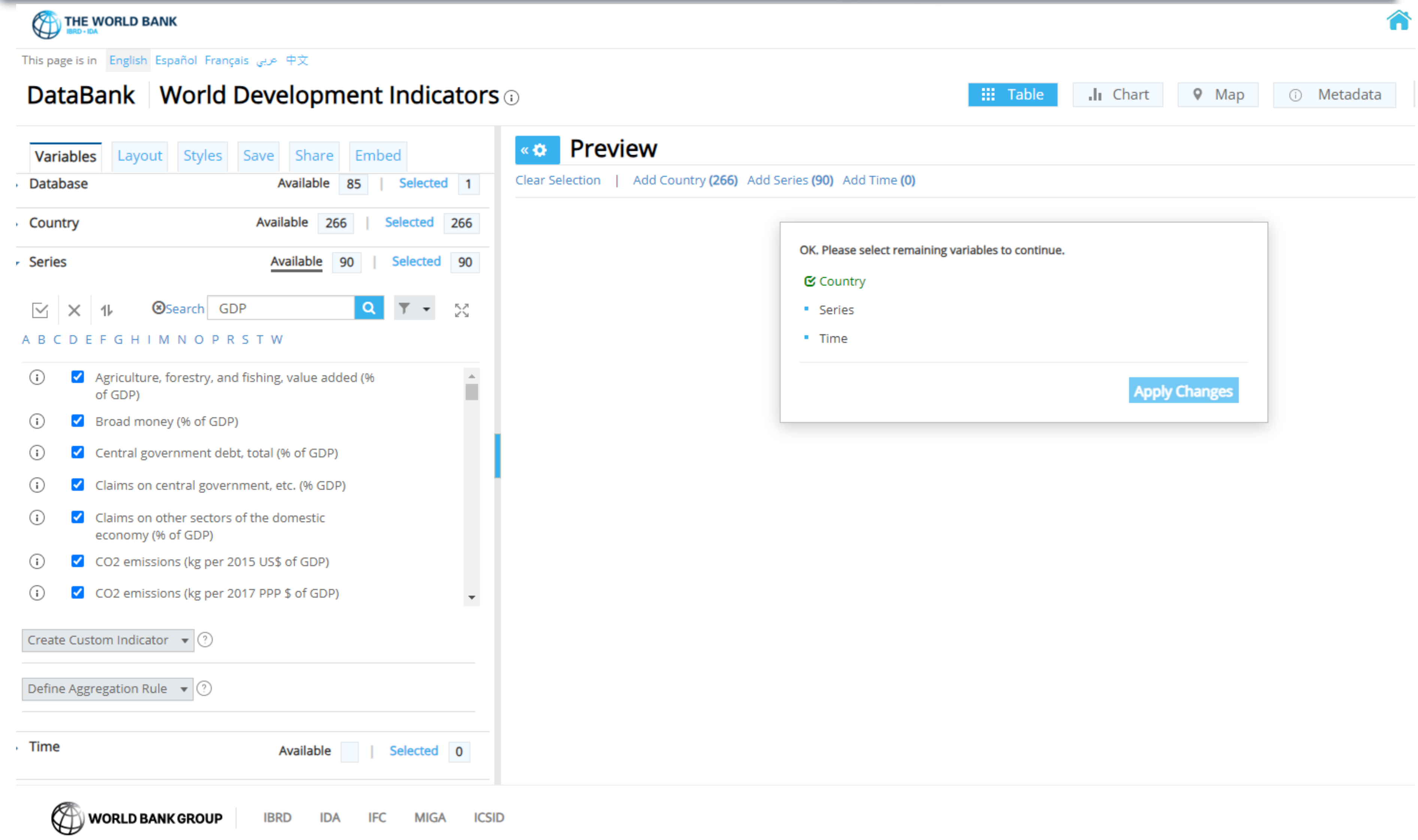
- Country
- Series
- Time

Apply Changes

WORLD BANK GROUP IBRD IDA IFC MIGA ICSID

World Bank DataBank World Development Indicators. Available at: https://databank.worldbank.org/source/world-development-indicators

The Data We Will Use



THE WORLD BANK IBRD • IDA

This page is in English Español Français عربي 中文

DataBank | World Development Indicators ⓘ

Table | Chart | Map | Metadata

Variables | Layout | Styles | Save | Share | Embed

	Available		Selected	
Database	Available	85	Selected	1
Country	Available	266	Selected	266
Series	Available	90	Selected	90

Search GDP

A B C D E F G H I M N O P R S T W

- ☑ Agriculture, forestry, and fishing, value added (% of GDP)
- ☑ Broad money (% of GDP)
- ☑ Central government debt, total (% of GDP)
- ☑ Claims on central government, etc. (% GDP)
- ☑ Claims on other sectors of the domestic economy (% of GDP)
- ☑ CO2 emissions (kg per 2015 US$ of GDP)
- ☑ CO2 emissions (kg per 2017 PPP $ of GDP)

Create Custom Indicator ▾ ?

Define Aggregation Rule ▾ ?

	Available		Selected	
Time	Available		Selected	0

Preview

Clear Selection | Add Country (266) Add Series (90) Add Time (0)

OK. Please select remaining variables to continue.

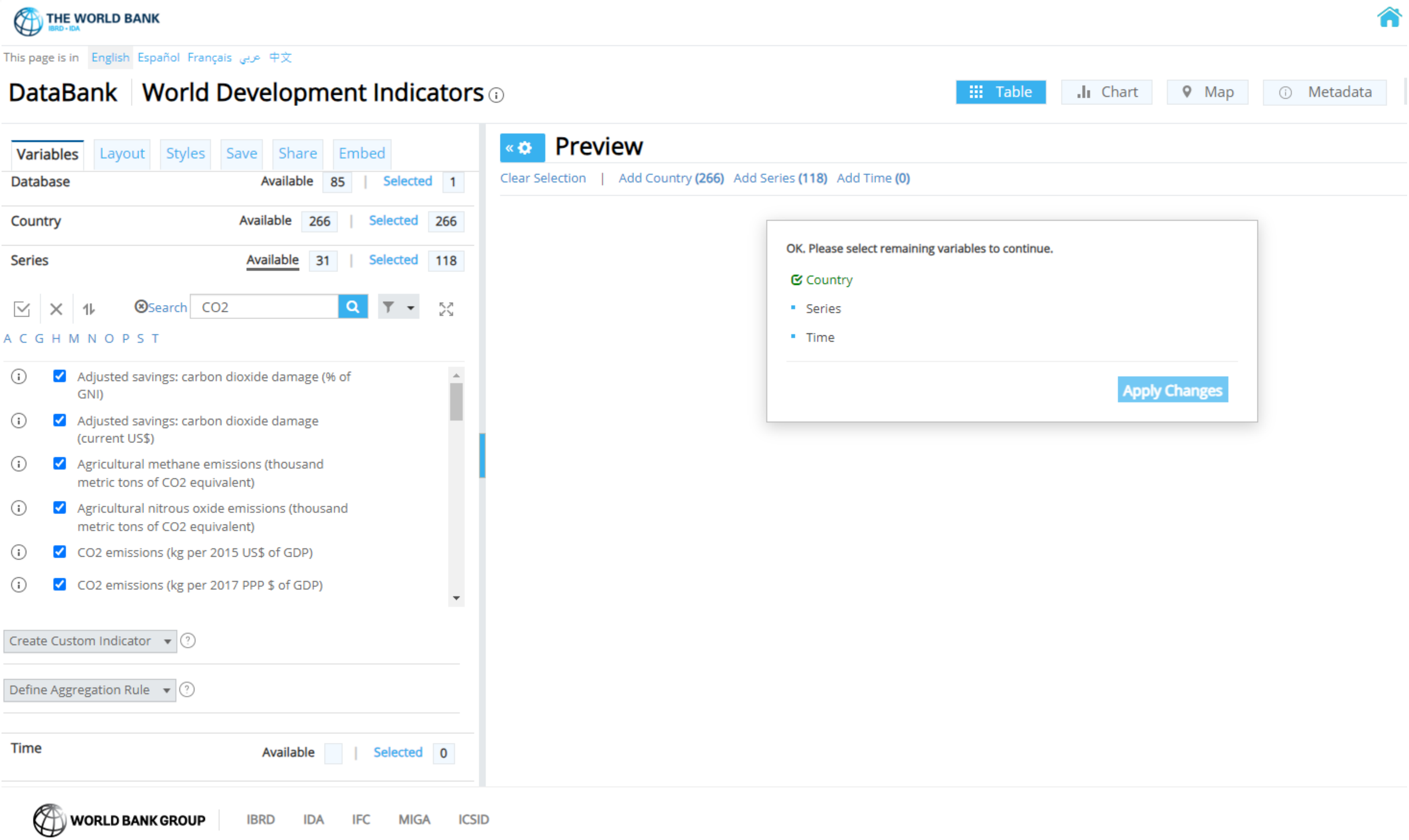
- ✓ Country
- Series
- Time

Apply Changes

WORLD BANK GROUP | IBRD | IDA | IFC | MIGA | ICSID

World Bank DataBank World Development Indicators. Available at: https://databank.worldbank.org/source/world-development-indicators

The Data We Will Use



THE WORLD BANK
IBRD · IDA

This page is in English Español Français عربي 中文

DataBank | World Development Indicators

Table | Chart | Map | Metadata

Variables | Layout | Styles | Save | Share | Embed

	Available		Selected	
Database	Available	85	Selected	1
Country	Available	266	Selected	266
Series	Available	31	Selected	118

Search: CO2

A C G H M N O P S T

- ☑ Adjusted savings: carbon dioxide damage (% of GNI)
- ☑ Adjusted savings: carbon dioxide damage (current US$)
- ☑ Agricultural methane emissions (thousand metric tons of CO2 equivalent)
- ☑ Agricultural nitrous oxide emissions (thousand metric tons of CO2 equivalent)
- ☑ CO2 emissions (kg per 2015 US$ of GDP)
- ☑ CO2 emissions (kg per 2017 PPP $ of GDP)

Create Custom Indicator

Define Aggregation Rule

Time Available | Selected 0

Preview

Clear Selection | Add Country (266) Add Series (118) Add Time (0)

OK. Please select remaining variables to continue.

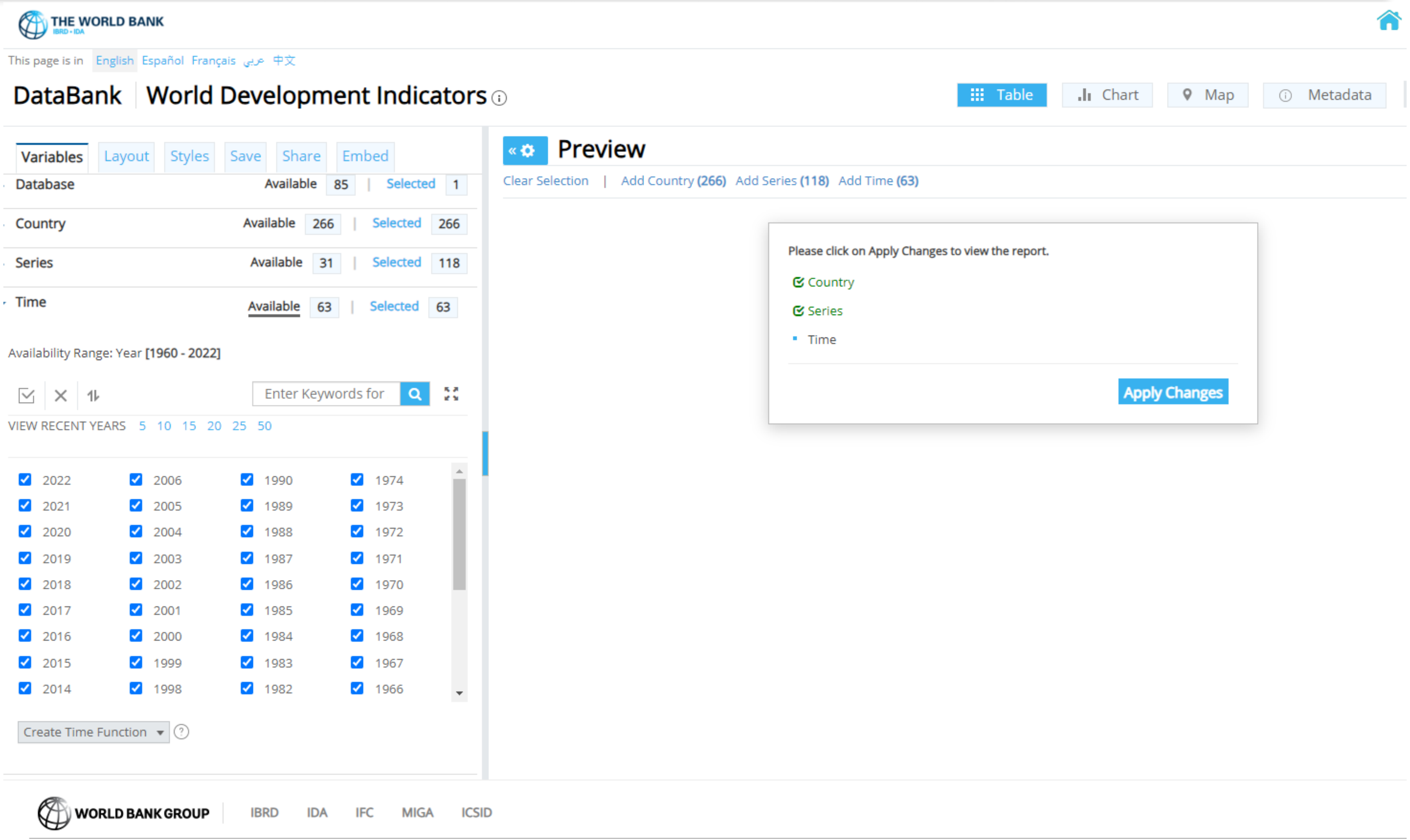
- ✓ Country
- Series
- Time

Apply Changes

WORLD BANK GROUP | IBRD | IDA | IFC | MIGA | ICSID

World Bank DataBank World Development Indicators. Available at: https://databank.worldbank.org/source/world-development-indicators

The Data We Will Use



THE WORLD BANK
IBRD • IDA

This page is in English Español Français عربي 中文

DataBank World Development Indicators

Table | Chart | Map | Metadata

Variables | Layout | Styles | Save | Share | Embed

Database	Available 85	Selected 1
Country	Available 266	Selected 266
Series	Available 31	Selected 118
Time	Available 63	Selected 63

Availability Range: Year [1960 - 2022]

Enter Keywords for

VIEW RECENT YEARS 5 10 15 20 25 50

☑ 2022
☑ 2021
☑ 2020
☑ 2019
☑ 2018
☑ 2017
☑ 2016
☑ 2015
☑ 2014
☑ 2006
☑ 2005
☑ 2004
☑ 2003
☑ 2002
☑ 2001
☑ 2000
☑ 1999
☑ 1998
☑ 1990
☑ 1989
☑ 1988
☑ 1987
☑ 1986
☑ 1985
☑ 1984
☑ 1983
☑ 1982
☑ 1974
☑ 1973
☑ 1972
☑ 1971
☑ 1970
☑ 1969
☑ 1968
☑ 1967
☑ 1966

Create Time Function

Preview

Clear Selection | Add Country (266) Add Series (118) Add Time (63)

Please click on Apply Changes to view the report.

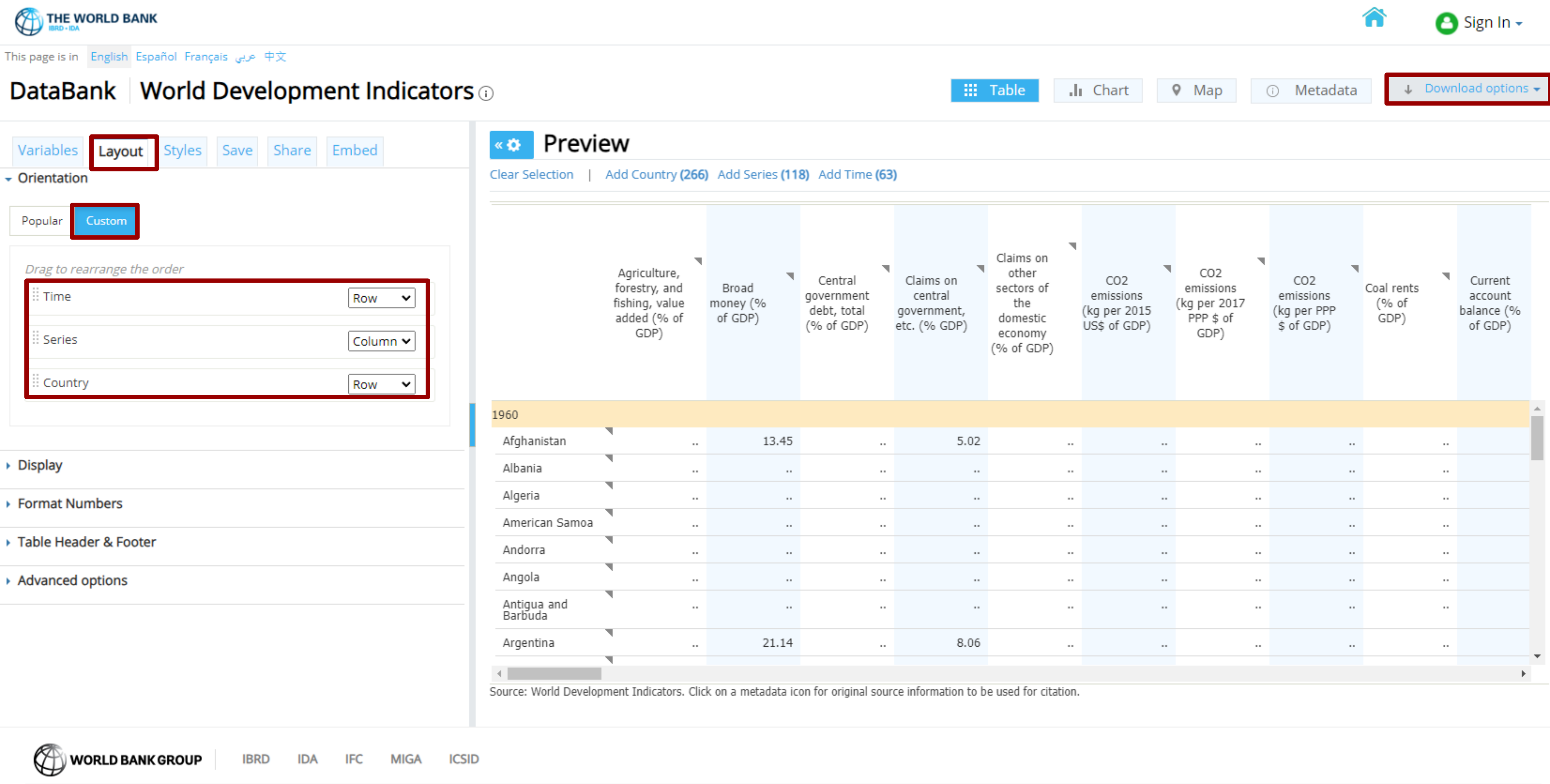
☑ Country
☑ Series
▪ Time

Apply Changes

WORLD BANK GROUP IBRD IDA IFC MIGA ICSID

World Bank DataBank World Development Indicators. Available at: https://databank.worldbank.org/source/world-development-indicators

The Data We Will Use



THE WORLD BANK IBRD • IDA

Sign In

This page is in English Español Français عربي 中文

DataBank World Development Indicators

Table | Chart | Map | Metadata | Download options

Variables | Layout | Styles | Save | Share | Embed

Orientation

Popular | Custom

Drag to rearrange the order

Time	Row
Series	Column
Country	Row

Display

Format Numbers

Table Header & Footer

Advanced options

Preview

Clear Selection | Add Country (266) Add Series (118) Add Time (63)

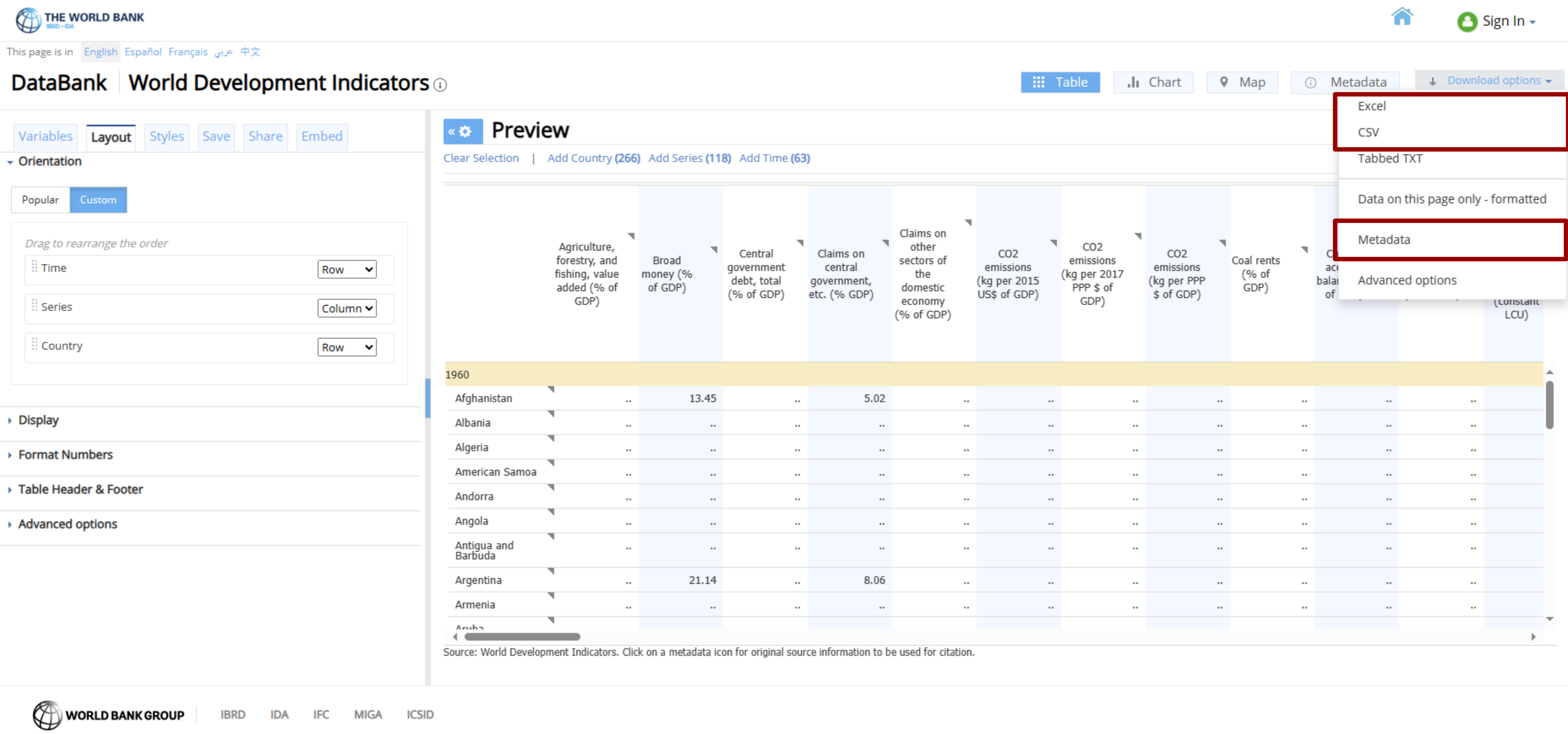
	Agriculture, forestry, and fishing, value added (% of GDP)	Broad money (% of GDP)	Central government debt, total (% of GDP)	Claims on central government, etc. (% GDP)	Claims on other sectors of the domestic economy (% of GDP)	CO2 emissions (kg per 2015 US$ of GDP)	CO2 emissions (kg per 2017 PPP $ of GDP)	CO2 emissions (kg per PPP $ of GDP)	Coal rents (% of GDP)	Current account balance (% of GDP)
1960										
Afghanistan	..	13.45	..	5.02	..	..	..	..	..	
Albania	..	..	..	..	..	..	..	..	..	
Algeria	..	..	..	..	..	..	..	..	..	
American Samoa	..	..	..	..	..	..	..	..	..	
Andorra	..	..	..	..	..	..	..	..	..	
Angola	..	..	..	..	..	..	..	..	..	
Antigua and Barbuda	..	..	..	..	..	..	..	..	..	
Argentina	..	21.14	..	8.06	..	..	..	..	..	

Source: World Development Indicators. Click on a metadata icon for original source information to be used for citation.

WORLD BANK GROUP IBRD IDA IFC MIGA ICSID

World Bank DataBank World Development Indicators. Available at: https://databank.worldbank.org/source/world-development-indicators

The Data We Will Use



THE WORLD BANK IBRD • IDA

Sign In

This page is in English Español Français عربي 中文

DataBank World Development Indicators

- Excel
- CSV
- Tabbed TXT
- Data on this page only - formatted
- Metadata
- Advanced options

Orientation

Drag to rearrange the order

Time	Row
Series	Column
Country	Row

Preview

Clear Selection | Add Country (266) Add Series (118) Add Time (63)

	Agriculture, forestry, and fishing, value added (% of GDP)	Broad money (% of GDP)	Central government debt, total (% of GDP)	Claims on central government, etc. (% GDP)	Claims on other sectors of the domestic economy (% of GDP)	CO2 emissions (kg per 2015 US$ of GDP)	CO2 emissions (kg per 2017 PPP $ of GDP)	CO2 emissions (kg per PPP $ of GDP)	Coal rents (% of GDP)	C acc balan of		(constant LCU)
1960												
Afghanistan	..	13.45	..	5.02	..	..	..	..	..	..	..	
Albania	..	..	..	..	..	..	..	..	..	..	..	
Algeria	..	..	..	..	..	..	..	..	..	..	..	
American Samoa	..	..	..	..	..	..	..	..	..	..	..	
Andorra	..	..	..	..	..	..	..	..	..	..	..	
Angola	..	..	..	..	..	..	..	..	..	..	..	
Antigua and Barbuda	..	..	..	..	..	..	..	..	..	..	..	
Argentina	..	21.14	..	8.06	..	..	..	..	..	..	..	
Armenia	..	..	..	..	..	..	..	..	..	..	..	
Aruba												

Source: World Development Indicators. Click on a metadata icon for original source information to be used for citation.

WORLD BANK GROUP IBRD IDA IFC MIGA ICSID

World Bank DataBank World Development Indicators. Available at: https://databank.worldbank.org/source/world-development-indicators

The Data We Will Use

- World_Bank_Data_1.csv
  - **GDP-related variables (Search “GDP”)**
  - **CO_2 Emission-related variables (Search “CO2”)**
  - All the countries (266)
  - 1960-2022 (63 years)
  - 118 series (columns)
  - **World Development Indicators** Database

- World_Bank_Data_2.csv
  - **GDP-related variables (Search “GDP”)**
  - All the countries (**272**)
  - 1960-**2100** (**101** years)
  - 73 series (columns)
  - **Education Statistics – All Indicators** Database

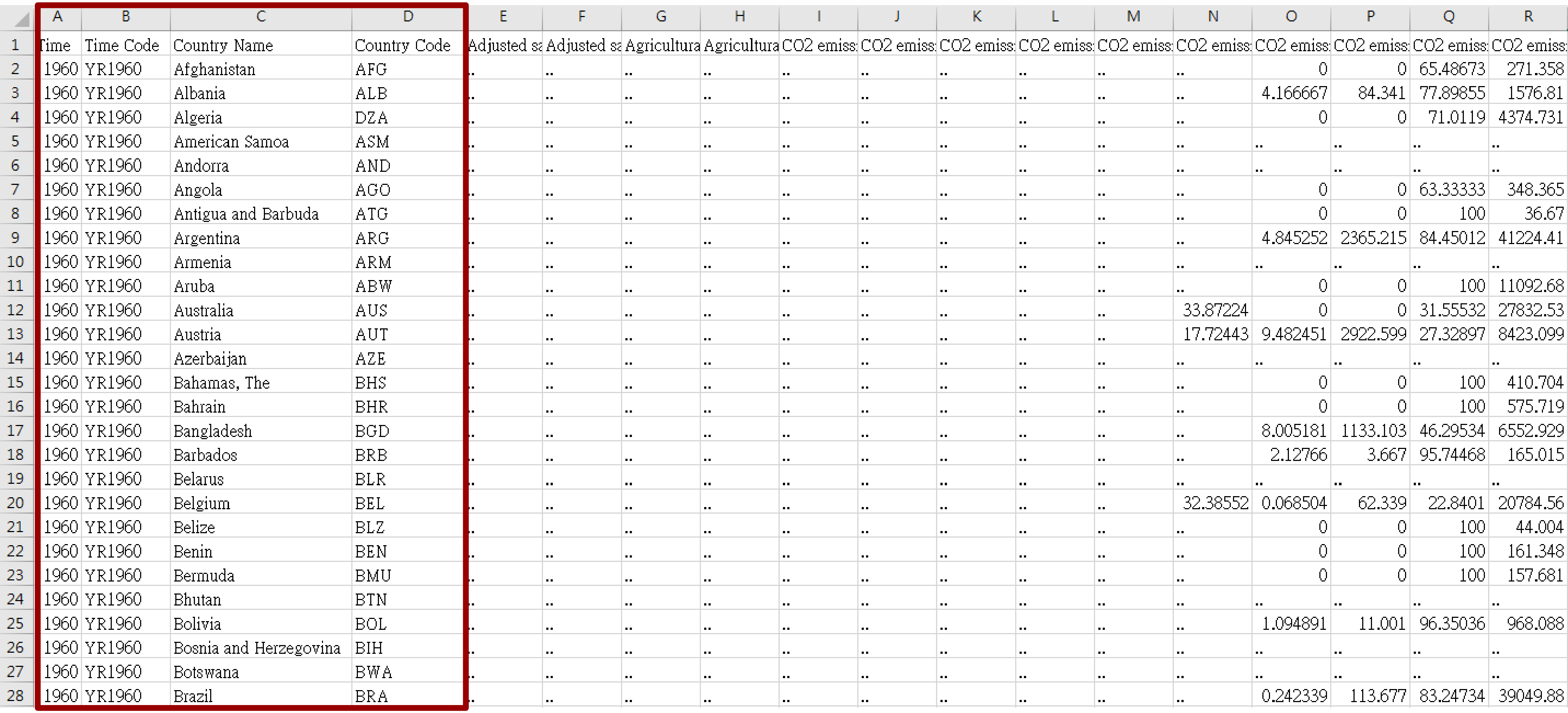
World Bank DataBank World Development Indicators. Available at: https://databank.worldbank.org/source/world-development-indicators

The Data We Will Use

- World_Bank_Data_3.csv
  - **Population-related variables (Search “Gender”)**
  - All the countries (266)
  - 1960-2022 (63 years)
  - **Health Nutrition and Population Statistics** Database
  - 149 series (columns)

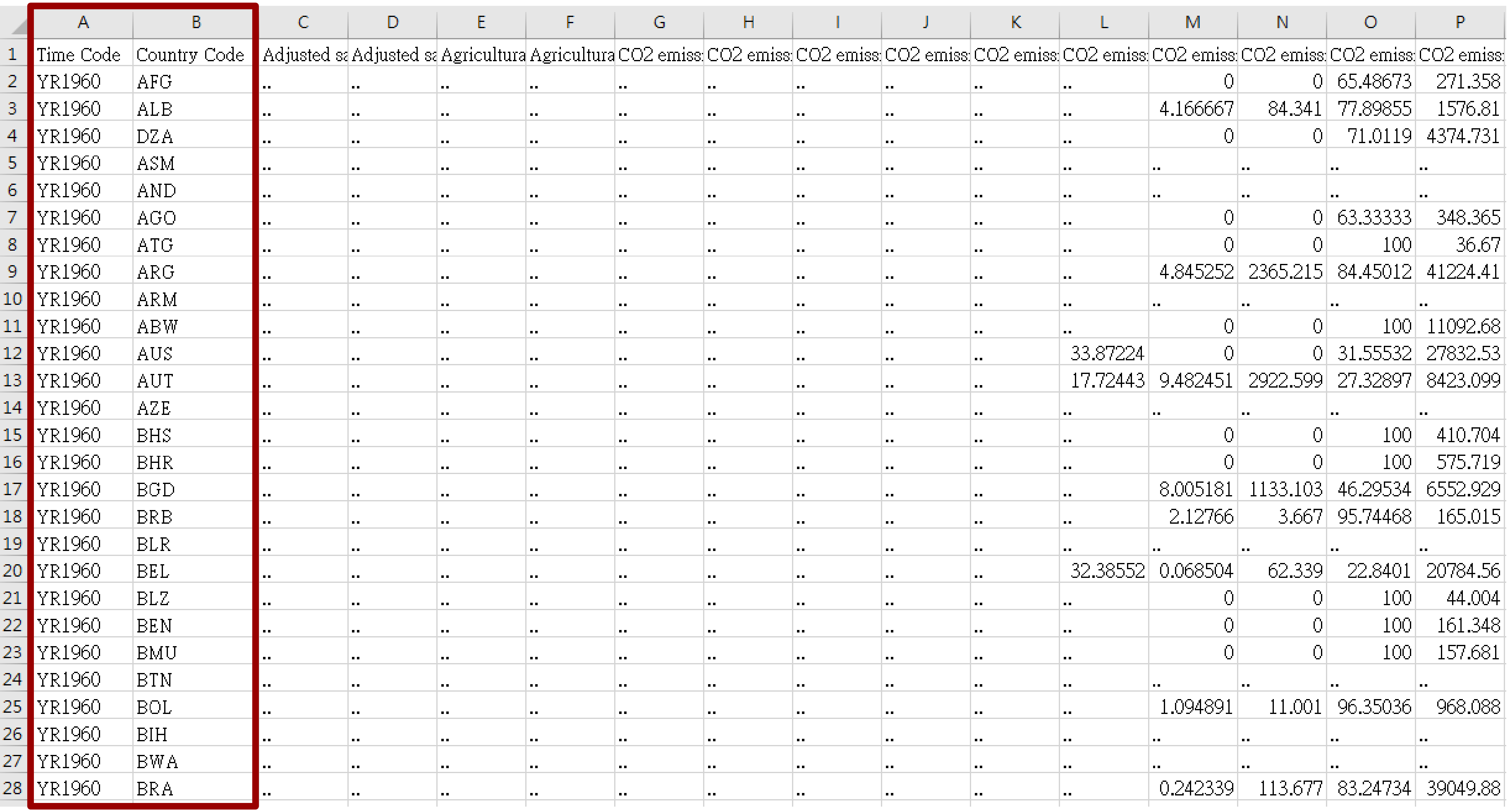
World Bank DataBank World Development Indicators. Available at: https://databank.worldbank.org/source/world-development-indicators

Original World_Bank_Data_1



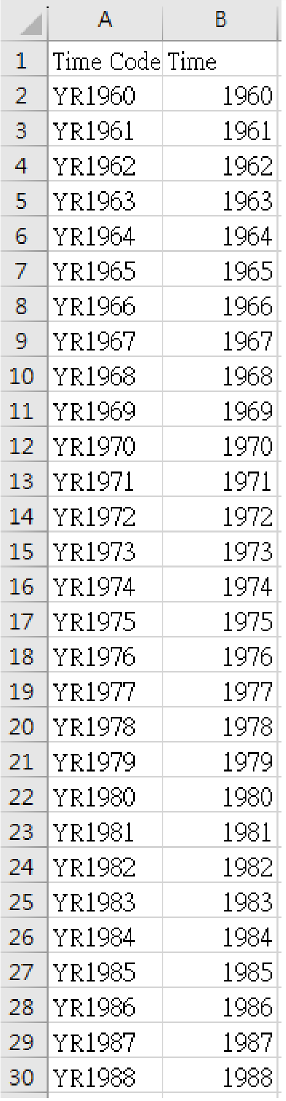
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Time	Time Code	Country Name	Country Code	Adjusted sa	Adjusted sa	Agricultura	Agricultura	CO2 emiss:	CO2 emiss:	CO2 emiss:	CO2 emiss:	CO2 emiss:	CO2 emiss:	CO2 emiss:	CO2 emiss:	CO2 emiss:	CO2 emiss:
2	1960	YR1960	Afghanistan	AFG	..	..	..	..	..	..	..	..	..	..	0	0	65.48673	271.358
3	1960	YR1960	Albania	ALB	..	..	..	..	..	..	..	..	..	..	4.166667	84.341	77.89855	1576.81
4	1960	YR1960	Algeria	DZA	..	..	..	..	..	..	..	..	..	..	0	0	71.0119	4374.731
5	1960	YR1960	American Samoa	ASM	..	..	..	..	..	..	..	..	..	..	..	..	..	..
6	1960	YR1960	Andorra	AND	..	..	..	..	..	..	..	..	..	..	..	..	..	..
7	1960	YR1960	Angola	AGO	..	..	..	..	..	..	..	..	..	..	0	0	63.33333	348.365
8	1960	YR1960	Antigua and Barbuda	ATG	..	..	..	..	..	..	..	..	..	..	0	0	100	36.67
9	1960	YR1960	Argentina	ARG	..	..	..	..	..	..	..	..	..	..	4.845252	2365.215	84.45012	41224.41
10	1960	YR1960	Armenia	ARM	..	..	..	..	..	..	..	..	..	..	..	..	..	..
11	1960	YR1960	Aruba	ABW	..	..	..	..	..	..	..	..	..	..	0	0	100	11092.68
12	1960	YR1960	Australia	AUS	..	..	..	..	..	..	..	..	..	33.87224	0	0	31.55532	27832.53
13	1960	YR1960	Austria	AUT	..	..	..	..	..	..	..	..	..	17.72443	9.482451	2922.599	27.32897	8423.099
14	1960	YR1960	Azerbaijan	AZE	..	..	..	..	..	..	..	..	..	..	..	..	..	..
15	1960	YR1960	Bahamas, The	BHS	..	..	..	..	..	..	..	..	..	..	0	0	100	410.704
16	1960	YR1960	Bahrain	BHR	..	..	..	..	..	..	..	..	..	..	0	0	100	575.719
17	1960	YR1960	Bangladesh	BGD	..	..	..	..	..	..	..	..	..	..	8.005181	1133.103	46.29534	6552.929
18	1960	YR1960	Barbados	BRB	..	..	..	..	..	..	..	..	..	..	2.12766	3.667	95.74468	165.015
19	1960	YR1960	Belarus	BLR	..	..	..	..	..	..	..	..	..	..	..	..	..	..
20	1960	YR1960	Belgium	BEL	..	..	..	..	..	..	..	..	..	32.38552	0.068504	62.339	22.8401	20784.56
21	1960	YR1960	Belize	BLZ	..	..	..	..	..	..	..	..	..	..	0	0	100	44.004
22	1960	YR1960	Benin	BEN	..	..	..	..	..	..	..	..	..	..	0	0	100	161.348
23	1960	YR1960	Bermuda	BMU	..	..	..	..	..	..	..	..	..	..	0	0	100	157.681
24	1960	YR1960	Bhutan	BTN	..	..	..	..	..	..	..	..	..	..	..	..	..	..
25	1960	YR1960	Bolivia	BOL	..	..	..	..	..	..	..	..	..	..	1.094891	11.001	96.35036	968.088
26	1960	YR1960	Bosnia and Herzegovina	BIH	..	..	..	..	..	..	..	..	..	..	..	..	..	..
27	1960	YR1960	Botswana	BWA	..	..	..	..	..	..	..	..	..	..	..	..	..	..
28	1960	YR1960	Brazil	BRA	..	..	..	..	..	..	..	..	..	..	0.242339	113.677	83.24734	39049.88

World_Bank_Data_1

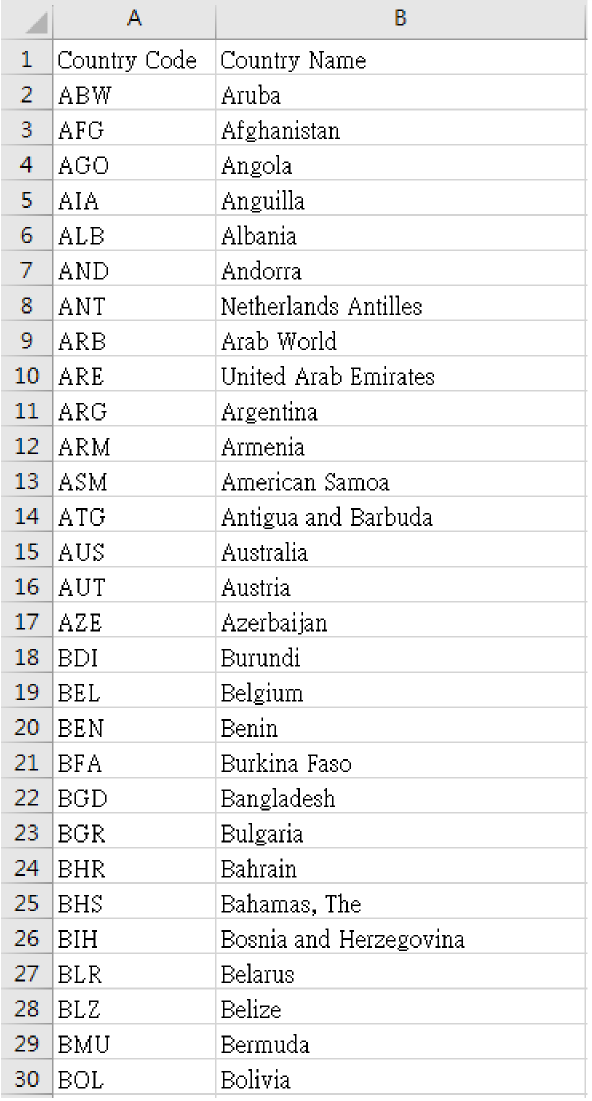


	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Time Code	Country Code	Adjusted sa	Adjusted sa	Agricultura	Agricultura	CO2 emiss	CO2 emiss	CO2 emiss	CO2 emiss	CO2 emiss	CO2 emiss	CO2 emiss	CO2 emiss	CO2 emiss	CO2 emiss
2	YR1960	AFG	..	..	..	..	..	..	..	..	..	..	0	0	65.48673	271.358
3	YR1960	ALB	..	..	..	..	..	..	..	..	..	..	4.166667	84.341	77.89855	1576.81
4	YR1960	DZA	..	..	..	..	..	..	..	..	..	..	0	0	71.0119	4374.731
5	YR1960	ASM	..	..	..	..	..	..	..	..	..	..	..	..	..	..
6	YR1960	AND	..	..	..	..	..	..	..	..	..	..	..	..	..	..
7	YR1960	AGO	..	..	..	..	..	..	..	..	..	..	0	0	63.33333	348.365
8	YR1960	ATG	..	..	..	..	..	..	..	..	..	..	0	0	100	36.67
9	YR1960	ARG	..	..	..	..	..	..	..	..	..	..	4.845252	2365.215	84.45012	41224.41
10	YR1960	ARM	..	..	..	..	..	..	..	..	..	..	..	..	..	..
11	YR1960	ABW	..	..	..	..	..	..	..	..	..	..	0	0	100	11092.68
12	YR1960	AUS	..	..	..	..	..	..	..	..	..	33.87224	0	0	31.55532	27832.53
13	YR1960	AUT	..	..	..	..	..	..	..	..	..	17.72443	9.482451	2922.599	27.32897	8423.099
14	YR1960	AZE	..	..	..	..	..	..	..	..	..	..	..	..	..	..
15	YR1960	BHS	..	..	..	..	..	..	..	..	..	..	0	0	100	410.704
16	YR1960	BHR	..	..	..	..	..	..	..	..	..	..	0	0	100	575.719
17	YR1960	BGD	..	..	..	..	..	..	..	..	..	..	8.005181	1133.103	46.29534	6552.929
18	YR1960	BRB	..	..	..	..	..	..	..	..	..	..	2.12766	3.667	95.74468	165.015
19	YR1960	BLR	..	..	..	..	..	..	..	..	..	..	..	..	..	..
20	YR1960	BEL	..	..	..	..	..	..	..	..	..	32.38552	0.068504	62.339	22.8401	20784.56
21	YR1960	BLZ	..	..	..	..	..	..	..	..	..	..	0	0	100	44.004
22	YR1960	BEN	..	..	..	..	..	..	..	..	..	..	0	0	100	161.348
23	YR1960	BMU	..	..	..	..	..	..	..	..	..	..	0	0	100	157.681
24	YR1960	BTN	..	..	..	..	..	..	..	..	..	..	..	..	..	..
25	YR1960	BOL	..	..	..	..	..	..	..	..	..	..	1.094891	11.001	96.35036	968.088
26	YR1960	BIH	..	..	..	..	..	..	..	..	..	..	..	..	..	..
27	YR1960	BWA	..	..	..	..	..	..	..	..	..	..	..	..	..	..
28	YR1960	BRA	..	..	..	..	..	..	..	..	..	..	0.242339	113.677	83.24734	39049.88

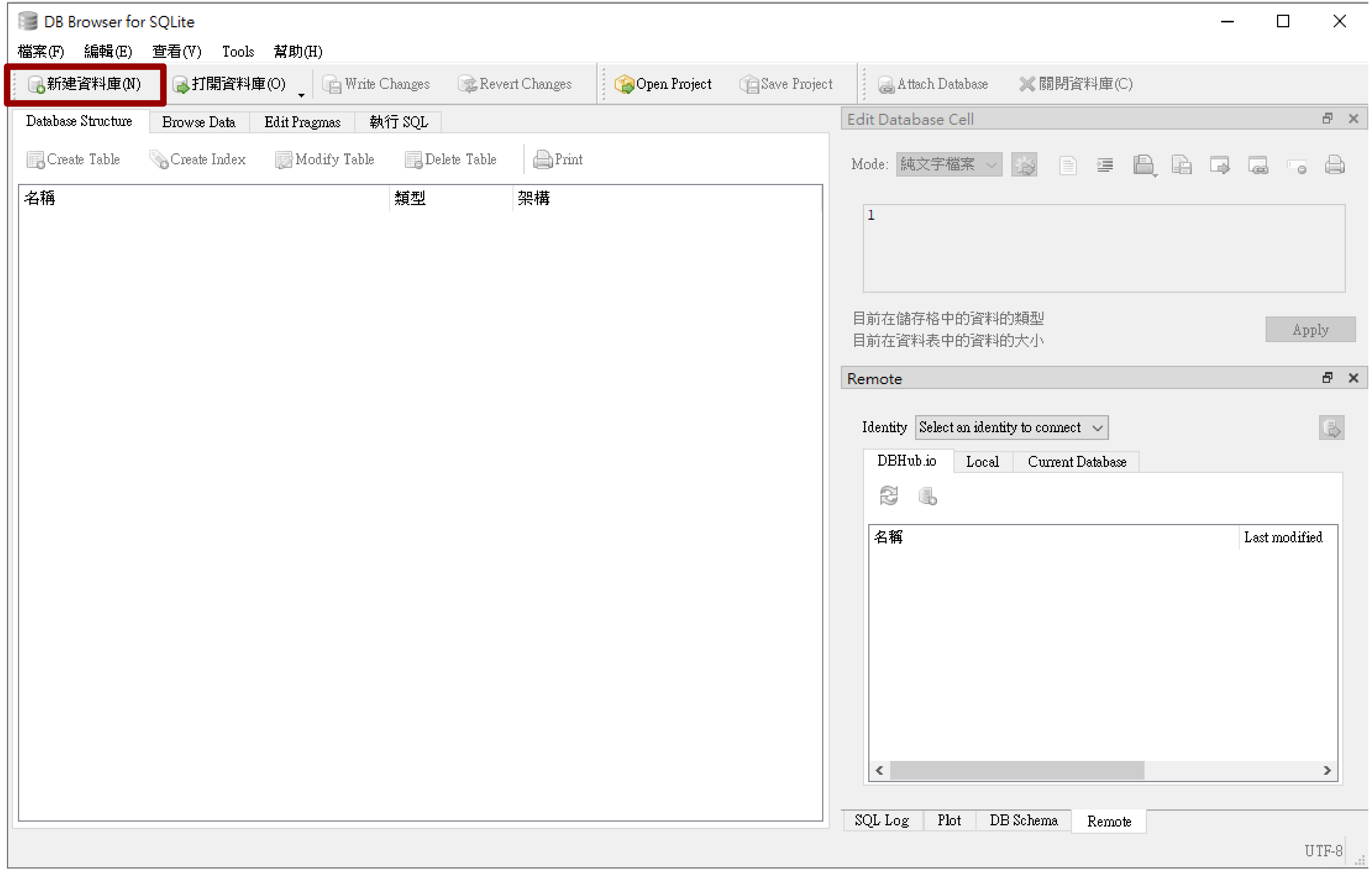
Year & Countries_1



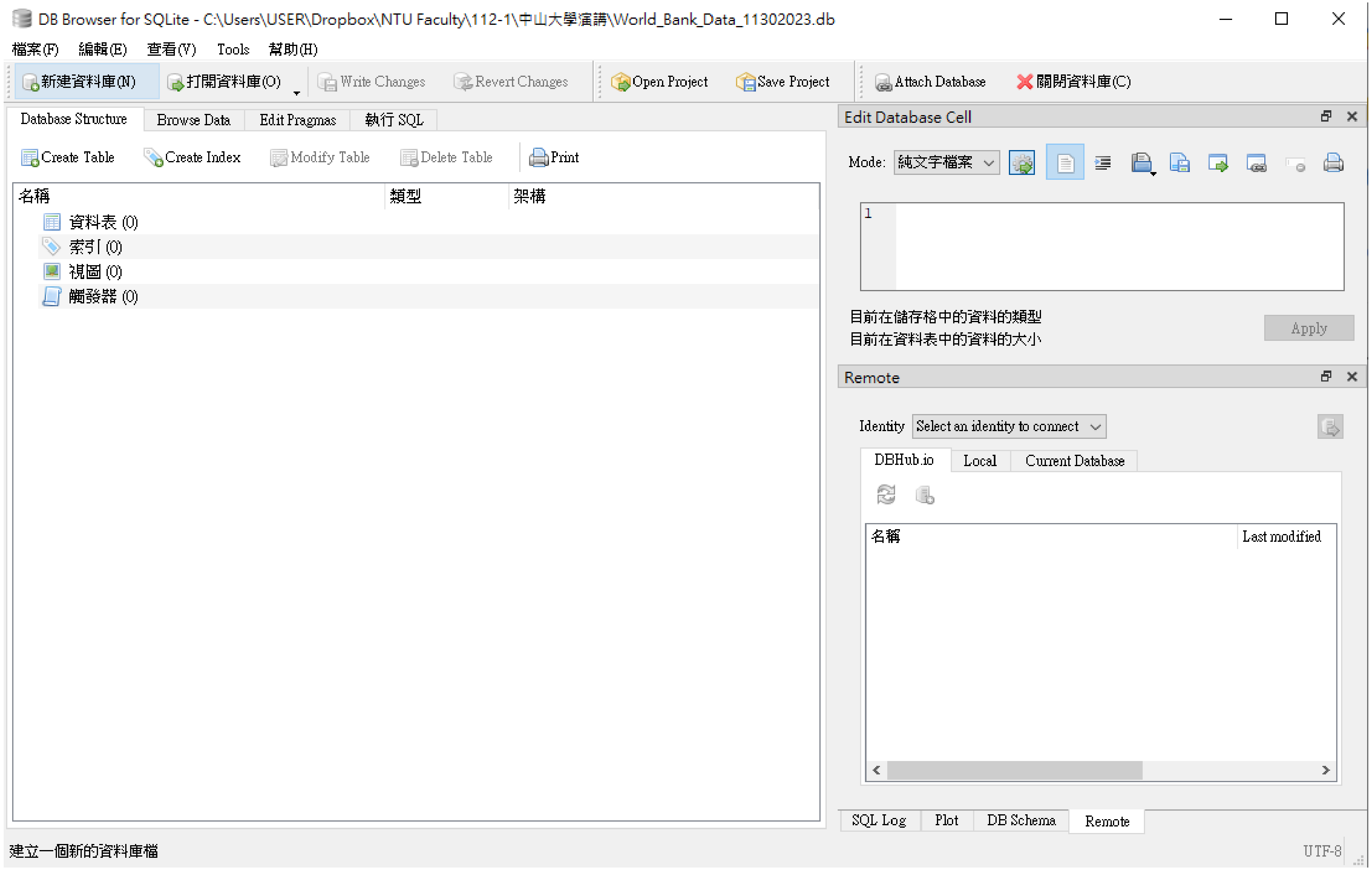
	A	B
1	Time Code	Time
2	YR1960	1960
3	YR1961	1961
4	YR1962	1962
5	YR1963	1963
6	YR1964	1964
7	YR1965	1965
8	YR1966	1966
9	YR1967	1967
10	YR1968	1968
11	YR1969	1969
12	YR1970	1970
13	YR1971	1971
14	YR1972	1972
15	YR1973	1973
16	YR1974	1974
17	YR1975	1975
18	YR1976	1976
19	YR1977	1977
20	YR1978	1978
21	YR1979	1979
22	YR1980	1980
23	YR1981	1981
24	YR1982	1982
25	YR1983	1983
26	YR1984	1984
27	YR1985	1985
28	YR1986	1986
29	YR1987	1987
30	YR1988	1988



	A	B
1	Country Code	Country Name
2	ABW	Aruba
3	AFG	Afghanistan
4	AGO	Angola
5	AIA	Anguilla
6	ALB	Albania
7	AND	Andorra
8	ANT	Netherlands Antilles
9	ARB	Arab World
10	ARE	United Arab Emirates
11	ARG	Argentina
12	ARM	Armenia
13	ASM	American Samoa
14	ATG	Antigua and Barbuda
15	AUS	Australia
16	AUT	Austria
17	AZE	Azerbaijan
18	BDI	Burundi
19	BEL	Belgium
20	BEN	Benin
21	BFA	Burkina Faso
22	BGD	Bangladesh
23	BGR	Bulgaria
24	BHR	Bahrain
25	BHS	Bahamas, The
26	BIH	Bosnia and Herzegovina
27	BLR	Belarus
28	BLZ	Belize
29	BMU	Bermuda
30	BOL	Bolivia



目前在儲存格中的資料的類型
目前在資料表中的資料的大小



DB Browser for SQLite - C:\Users\USER\Dropbox\NTU Faculty\112-1\中山大學演講\World_Bank_Data_11302023.db

名稱	類型	架構
資料表 (0)		
索引 (0)		
視圖 (0)		
觸發器 (0)		

Mode: 純文字檔案

1

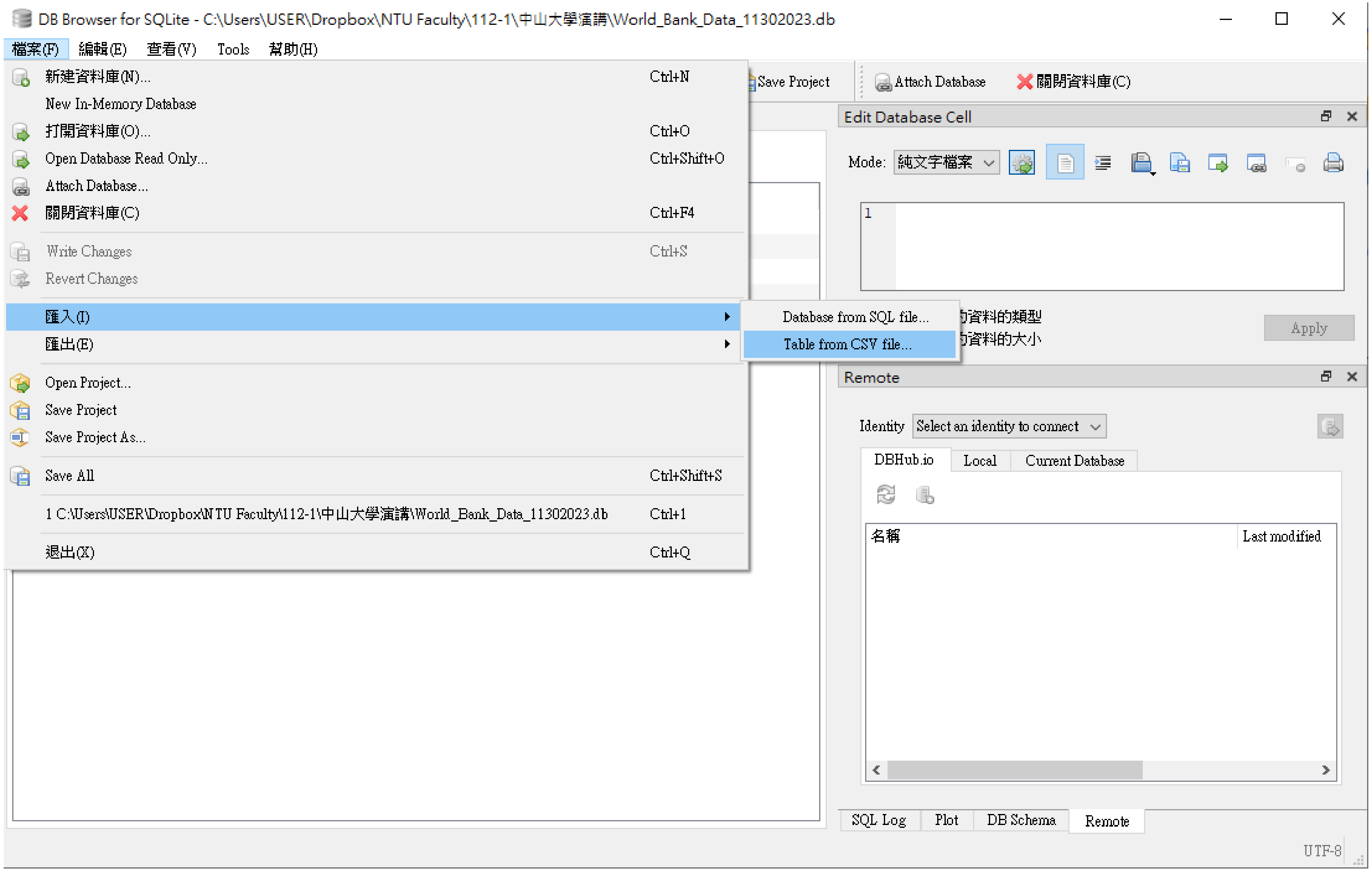
目前在儲存格中的資料的類型
目前在資料表中的資料的大小

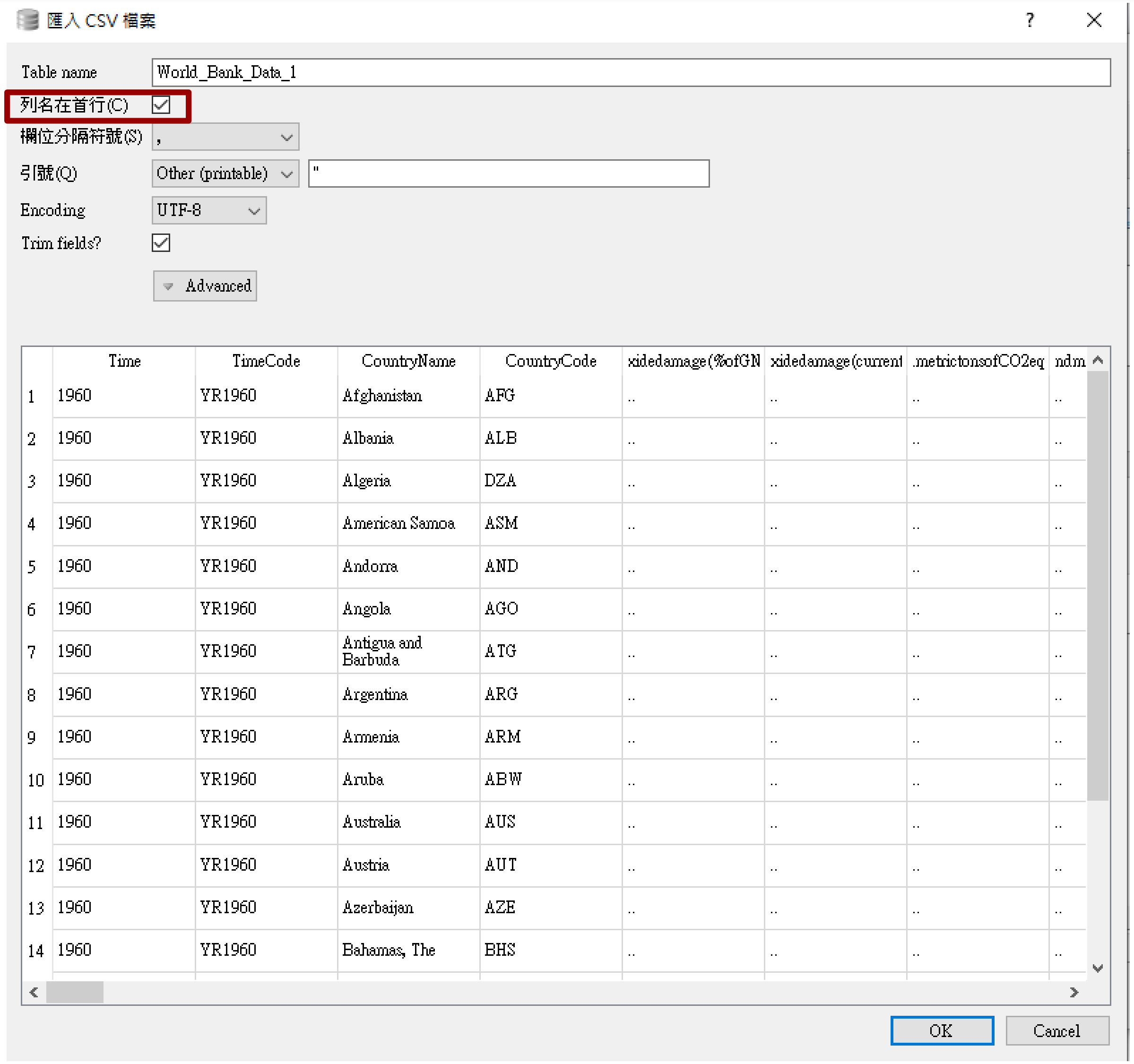
Identity: Select an identity to connect

名稱	Last modified

建立一個新的資料庫檔

UTF-8


DB Browser for SQLite - C:\Users\USER\Dropbox\NTU Faculty\112-1\中山大學演講\World_Bank_Data_11302023.db
檔案(F)
編輯(E)
查看(V)
Tools
幫助(H)
新建資料庫(N)... Ctrl+N
New In-Memory Database
打開資料庫(O)... Ctrl+O
Open Database Read Only... Ctrl+Shift+O
Attach Database...
關閉資料庫(C) Ctrl+F4
Write Changes Ctrl+S
Revert Changes
匯入(I)
匯出(E)
Open Project...
Save Project
Save Project As...
Save All Ctrl+Shift+S
1 C:\Users\USER\Dropbox\NTU Faculty\112-1\中山大學演講\World_Bank_Data_11302023.db Ctrl+1
退出(X) Ctrl+Q
Database from SQL file...
Table from CSV file...
Save Project
Attach Database
關閉資料庫(C)
Edit Database Cell
Mode: 純文字檔案
1
的資料的類型
的資料的大小
Apply
Remote
Identity Select an identity to connect
DBHub.io
Local
Current Database
名稱
Last modified
SQL Log
Plot
DB Schema
Remote
UTF-8



匯入 CSV 檔案

Table name: World_Bank_Data_1

列名在首行(C) ☑

欄位分隔符號(S): ,

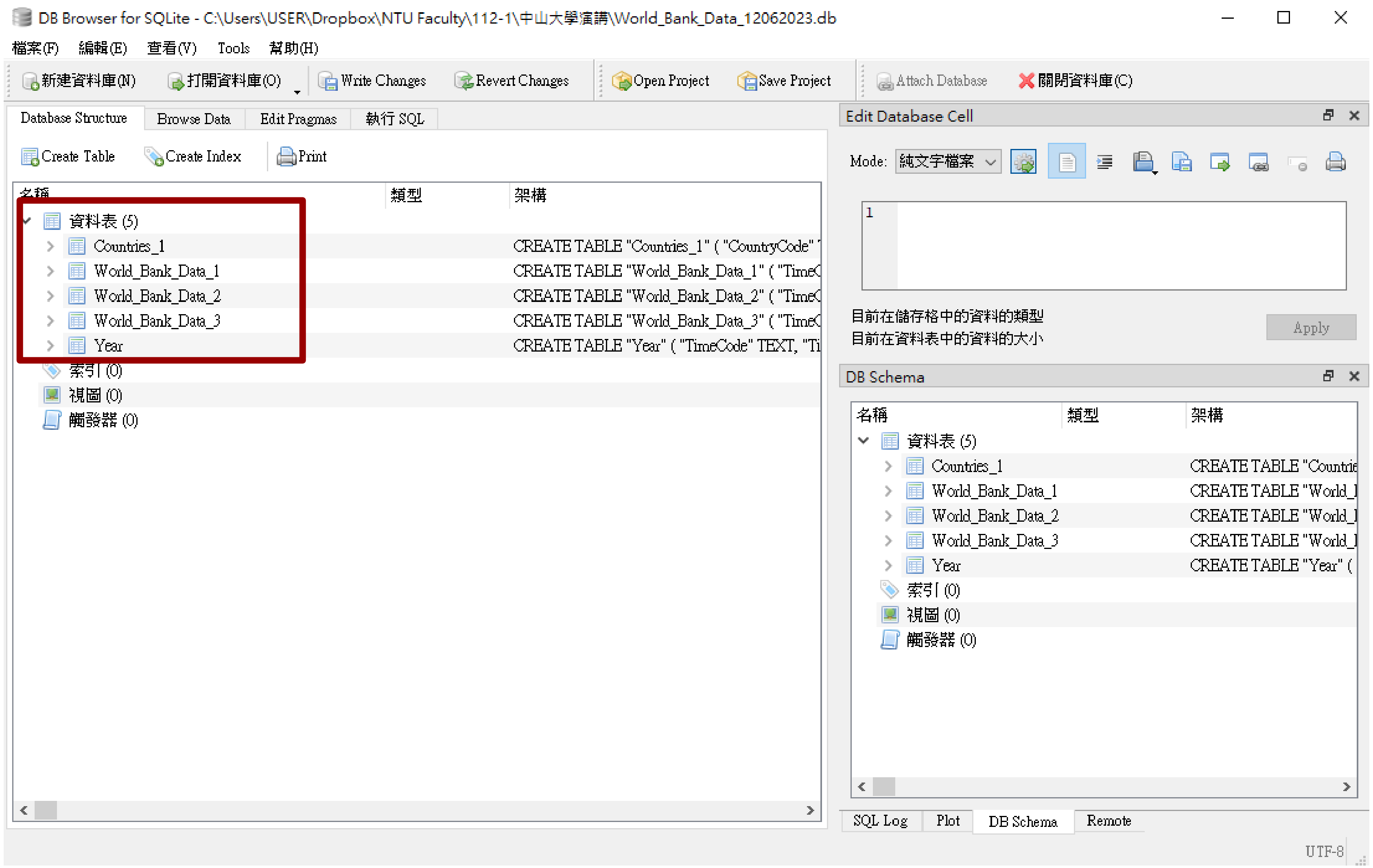
引號(Q): Other (printable) "

Encoding: UTF-8

Trim fields? ☑

Advanced

	Time	TimeCode	CountryName	CountryCode	xidedamage(%ofGN	xidedamage(current	.metrictonsofCO2eq	ndm
1	1960	YR1960	Afghanistan	AFG	..	..	..	..
2	1960	YR1960	Albania	ALB	..	..	..	..
3	1960	YR1960	Algeria	DZA	..	..	..	..
4	1960	YR1960	American Samoa	ASM	..	..	..	..
5	1960	YR1960	Andorra	AND	..	..	..	..
6	1960	YR1960	Angola	AGO	..	..	..	..
7	1960	YR1960	Antigua and Barbuda	ATG	..	..	..	..
8	1960	YR1960	Argentina	ARG	..	..	..	..
9	1960	YR1960	Armenia	ARM	..	..	..	..
10	1960	YR1960	Aruba	ABW	..	..	..	..
11	1960	YR1960	Australia	AUS	..	..	..	..
12	1960	YR1960	Austria	AUT	..	..	..	..
13	1960	YR1960	Azerbaijan	AZE	..	..	..	..
14	1960	YR1960	Bahamas, The	BHS	..	..	..	..

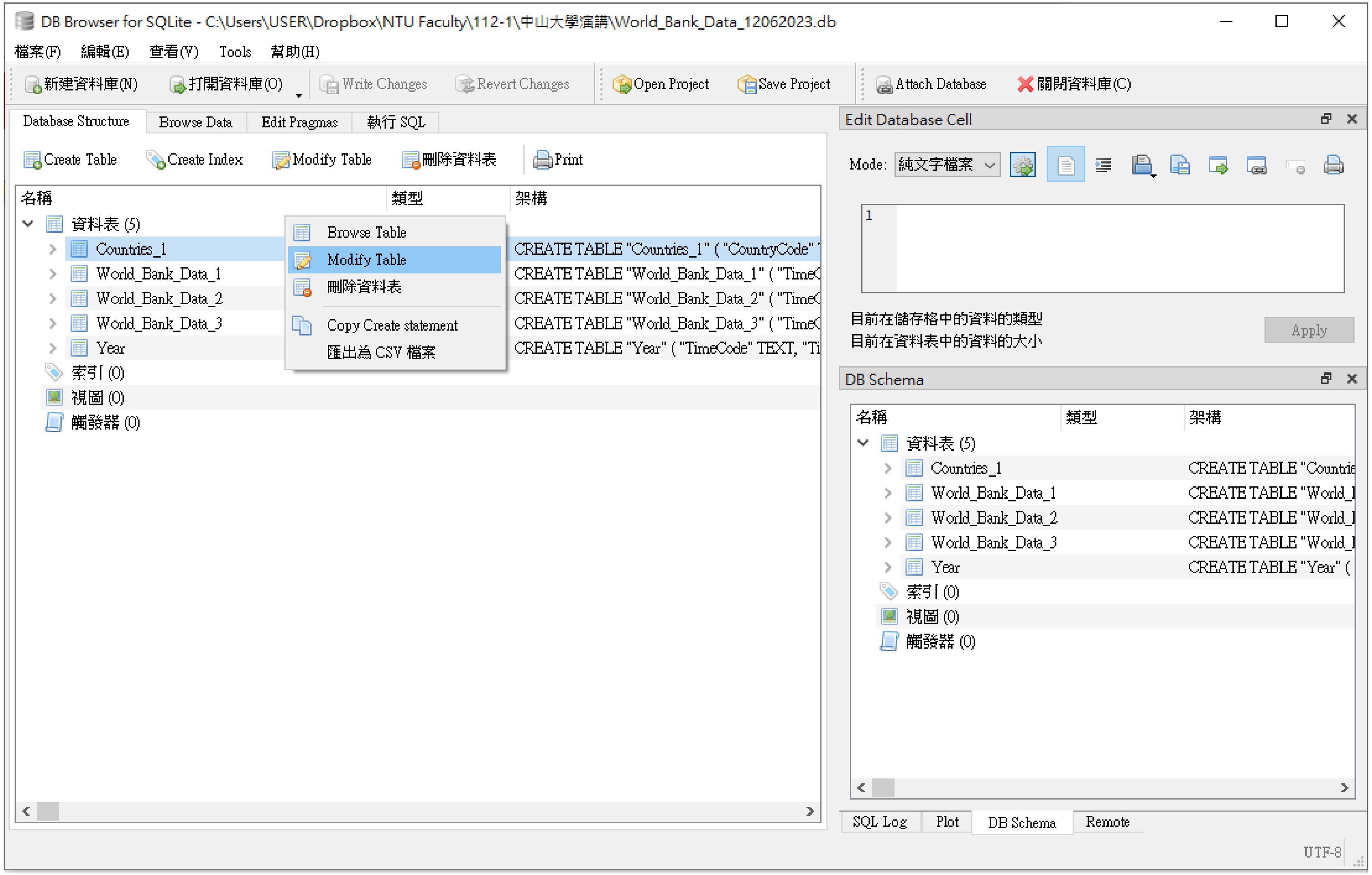


名稱	類型	架構
資料表 (5)		
Countries_1		CREATE TABLE "Countries_1" ("CountryCode"
World_Bank_Data_1		CREATE TABLE "World_Bank_Data_1" ("TimeC
World_Bank_Data_2		CREATE TABLE "World_Bank_Data_2" ("TimeC
World_Bank_Data_3		CREATE TABLE "World_Bank_Data_3" ("TimeC
Year		CREATE TABLE "Year" ("TimeCode" TEXT, "Ti
索引 (0)		
視圖 (0)		
觸發器 (0)		

1

目前在儲存格中的資料的類型
目前在資料表中的資料的大小

名稱	類型	架構
資料表 (5)		
Countries_1		CREATE TABLE "Countrie
World_Bank_Data_1		CREATE TABLE "World_I
World_Bank_Data_2		CREATE TABLE "World_I
World_Bank_Data_3		CREATE TABLE "World_I
Year		CREATE TABLE "Year" (
索引 (0)		
視圖 (0)		
觸發器 (0)		



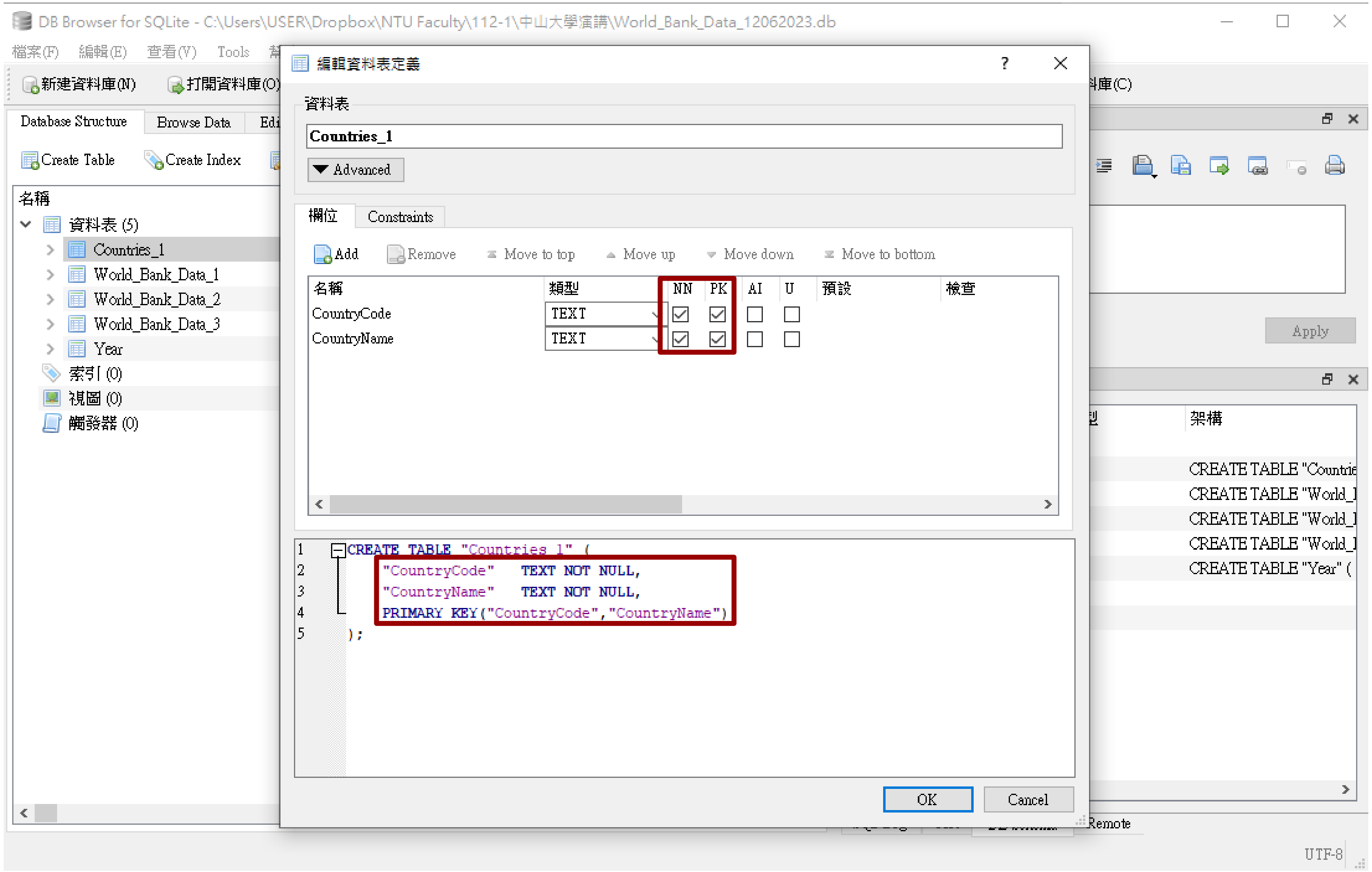
DB Browser for SQLite - C:\Users\USER\Dropbox\NTU Faculty\112-1\中山大學演講\World_Bank_Data_12062023.db

名稱	類型	架構
資料表 (5)		
Countries_1		CREATE TABLE "Countries_1" ("CountryCode"
World_Bank_Data_1		CREATE TABLE "World_Bank_Data_1" ("TimeC
World_Bank_Data_2		CREATE TABLE "World_Bank_Data_2" ("TimeC
World_Bank_Data_3		CREATE TABLE "World_Bank_Data_3" ("TimeC
Year		CREATE TABLE "Year" ("TimeCode" TEXT, "Ti
索引 (0)		
視圖 (0)		
觸發器 (0)		

1

目前在儲存格中的資料的類型
目前在資料表中的資料的大小

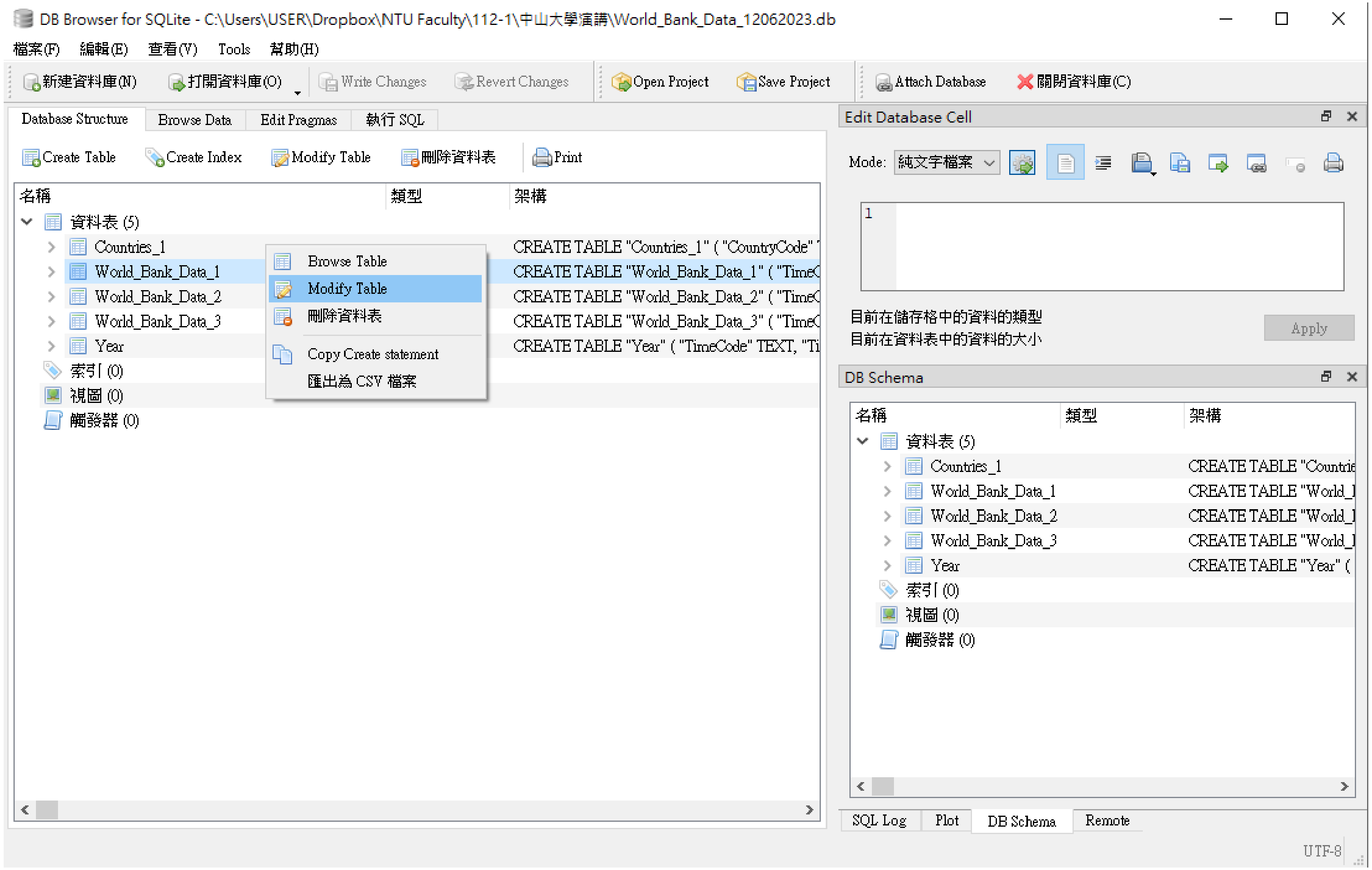
名稱	類型	架構
資料表 (5)		
Countries_1		CREATE TABLE "Countrie
World_Bank_Data_1		CREATE TABLE "World_I
World_Bank_Data_2		CREATE TABLE "World_I
World_Bank_Data_3		CREATE TABLE "World_I
Year		CREATE TABLE "Year" (
索引 (0)		
視圖 (0)		
觸發器 (0)		



資料表: Countries_1

名稱	類型	NN	PK	AI	U	預設	檢查
CountryCode	TEXT	☑	☑	□	□		
CountryName	TEXT	☑	☑	□	□		

```
CREATE TABLE "Countries_1" (
	"CountryCode"	TEXT NOT NULL,
	"CountryName"	TEXT NOT NULL,
	PRIMARY KEY("CountryCode","CountryName")
);
```



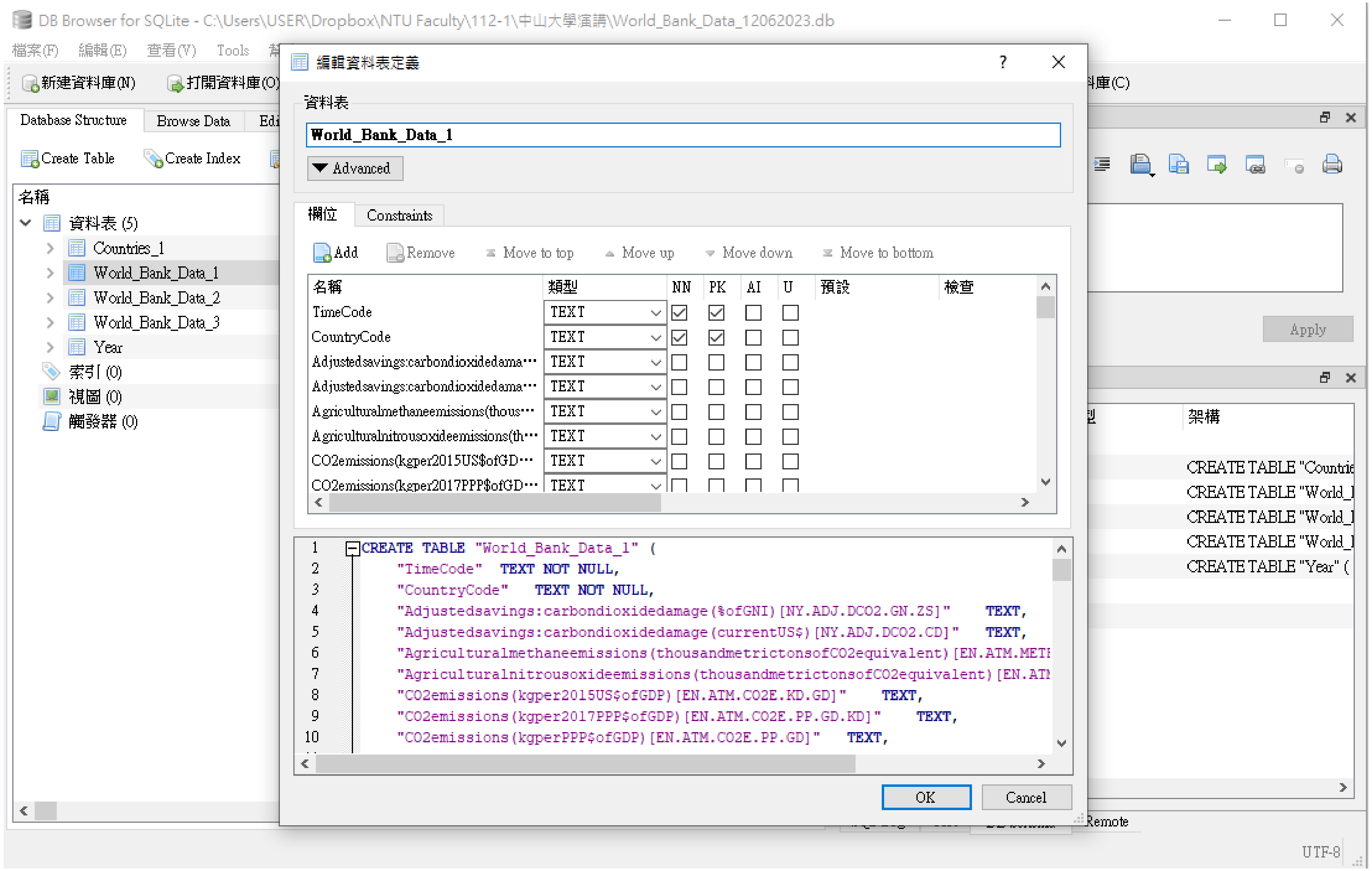
DB Browser for SQLite - C:\Users\USER\Dropbox\NTU Faculty\112-1\中山大學演講\World_Bank_Data_12062023.db

名稱	類型	架構
資料表 (5)		
Countries_1		CREATE TABLE "Countries_1" ("CountryCode"
World_Bank_Data_1		CREATE TABLE "World_Bank_Data_1" ("Time
World_Bank_Data_2		CREATE TABLE "World_Bank_Data_2" ("Time
World_Bank_Data_3		CREATE TABLE "World_Bank_Data_3" ("Time
Year		CREATE TABLE "Year" ("TimeCode" TEXT, "Ti
索引 (0)		
視圖 (0)		
觸發器 (0)		

1

目前在儲存格中的資料的類型
目前在資料表中的資料的大小

名稱	類型	架構
資料表 (5)		
Countries_1		CREATE TABLE "Countrie
World_Bank_Data_1		CREATE TABLE "World_
World_Bank_Data_2		CREATE TABLE "World_
World_Bank_Data_3		CREATE TABLE "World_
Year		CREATE TABLE "Year" (
索引 (0)		
視圖 (0)		
觸發器 (0)		

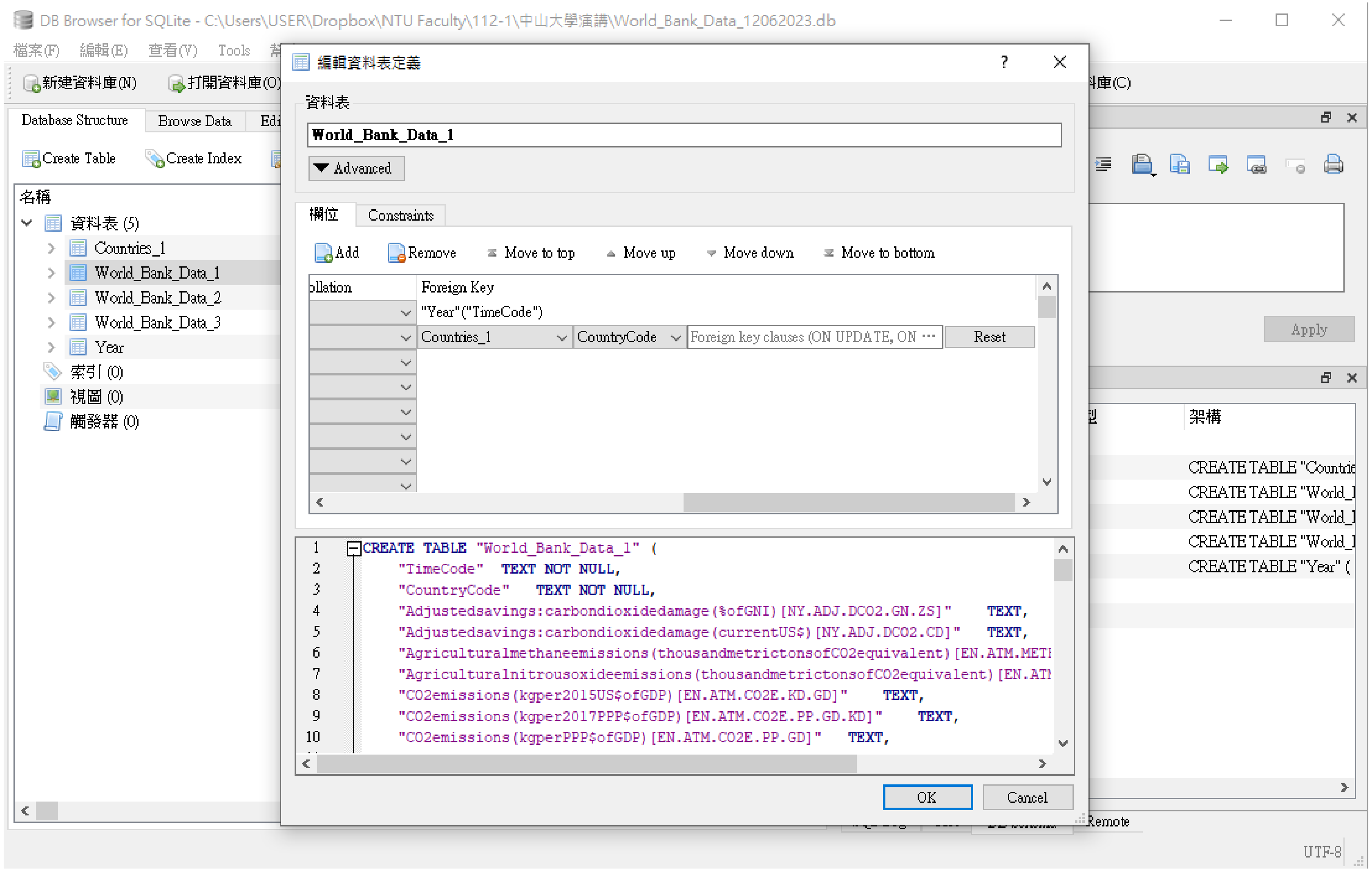


資料表: World_Bank_Data_1

名稱	類型	NN	PK	AI	U	預設	檢查
TimeCode	TEXT	☑	☑	□	□		
CountryCode	TEXT	☑	☑	□	□		
Adjustedsavings:carbondioxidedama···	TEXT	□	□	□	□		
Adjustedsavings:carbondioxidedama···	TEXT	□	□	□	□		
Agriculturalmethaneemissions(thous···	TEXT	□	□	□	□		
Agriculturalnitrousoxideemissions(th···	TEXT	□	□	□	□		
CO2emissions(kgper2015US$ofGD···	TEXT	□	□	□	□		
CO2emissions(kgper2017PPP$ofGD···	TEXT	□	□	□	□		

```
CREATE TABLE "World_Bank_Data_1" (
    "TimeCode"  TEXT NOT NULL,
    "CountryCode"   TEXT NOT NULL,
    "Adjustedsavings:carbondioxidedamage(%ofGNI)[NY.ADJ.DCO2.GN.ZS]"    TEXT,
    "Adjustedsavings:carbondioxidedamage(currentUS$)[NY.ADJ.DCO2.CD]"   TEXT,
    "Agriculturalmethaneemissions(thousandmetrictonsofCO2equivalent)[EN.ATM.METH
    "Agriculturalnitrousoxideemissions(thousandmetrictonsofCO2equivalent)[EN.ATM
    "CO2emissions(kgper2015US$ofGDP)[EN.ATM.CO2E.KD.GD]"    TEXT,
    "CO2emissions(kgper2017PPP$ofGDP)[EN.ATM.CO2E.PP.GD.KD]"    TEXT,
    "CO2emissions(kgperPPP$ofGDP)[EN.ATM.CO2E.PP.GD]"   TEXT,
```

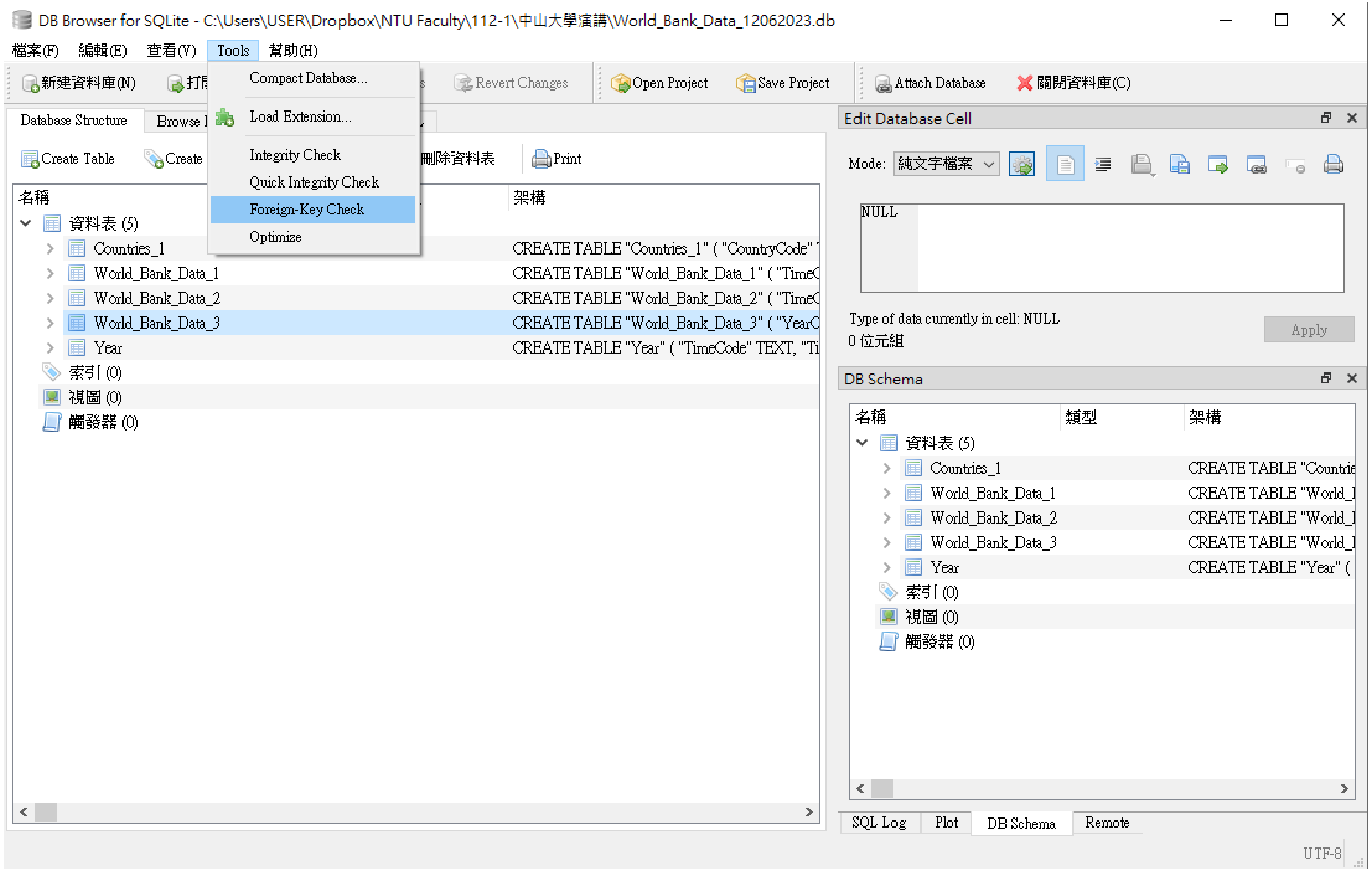
架構
CREATE TABLE "Countrie
CREATE TABLE "World_I
CREATE TABLE "World_I
CREATE TABLE "World_I
CREATE TABLE "Year" (



World_Bank_Data_1

ollation	Foreign Key			
	"Year"("TimeCode")			
	Countries_1	CountryCode	Foreign key clauses (ON UPDATE, ON ···	Reset

```
CREATE TABLE "World_Bank_Data_1" (
    "TimeCode"  TEXT NOT NULL,
    "CountryCode"   TEXT NOT NULL,
    "Adjustedsavings:carbondioxidedamage(%ofGNI)[NY.ADJ.DCO2.GN.ZS]"    TEXT,
    "Adjustedsavings:carbondioxidedamage(currentUS$)[NY.ADJ.DCO2.CD]"   TEXT,
    "Agriculturalmethaneemissions(thousandmetrictonsofCO2equivalent)[EN.ATM.METH
    "Agriculturalnitrousoxideemissions(thousandmetrictonsofCO2equivalent)[EN.ATM
    "CO2emissions(kgper2015US$ofGDP)[EN.ATM.CO2E.KD.GD]"    TEXT,
    "CO2emissions(kgper2017PPP$ofGDP)[EN.ATM.CO2E.PP.GD.KD]"    TEXT,
    "CO2emissions(kgperPPP$ofGDP)[EN.ATM.CO2E.PP.GD]"   TEXT,
```



名稱	架構
資料表 (5)	
Countries_1	CREATE TABLE "Countries_1" ("CountryCode" "
World_Bank_Data_1	CREATE TABLE "World_Bank_Data_1" ("TimeC
World_Bank_Data_2	CREATE TABLE "World_Bank_Data_2" ("TimeC
World_Bank_Data_3	CREATE TABLE "World_Bank_Data_3" ("YearC
Year	CREATE TABLE "Year" ("TimeCode" TEXT, "Ti
索引 (0)	
視圖 (0)	
觸發器 (0)	

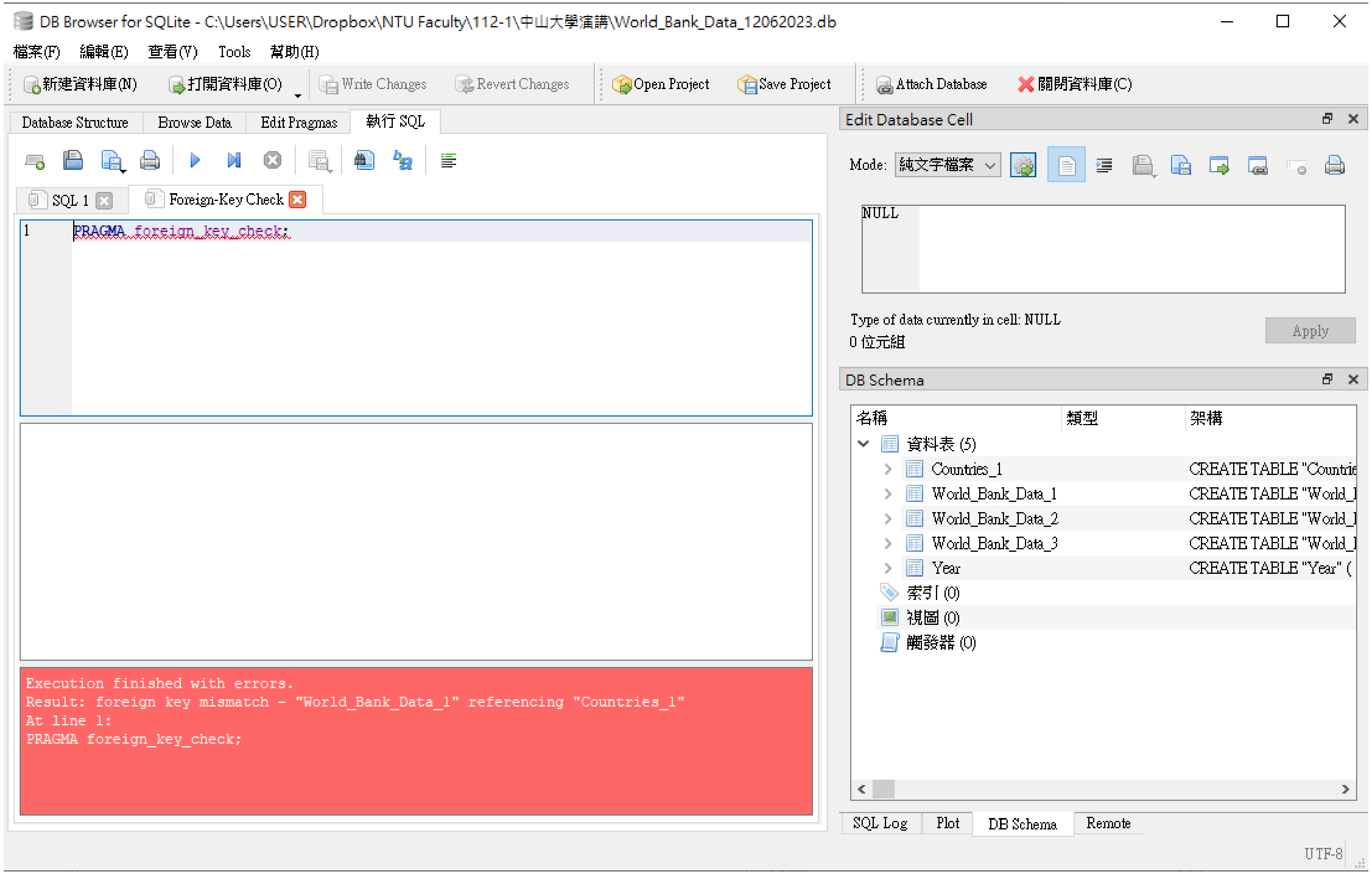
Mode: 純文字檔案

NULL

Type of data currently in cell: NULL

0 位元組

名稱	類型	架構
資料表 (5)		
Countries_1		CREATE TABLE "Countrie
World_Bank_Data_1		CREATE TABLE "World_I
World_Bank_Data_2		CREATE TABLE "World_I
World_Bank_Data_3		CREATE TABLE "World_I
Year		CREATE TABLE "Year" (
索引 (0)		
視圖 (0)		
觸發器 (0)		



DB Browser for SQLite - C:\Users\USER\Dropbox\NTU Faculty\112-1\中山大學演講\World_Bank_Data_12062023.db

```
PRAGMA foreign_key_check;
```

```
Execution finished with errors.
Result: foreign key mismatch - "World_Bank_Data_1" referencing "Countries_1"
At line 1:
PRAGMA foreign_key_check;
```

Mode: 純文字檔案

NULL

Type of data currently in cell: NULL

0 位元組

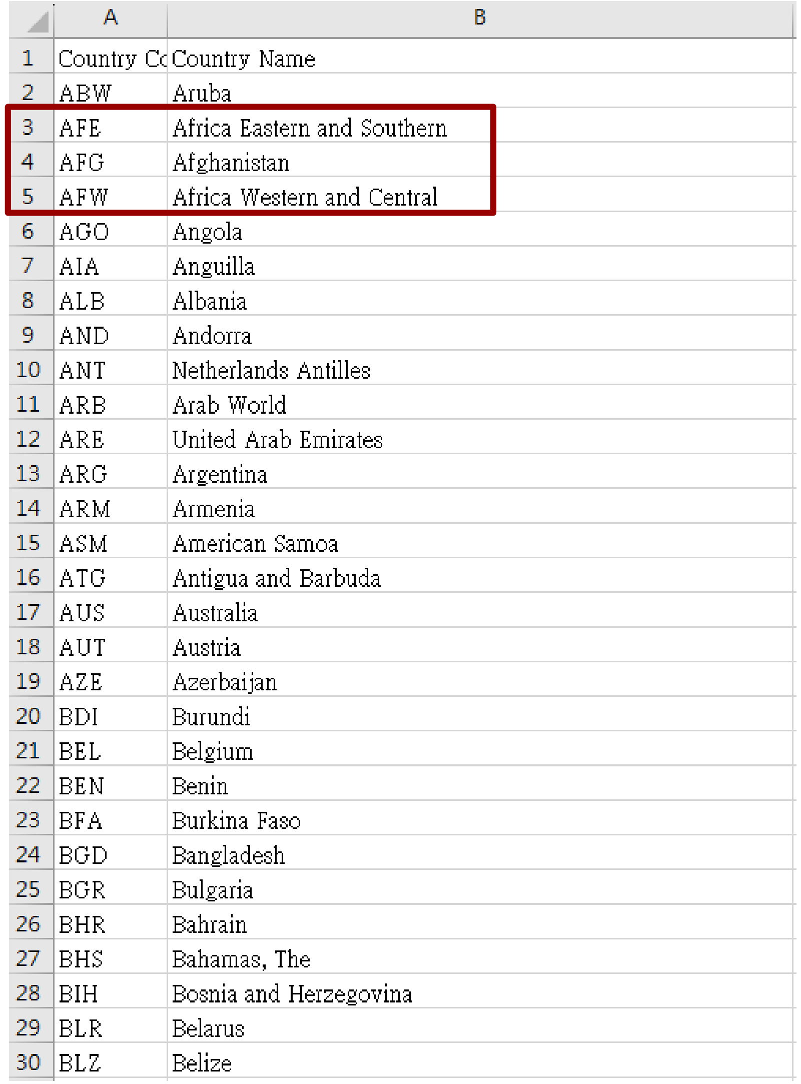
名稱	類型	架構
資料表 (5)		
Countries_1		CREATE TABLE "Countrie
World_Bank_Data_1		CREATE TABLE "World_I
World_Bank_Data_2		CREATE TABLE "World_I
World_Bank_Data_3		CREATE TABLE "World_I
Year		CREATE TABLE "Year" (
索引 (0)		
視圖 (0)		
觸發器 (0)		

Reason

- World_Bank_Data_1 & World_Bank_Data_3
  - **AFE**: Africa Eastern and Southern
  - **AFW**: Africa Western and Central
  - **INX**: Not classified

- World_Bank_Data_2
  - **AIA**: Anguilla
  - **COK**: Cook Islands
  - **FTI**: Global Partnership for Education
  - **LNX**: Lending category not classified
  - **NIU**: Niue
  - **TKL**: Tokelau

Countries_2



	A	B
1	Country Cc	Country Name
2	ABW	Aruba
3	AFE	Africa Eastern and Southern
4	AFG	Afghanistan
5	AFW	Africa Western and Central
6	AGO	Angola
7	AIA	Anguilla
8	ALB	Albania
9	AND	Andorra
10	ANT	Netherlands Antilles
11	ARB	Arab World
12	ARE	United Arab Emirates
13	ARG	Argentina
14	ARM	Armenia
15	ASM	American Samoa
16	ATG	Antigua and Barbuda
17	AUS	Australia
18	AUT	Austria
19	AZE	Azerbaijan
20	BDI	Burundi
21	BEL	Belgium
22	BEN	Benin
23	BFA	Burkina Faso
24	BGD	Bangladesh
25	BGR	Bulgaria
26	BHR	Bahrain
27	BHS	Bahamas, The
28	BIH	Bosnia and Herzegovina
29	BLR	Belarus
30	BLZ	Belize

referential integrity constraint

Queries

- Users may want **specific** information found in a relational database and not have to sort through all the files to get that information. So, they query (ask a question) the data.

- An example of a query might be:
  - List the GDP of **United States and China**.
  - List the GDP of countries having **CO_2 emissions (kg per 2015 US$ of GDP) higher than 1.5** between 2001 and 2020.
  - List the GDP of countries having **short-term debt (% of total external debt) higher than 20** between 2001 and 2020.

SQL Syntax Basics

```
SELECT   …
FROM     … ;
```

[Syntax]

```
SELECT  CountryCode
FROM  World_Bank_Data_1;
```

SQL Syntax Basics

```
SELECT   …
FROM     … ;
```

[Syntax]

```
SELECT  *
FROM  World_Bank_Data_1;
```

SQL Syntax Basics

```
SELECT   …
FROM     … ;
```

[Syntax]

```
SELECT  DISTINCT CountryCode
FROM  World_Bank_Data_1;
```

SQL Syntax Basics

```
SELECT   …
FROM     …
WHERE  … ;
```

```
[Syntax]
SELECT   *
FROM World_Bank_Data_1
WHERE CountryCode = USA;
```

✖

SQL Syntax Basics

SELECT …

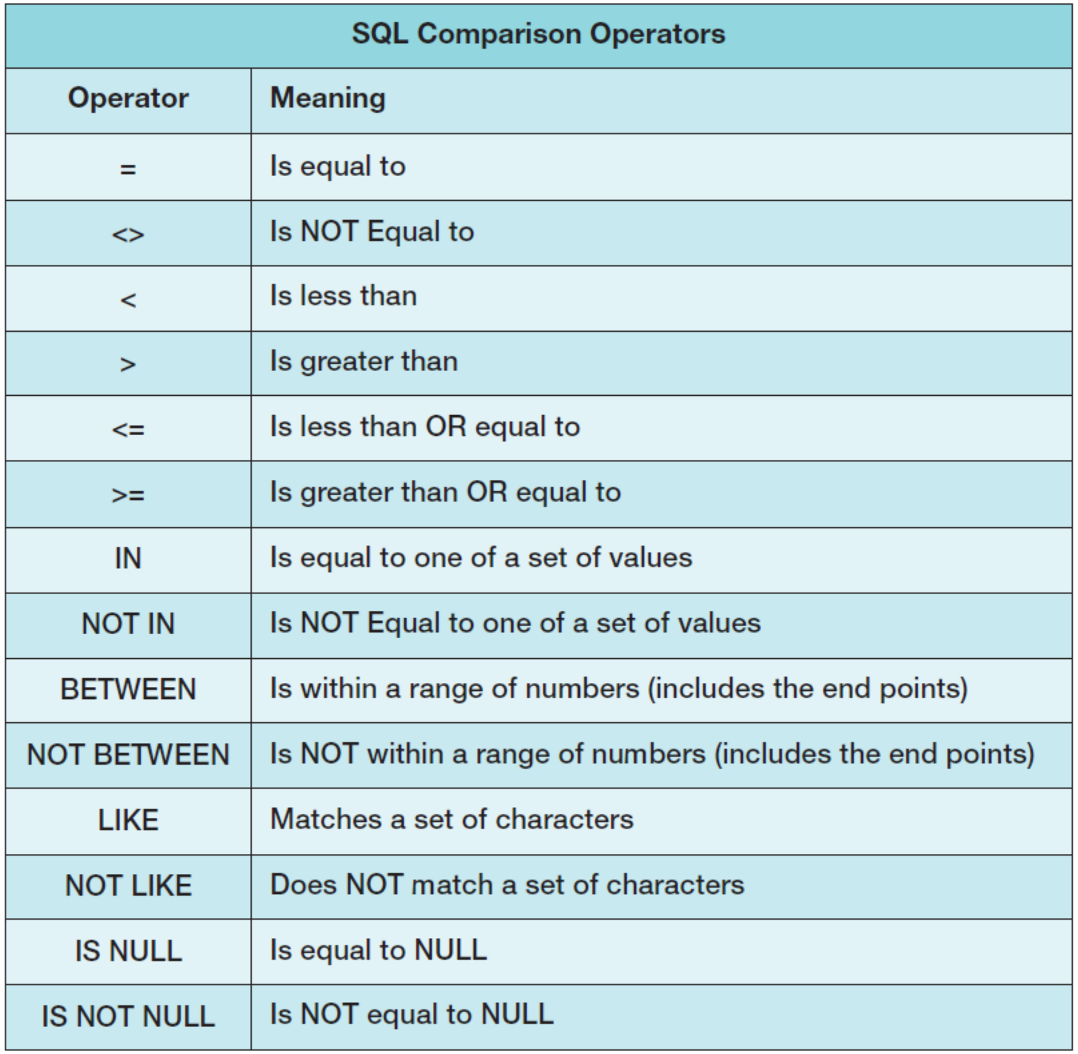
FROM …

WHERE … ;

[Syntax]

```
SELECT   *
FROM World_Bank_Data_1
WHERE CountryCode = 'USA';
```

SQL Comparison Operators



SQL Comparison Operators	
Operator	**Meaning**
=	Is equal to
<>	Is NOT Equal to
<	Is less than
>	Is greater than
<=	Is less than OR equal to
>=	Is greater than OR equal to
IN	Is equal to one of a set of values
NOT IN	Is NOT Equal to one of a set of values
BETWEEN	Is within a range of numbers (includes the end points)
NOT BETWEEN	Is NOT within a range of numbers (includes the end points)
LIKE	Matches a set of characters
NOT LIKE	Does NOT match a set of characters
IS NULL	Is equal to NULL
IS NOT NULL	Is NOT equal to NULL

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

SQL Syntax Basics

```
SELECT   …
FROM     …
WHERE  … ;
```

[Syntax]

```
SELECT   *
FROM World_Bank_Data_1
WHERE CountryCode = 'USA'
    OR CountryCode = 'CHN';
```

SQL Syntax Basics

```
SELECT   …
FROM     …
WHERE  …
   IN;
```

[Syntax]

```
SELECT   *
FROM World_Bank_Data_1
WHERE CountryCode IN ('USA', 'CHN', 'JPN');
```

SQL Syntax Basics

```
SELECT   …
FROM     …
WHERE  …
   IN;
```

[Syntax]

```
SELECT   *
FROM World_Bank_Data_1
WHERE CountryCode NOT IN ('USA', 'CHN', 'JPN');
```

SQL Syntax Basics

```
SELECT   …
FROM     …
WHERE  …
   AND;
```

[Syntax]

```
SELECT   *
FROM  World_Bank_Data_1
WHERE CO2_emissions_kg_per_2015_USD_of_GDP >= 4.0
   AND CO2_emissions_kg_per_2015_USD_of_GDP <= 5.0;
```

SQL Syntax Basics

```
SELECT   …
FROM     …
WHERE  …
     BETWEEN     AND ;
```

[Syntax]

```
SELECT   *
FROM  World_Bank_Data_1
WHERE CO2_emissions_kg_per_2015_USD_of_GDP BETWEEN
4.0 AND 5.0 ;
```

SQL Syntax Basics

```
SELECT   …
FROM     …
WHERE  …
    NOT BETWEEN     AND ;
```

[Syntax]

```
SELECT   *
FROM  World_Bank_Data_1
WHERE CO2_emissions_kg_per_2015_USD_of_GDP NOT
BETWEEN 4.0 AND 5.0 ;
```

SQL Syntax Basics

```
SELECT   …
FROM    …
WHERE  …
   LIKE;
```

%: represent zero, one, or more character

[Syntax]

```
SELECT   *
FROM  World_Bank_Data_1
WHERE CountryCode LIKE 'U%' ;
```

SQL Syntax Basics

```
SELECT   …
FROM     …
WHERE  …
   LIKE;
```

[Syntax]

```
SELECT   *
FROM World_Bank_Data_1
WHERE CountryCode LIKE '%U%' ;
```

SQL Syntax Basics

SELECT …

FROM …

WHERE …

LIKE;

_: represent one character

[Syntax]

SELECT *

FROM World_Bank_Data_1

WHERE CountryCode **LIKE '_U%'** ;

SQL Syntax Basics

```
SELECT   …
FROM     …
WHERE  …
   LIKE;
```

[Syntax]

```
SELECT   *
FROM World_Bank_Data_1
WHERE CountryCode LIKE '_U%'
     AND TimeCode IN ('YR2017', 'YR2018', 'YR2019');
```

SQL Syntax Basics

```
SELECT   …
FROM     …
WHERE  …
   IS NULL;
```

[Syntax]

```
SELECT   TimeCode, CountryCode, GDP_current_USD
FROM  World_Bank_Data_1
WHERE GDP_current_USD IS NULL ;
```

SQL Syntax Basics

```
SELECT   …
FROM     …
WHERE  …
   IS NOT NULL;
```

[Syntax]

```
SELECT   TimeCode, CountryCode, GDP_current_USD
FROM  World_Bank_Data_1
WHERE GDP_current_USD IS NOT NULL ;
```

SQL Syntax Basics

```
SELECT   …
FROM    …
ORDER BY  … ;
```

[Syntax]

```
SELECT   TimeCode, CountryCode, GDP_current_USD
FROM  World_Bank_Data_1
WHERE GDP_current_USD IS NOT NULL
ORDER BY GDP_current_USD;
```

SQL Syntax Basics

```
SELECT   …
FROM     …
ORDER BY  … ;
```

[Syntax]

```
SELECT   TimeCode, CountryCode, GDP_current_USD
FROM  World_Bank_Data_1
WHERE GDP_current_USD IS NOT NULL
ORDER BY GDP_current_USD DESC;
```

SQL Syntax Basics

- List the GDP of countries having **CO_2 emissions (kg per 2015 US$ of GDP) higher than 1.5** and sort by the CO_2 emissions (descending).

[Syntax]

```
SELECT   TimeCode, CountryCode, GDP_current_USD,
         CO2_emissions_kg_per_2015_USD_of_GDP
FROM  World_Bank_Data_1
WHERE GDP_current_USD IS NOT NULL
    AND CO2_emissions_kg_per_2015_USD_of_GDP > 1.5
ORDER BY CO2_emissions_kg_per_2015_USD_of_GDP DESC;
```

SQL Syntax Coding Convention

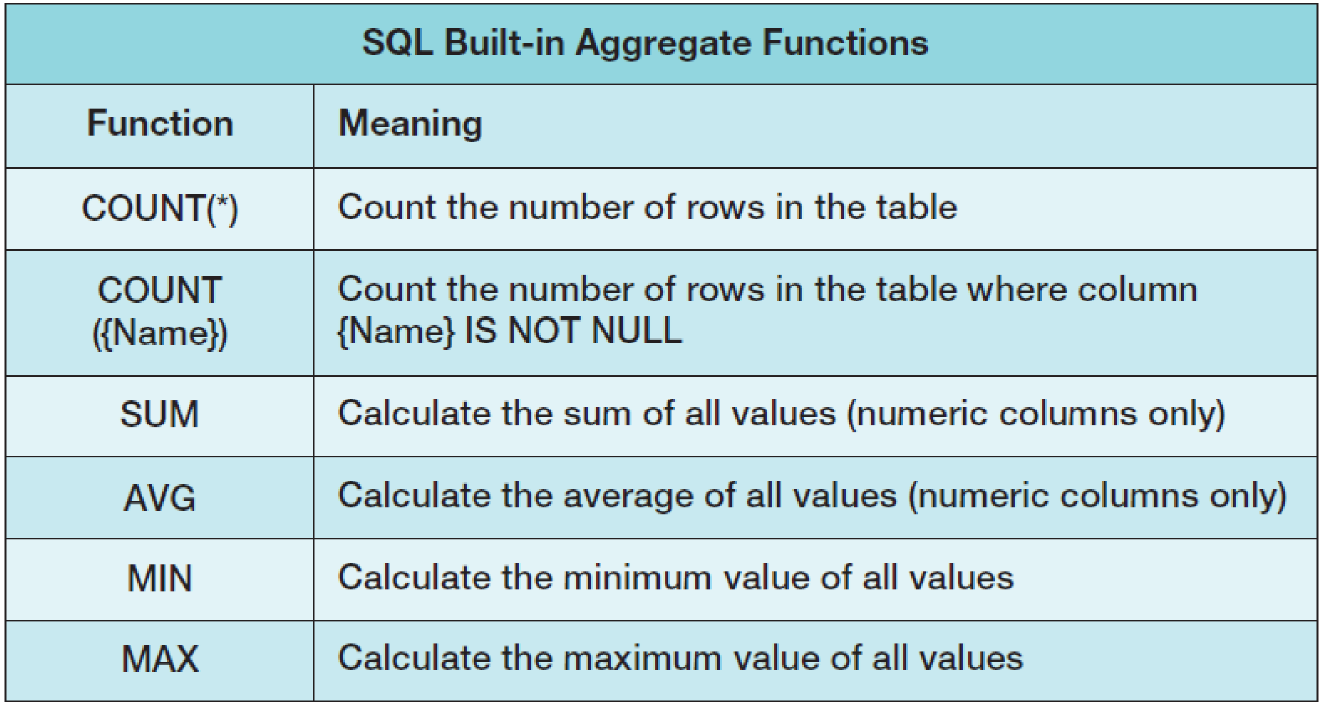
[Syntax]

```
SELECT   TimeCode, CountryCode, GDP_current_USD,
         CO2_emissions_kg_per_2015_USD_of_GDP
FROM  World_Bank_Data_1
WHERE GDP_current_USD IS NOT NULL
   AND CO2_emissions_kg_per_2015_USD_of_GDP > 1.5
ORDER BY CO2_emissions_kg_per_2015_USD_of_GDP DESC;
```

[Syntax]

```
SELECT   TimeCode, CountryCode, GDP_current_USD,
CO2_emissions_kg_per_2015_USD_of_GDP  FROM  World_Bank_Data_1 WHERE
GDP_current_USD IS NOT NULL  AND CO2_emissions_kg_per_2015_USD_of_GDP
> 1.5 ORDER BY CO2_emissions_kg_per_2015_USD_of_GDP DESC;
```

SQL Built-in Aggregate Functions



SQL Built-in Aggregate Functions	
Function	**Meaning**
COUNT(*)	Count the number of rows in the table
COUNT ({Name})	Count the number of rows in the table where column {Name} IS NOT NULL
SUM	Calculate the sum of all values (numeric columns only)
AVG	Calculate the average of all values (numeric columns only)
MIN	Calculate the minimum value of all values
MAX	Calculate the maximum value of all values

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

SQL Syntax Basics

```
SELECT   SUM…
FROM     … ;
```

[Syntax]

```
SELECT   AVG(GDP_current_USD)
FROM     World_Bank_Data_1;
```

SQL Syntax Basics

```
SELECT   SUM…
FROM     … ;
```

[Syntax]

```
SELECT   AVG(GDP_current_USD)
FROM     World_Bank_Data_1
WHERE  GDP_current_USD IS NOT NULL;
```

SQL Syntax Basics

```
SELECT   SUM… AS …
FROM     … ;
```

[Syntax]

```
SELECT   AVG(GDP_current_USD) AS GDP_AVG
FROM     World_Bank_Data_1
WHERE  GDP_current_USD IS NOT NULL;
```

SQL Syntax Basics

```
SELECT   SUM… AS …
FROM     … ;
```

[Syntax]

```
SELECT   AVG(GDP_current_USD) AS GDP_AVG,
         SUM(GDP_current_USD) AS GDP_SUM,
         MIN(GDP_current_USD) AS GDP_MIN,
          MAX(GDP_current_USD) AS GDP_MAX
FROM     World_Bank_Data_1
WHERE    GDP_current_USD IS NOT NULL;
```

SQL Syntax Basics

```
SELECT   COUNT… AS …
FROM     … ;
```

[Syntax]

```
SELECT   COUNT(CountryCode) AS Country_Count
FROM     World_Bank_Data_1
```

SQL Syntax Basics

```
SELECT   COUNT… AS …
FROM    … ;
```

[Syntax]

```
SELECT   COUNT(CountryCode) AS Country_Count
FROM  (SELECT DISTINCT CountryCode
              FROM World_Bank_Data_1) ;
```

SQL Syntax Basics

```
SELECT   … || …
FROM     … ;
```

concatenate operator (||)

[Syntax]

```
SELECT   TimeCode, CountryCode, GDP_current_USD,
         (CountryCode ||  ' in ' ||   TimeCode) AS Country_Year
FROM      World_Bank_Data_1
WHERE     GDP_current_USD IS NOT NULL
ORDER BY  GDP_current_USD DESC;
```

SQL Syntax Basics

```
SELECT   … || …
FROM     …
WHERE    …
GROUP BY … ;
```

[Syntax]

```
SELECT     CountryCode, AVG(GDP_current_USD) AS GDP_AVG,
           COUNT (GDP_current_USD) AS GDP_Count
FROM       World_Bank_Data_1
WHERE      GDP_current_USD IS NOT NULL
GROUP BY   CountryCode ;
```

SQL Syntax Basics

```
SELECT   … || …
FROM     …
WHERE    …
GROUP BY …
HAVING   … ;
```

[Syntax]

```
SELECT      CountryCode, AVG(GDP_current_USD) AS GDP_AVG,
            COUNT (GDP_current_USD) AS GDP_Count
FROM        World_Bank_Data_1
WHERE       GDP_current_USD IS NOT NULL
GROUP BY    CountryCode
HAVING      COUNT (GDP_current_USD) > 40;
```

SQL Syntax Basics

```
SELECT   … || …
FROM     …
WHERE    …
GROUP BY …
HAVING   … ;
```

[Syntax]

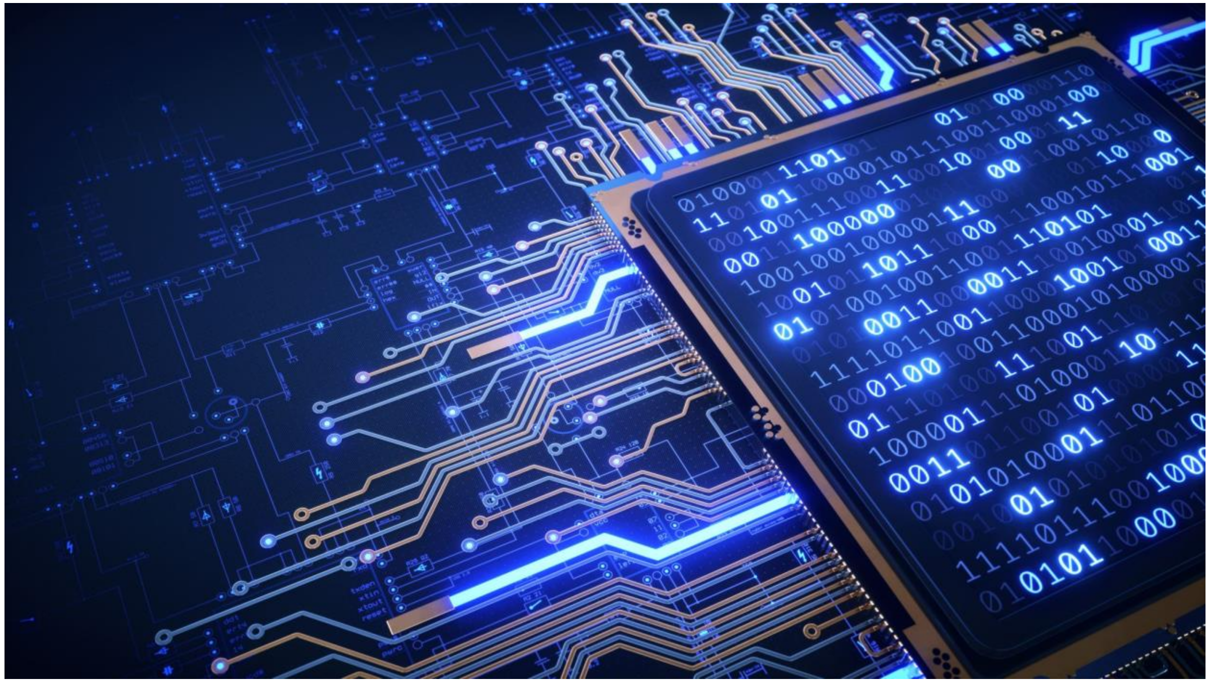
```
SELECT      CountryCode, AVG(GDP_current_USD) AS GDP_AVG,
            COUNT (GDP_current_USD) AS GDP_Count
FROM        World_Bank_Data_1
WHERE       GDP_current_USD IS NOT NULL
GROUP BY    CountryCode
HAVING      GDP_Count > 40;
```

SQL Syntax Basics

```
SELECT   … || …
FROM    …
WHERE  …
GROUP BY …
HAVING  …
ORDER BY …;
```

[Syntax]

```
SELECT        CountryCode, AVG(GDP_current_USD) AS GDP_AVG,
              COUNT (GDP_current_USD) AS GDP_Count
FROM          World_Bank_Data_1
WHERE         GDP_current_USD IS NOT NULL
GROUP BY      CountryCode
HAVING        GDP_Count > 40
ORDER BY      GDP_Count DESC;
```



Querying Two or More Tables with SQL

SQL Subquery

- Show the countries names whose **government expenditures on education have ever been greater than 8% of GDP**.
  - Government_expenditure_on_education_GDP_percent
    → World_Bank_Data_2
  - CountryName
    → Countries_2

SQL Subquery

```
SELECT  CountryCode
FROM    World_Bank_Data_2
WHERE   Government_expenditure_on_education_GDP_percent
        > 8.0;
```

SQL Subquery

```
SELECT   CountryName
FROM    Countries_2                The Second Query
WHERE  CountryCode IN
          (SELECT  CountryCode
           FROM    World_Bank_Data_2          The First Query
           WHERE  Government_expenditure_on_education_GDP_percent
              > 8.0)
ORDER BY CountryName DESC;
```

Challenge!!

Show the countries names whose whose **age 15-64 populations have ever been greater than 60% of total population**.

Hint:

World_Bank_Data_3

Population_ages_15_64_population_percent

SQL Join

```
SELECT DISTINCT CountryName
FROM    Countries_2, World_Bank_Data_2
WHERE  Countries_2.CountryCode = World_Bank_Data_2.CountryCode
       AND Government_expenditure_on_education_GDP_percent > 8.0
ORDER BY CountryName DESC;
```

SQL Join

- The process of creating a result table by joining two tables via an SQL join operation is called **joining the two tables**.

- When the tables are joined using an **inner join** with an **is equal to condition**, this join is called an **equijoin**.

- When people say join, 99% of the time they mean an **equijoin**.

Challenge!!

Show the countries names whose whose **age 15-64 populations have ever been greater than 60% of total population. (JOIN)**

Hint:

World_Bank_Data_3

Population_ages_15_64_population_percent

Comparing Subqueries and Joins

- A **subquery** can be used only to retrieve data from **the top table.**
- A **join** can be used to obtain data from **any number of tables**.

Subquery

```
SELECT CountryName, TimeCode,
            Government_expenditure_on_education_GDP_percent
FROM    Countries_2
WHERE  CountryCode IN
             (SELECT  CountryCode
               FROM    World_Bank_Data_2
               WHERE  Government_expenditure_on_education_GDP_percent
                  > 8.0)
ORDER BY CountryName DESC;
```

✗

JOIN

```
SELECT CountryName, TimeCode,
            Government_expenditure_on_education_GDP_percent
FROM    Countries_2, World_Bank_Data_2
WHERE  Countries_2.CountryCode = World_Bank_Data_2.CountryCode
        AND Government_expenditure_on_education_GDP_percent > 8.0
ORDER BY CountryName DESC;
```

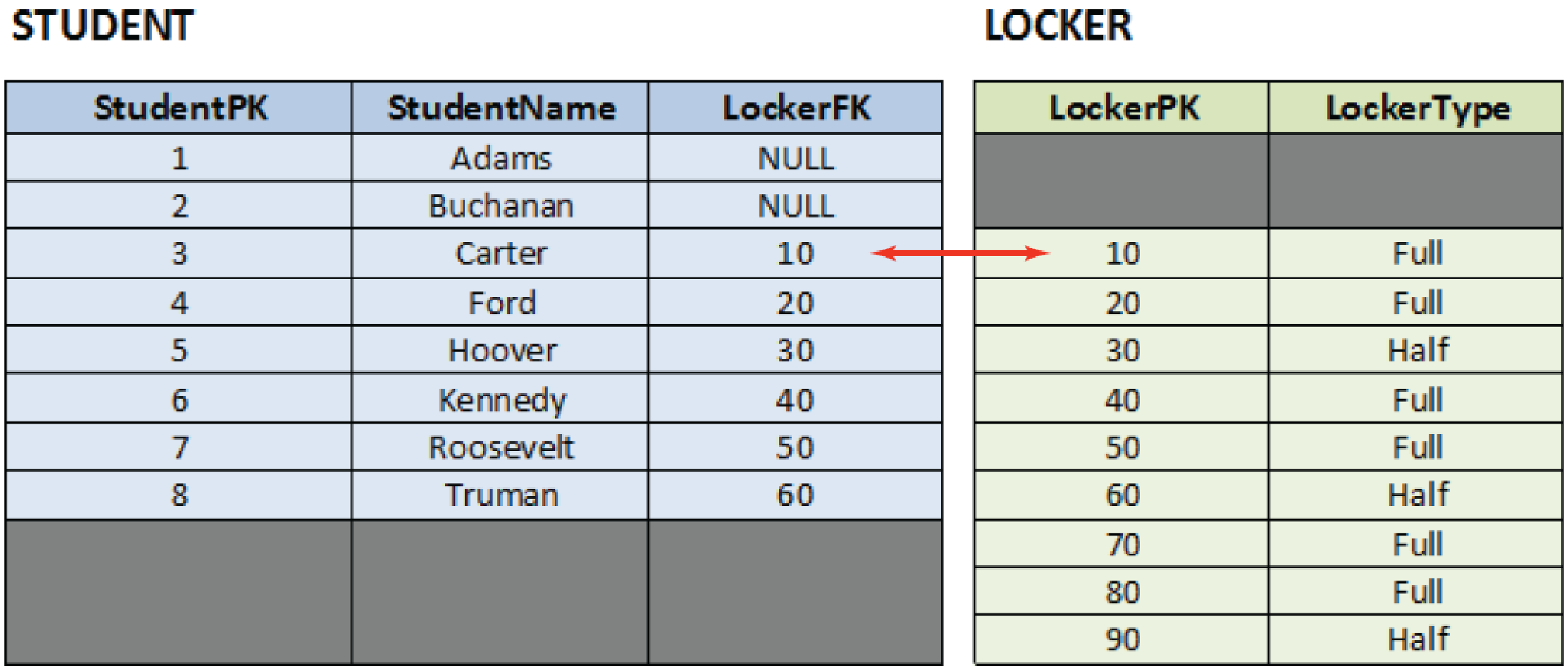
Super Challenge!!!

List the GDP (US$) of countries whose **(1) government expenditures on education are greater than 8% of GDP & age 15-64 populations are greater than 60% of total population**. **Show the country names and years**.

Hint:
World_Bank_Data_1
GDP_current_USD
World_Bank_Data_2
Government_expenditure_on_education_GDP_percent
World_Bank_Data_3
Population_ages_15_64_population_percent

Year & Countries_2

SQL Outer Join On



STUDENT

StudentPK	StudentName	LockerFK
1	Adams	NULL
2	Buchanan	NULL
3	Carter	10
4	Ford	20
5	Hoover	30
6	Kennedy	40
7	Roosevelt	50
8	Truman	60

LOCKER

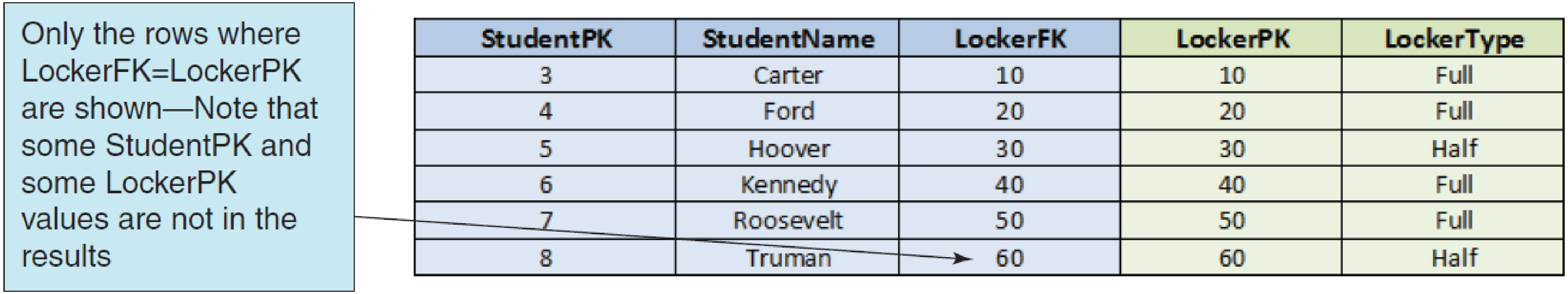
LockerPK	LockerType
10	Full
20	Full
30	Half
40	Full
50	Full
60	Half
70	Full
80	Full
90	Half

(a) The STUDENT and LOCKER Tables Aligned to Show Row Relationships

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

SQL Inner Join On

```
SELECT StudentPK, StudentName, LockerFK, LockerPK, LockerType
FROM STUDENT INNER JOIN LOCKER
    ON STUDENT.LockerFK = LOCKER.LockerPK
```



Only the rows where LockerFK=LockerPK are shown—Note that some StudentPK and some LockerPK values are not in the results

StudentPK	StudentName	LockerFK	LockerPK	LockerType
3	Carter	10	10	Full
4	Ford	20	20	Full
5	Hoover	30	30	Half
6	Kennedy	40	40	Full
7	Roosevelt	50	50	Full
8	Truman	60	60	Half

(b) INNER JOIN of the STUDENT and LOCKER Tables

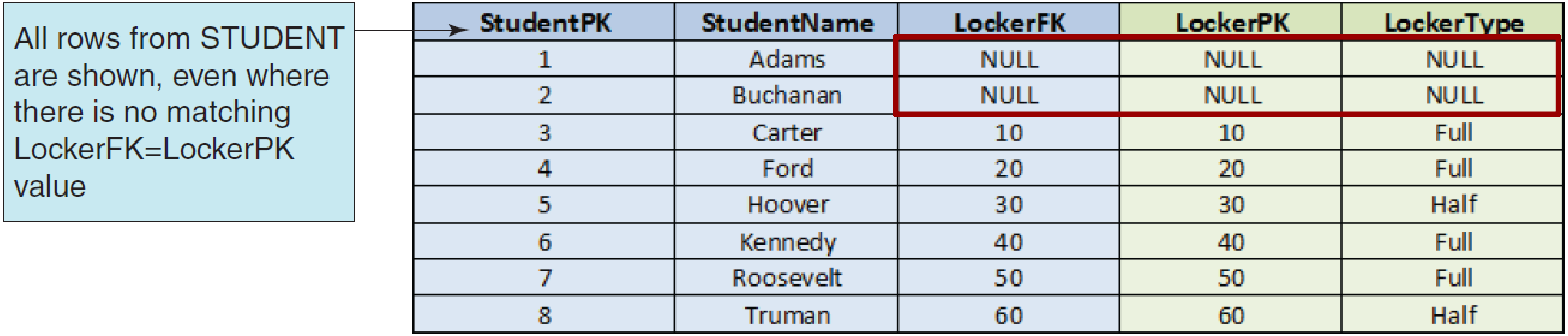
D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

SQL LEFT Outer Join On

SELECT StudentPK, StudentName, LockerFK, LockerPK, LockerType

FROM **STUDENT LEFT OUTER JOIN LOCKER**

ON STUDENT.LockerFK = LOCKER.LockerPK



All rows from STUDENT are shown, even where there is no matching LockerFK=LockerPK value

StudentPK	StudentName	LockerFK	LockerPK	LockerType
1	Adams	NULL	NULL	NULL
2	Buchanan	NULL	NULL	NULL
3	Carter	10	10	Full
4	Ford	20	20	Full
5	Hoover	30	30	Half
6	Kennedy	40	40	Full
7	Roosevelt	50	50	Full
8	Truman	60	60	Half

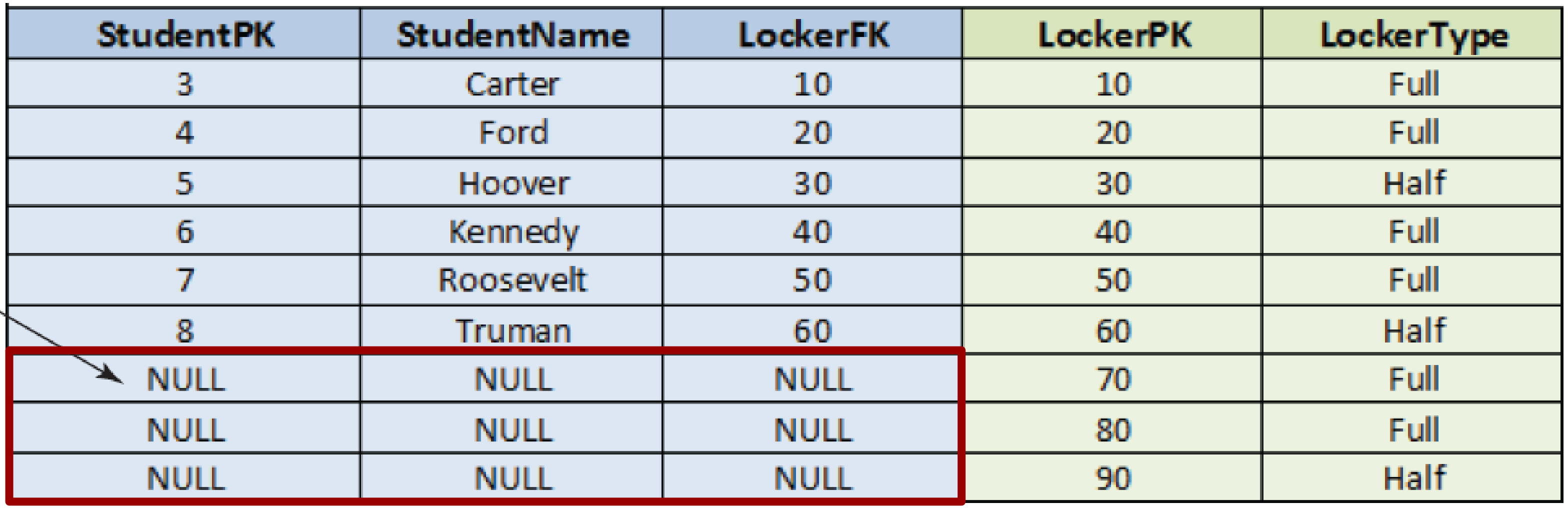
(c) LEFT OUTER JOIN of the STUDENT and LOCKER Tables

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.

SQL RIGHT Outer Join On

```
SELECT StudentPK, StudentName, LockerFK, LockerPK, LockerType
FROM STUDENT RIGHT OUTER JOIN LOCKER
    ON STUDENT.LockerFK = LOCKER.LockerPK
```

All rows from LOCKER are shown, even where there is no matching LockerFK=LockerPK value



StudentPK	StudentName	LockerFK	LockerPK	LockerType
3	Carter	10	10	Full
4	Ford	20	20	Full
5	Hoover	30	30	Half
6	Kennedy	40	40	Full
7	Roosevelt	50	50	Full
8	Truman	60	60	Half
NULL	NULL	NULL	70	Full
NULL	NULL	NULL	80	Full
NULL	NULL	NULL	90	Half

(d) RIGHT OUTER JOIN of the STUDENT and LOCKER Tables

D. Kroenke and D. Auer, DATABASE PROCESSING: FUNDAMENTALS, DESIGN & IMPLEMENTATION, 14th Edition, Pearson/Prentice-Hall, 2016.