

```
In [88]: import pandas as pd  
import numpy as np
```

```
In [8]: person = pd.read_csv("Person.Person.csv", sep = ";")
```

```
In [9]: person.head()
```

Out[9]:

	BusinessEntityID	PersonType	NameStyle	Title	FirstName	MiddleName	LastName	Suffix	...
0	1	EM	0	NaN	Ken	J	S?nchez	NaN	
1	2	EM	0	NaN	Terri	Lee	Duffy	NaN	
2	3	EM	0	NaN	Roberto	NaN	Tamburello	NaN	
3	4	EM	0	NaN	Rob	NaN	Walters	NaN	

```
In [145]: person2 = person[["BusinessEntityID", "FirstName"]]  
person2.head()
```

Out[145]:

	BusinessEntityID	FirstName
0	1	Ken
1	2	Terri
2	3	Roberto
3	4	Rob
4	5	Gail

```
In [151]: person2.to_csv("01seller.csv", index=False)
```

```
In [12]: person['AdditionalContactInfo'].isna().sum()
```

Out[12]: 19962

```
In [25]: person[person.AdditionalContactInfo.notnull()]
```

Out[25]:

	BusinessEntityID	PersonType	NameStyle	Title	FirstName	MiddleName	LastName	Suffix
290	291	SC	0	Mr.	Gustavo	NaN	Achong	NaN
291	293	SC	0	Ms.	Catherine	R.	Abel	NaN
292	295	SC	0	Ms.	Kim	NaN	Abercrombie	NaN
293	297	SC	0	Sr.	Humberto	NaN	Acevedo	NaN
294	299	SC	0	Sra.	Pilar	NaN	Ackerman	NaN
295	301	SC	0	Ms.	Frances	B.	Adams	NaN
296	303	SC	0	Ms.	Margaret	J.	Smith	NaN
297	305	SC	0	Ms.	Carla	J.	Adams	NaN
298	307	SC	0	Mr.	Jay	NaN	Adams	NaN
299	309	SC	0	Mr.	Ronald	L.	Adina	NaN



In [17]: `person.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 19972 entries, 0 to 19971
Data columns (total 13 columns):
#   Column                Non-Null Count  Dtype
---  -
0   BusinessEntityID      19972 non-null  int64
1   PersonType            19972 non-null  object
2   NameStyle             19972 non-null  int64
3   Title                 1009 non-null   object
4   FirstName             19972 non-null  object
5   MiddleName            11473 non-null  object
6   LastName              19972 non-null  object
7   Suffix                53 non-null     object
8   EmailPromotion        19972 non-null  int64
9   AdditionalContactInfo 10 non-null     object
10  Demographics          19972 non-null  object
11  rowguid               19972 non-null  object
12  ModifiedDate          19972 non-null  object
dtypes: int64(3), object(10)
memory usage: 2.0+ MB
```

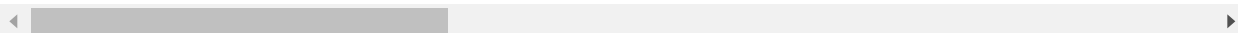
In [26]: `product = pd.read_csv("Production.Product.csv", sep = ";")`

In [27]: `product.head()`

Out[27]:

	ProductID	Name	ProductNumber	MakeFlag	FinishedGoodsFlag	Color	SafetyStockLevel	F
0	1	Adjustable Race	AR-5381	0	0	NaN	1000	
1	2	Bearing Ball	BA-8327	0	0	NaN	1000	
2	3	BB Ball Bearing	BE-2349	1	0	NaN	800	
3	4	Headset Ball Bearings	BE-2908	0	0	NaN	800	
4	316	Blade	BL-2036	1	0	NaN	800	

5 rows × 25 columns



In [28]: `product.columns`

Out[28]: Index(['ProductID', 'Name', 'ProductNumber', 'MakeFlag', 'FinishedGoodsFlag', 'Color', 'SafetyStockLevel', 'ReorderPoint', 'StandardCost', 'ListPrice', 'Size', 'SizeUnitMeasureCode', 'WeightUnitMeasureCode', 'Weight', 'DaysToManufacture', 'ProductLine', 'Class', 'Style', 'ProductSubcategoryID', 'ProductModelID', 'SellStartDate', 'SellEndDate', 'DiscontinuedDate', 'rowguid', 'ModifiedDate'], dtype='object')

In [123]: `product2 = product[["ProductID", "Name", "StandardCost", "DaysToManufacture"]]`
`product2.head()`

Out[123]:

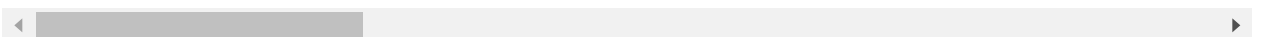
	ProductID	Name	StandardCost	DaysToManufacture
0	1	Adjustable Race	0.0	0
1	2	Bearing Ball	0.0	0
2	3	BB Ball Bearing	0.0	1
3	4	Headset Ball Bearings	0.0	0
4	316	Blade	0.0	1

In [147]: `product2.to_csv("01producto.csv", index=False)`

In [37]: `pd.set_option('display.max_columns', None)`
`product[product.Style.notnull()].head()`

Out[37]:

	ProductID	Name	ProductNumber	MakeFlag	FinishedGoodsFlag	Color	SafetyStockLevel
209	680	HL Road Frame - Black, 58	FR-R92B-58	1	1	Black	500
210	706	HL Road Frame - Red, 58	FR-R92R-58	1	1	Red	500
213	709	Mountain Bike Socks, M	SO-B909-M	0	1	White	4
214	710	Mountain Bike Socks, L	SO-B909-L	0	1	White	4
216	712	AWC Logo Cap	CA-1098	0	1	Multi	4



In [36]: `product.shape`

Out[36]: (504, 25)

```
In [38]: customer = pd.read_csv("Sales.Customer.csv", sep = ";")
```

```
In [39]: customer.head()
```

```
Out[39]:
```

	CustomerID	PersonID	StoreID	TerritoryID	AccountNumber	rowguid	ModifiedDate
0	1	NaN	934.0	1	AW00000001	3F5AE95E-B87D-4AED-95B4-C3797AFCB74F	2014-09-12 11:15:07.263
1	2	NaN	1028.0	1	AW00000002	E552F657-A9AF-4A7D-A645-C429D6E02491	2014-09-12 11:15:07.263
2	3	NaN	642.0	4	AW00000003	130774B1-DB21-4EF3-98C8-C104BCD6ED6D	2014-09-12 11:15:07.263
3	4	NaN	932.0	4	AW00000004	FF862851-1DAA-4044-BE7C-3E85583C054D	2014-09-12 11:15:07.263
4	5	NaN	1026.0	4	AW00000005	83905BDC-6F5E-4F71-B162-C98DA069F38A	2014-09-12 11:15:07.263

```
In [41]: customer[customer.PersonID.notnull()].head()
```

```
Out[41]:
```

	CustomerID	PersonID	StoreID	TerritoryID	AccountNumber	rowguid	ModifiedDate
701	11000	13531.0	NaN	9	AW00011000	477586B3-2977-4E54-B1A8-569AB2C7C4D4	2014-09-12 11:15:07.263
702	11001	5454.0	NaN	9	AW00011001	C32A8084-9077-4F13-9738-1E2DA7C1DCD9	2014-09-12 11:15:07.263
703	11002	11269.0	NaN	9	AW00011002	45715DD8-2F57-4A39-BEB4-6A8F99D59794	2014-09-12 11:15:07.263
704	11003	11358.0	NaN	9	AW00011003	7E240EFC-7EE6-4814-93A8-269821157E18	2014-09-12 11:15:07.263
705	11004	11901.0	NaN	9	AW00011004	61CCB4D0-2328-4BBB-AEF9-E7E0B0FDD67A	2014-09-12 11:15:07.263

```
In [42]: customer.columns
```

```
Out[42]: Index(['CustomerID', 'PersonID', 'StoreID', 'TerritoryID', 'AccountNumber',  
               'rowguid', 'ModifiedDate'],  
              dtype='object')
```

```
In [43]: base = pd.read_csv("datos_base_clientes.csv", sep = ",")
base.head()
```

Out[43]:

	documento	tipo_doc	categoria	mnt_trx_mm	num_trx	pct_mnt_tot	pct_num_tr
0	-9222147298886477023	1	COMIDA	0.05	7	1.000000	1
1	-9221406660220722252	1	COMIDA	0.25	2	0.050916	C
2	-9221406660220722252	1	OTROS	3.24	4	0.659878	C
3	-9221406660220722252	1	TRANSPORTE	0.34	4	0.069246	C
4	-9221406660220722252	1	HOGAR	1.08	6	0.219959	C

```
In [44]: detail = pd.read_csv("Sales.SalesOrderDetail.csv", sep = ";")
detail.head()
```

Out[44]:

sOrderID	SalesOrderDetailID	CarrierTrackingNumber	OrderQty	ProductID	SpecialOfferID	UnitPrice
43659	1	4911-403C-98	1	776	1	2024.994
43659	2	4911-403C-98	3	777	1	2024.994
43659	3	4911-403C-98	1	778	1	2024.994
43659	4	4911-403C-98	1	771	1	2039.994
43659	5	4911-403C-98	1	772	1	2039.994

```
In [45]: detail.columns
```

...

```
In [129]: detail2 = detail[["SalesOrderID", "OrderQty", "ProductID", "UnitPrice", "LineTotal"]]  
detail2.head()
```

Out[129]:

	SalesOrderID	OrderQty	ProductID	UnitPrice	LineTotal
0	43659	1	776	2024.994	2024.994
1	43659	3	777	2024.994	6074.982
2	43659	1	778	2024.994	2024.994
3	43659	1	771	2039.994	2039.994
4	43659	1	772	2039.994	2039.994

```
In [148]: detail2.to_csv("01dettalle.csv", index=False)
```

```
In [48]: detail.duplicated()
```

Out[48]:

0	False
1	False
2	False
3	False
4	False
...	
121312	False
121313	False
121314	False
121315	False
121316	False

Length: 121317, dtype: bool

```
In [49]: territory = pd.read_csv("Sales.SalesTerritory.csv", sep = ";")
territory.head()
```

Out[49]:

	TerritoryID	Name	CountryRegionCode	Group	SalesYTD	SalesLastYear	CostYTD	CostLastYear
0	1	Northwest	US	North America	7.887187e+06	3.298694e+06	0.0	0.0
1	2	Northeast	US	North America	2.402177e+06	3.607149e+06	0.0	0.0
2	3	Central	US	North America	3.072175e+06	3.205014e+06	0.0	0.0
3	4	Southwest	US	North America	1.051085e+07	5.366576e+06	0.0	0.0
4	5	Southeast	US	North America	2.538667e+06	3.925071e+06	0.0	0.0



```
In [50]: territory.columns
```

Out[50]: Index(['TerritoryID', 'Name', 'CountryRegionCode', 'Group', 'SalesYTD', 'SalesLastYear', 'CostYTD', 'CostLastYear', 'rowguid', 'ModifiedDate'], dtype='object')

```
In [131]: territory2 = territory[["TerritoryID", "Name", "Group"]]
territory2.head()
```

Out[131]:

	TerritoryID	Name	Group
0	1	Northwest	North America
1	2	Northeast	North America
2	3	Central	North America
3	4	Southwest	North America
4	5	Southeast	North America

```
In [149]: territory2.to_csv("01territory.csv", index=False)
```


In [51]: `territory.SalesYTD`

Out[51]:

0	7.887187e+06
1	2.402177e+06
2	3.072175e+06
3	1.051085e+07
4	2.538667e+06
5	6.771829e+06
6	4.772398e+06
7	3.805202e+06
8	5.977815e+06
9	5.012905e+06

Name: SalesYTD, dtype: float64

In [53]: `sec = pd.read_csv("Secuenciales.csv", sep = ",")`
`sec.head()`

Out[53]:

	secuencial	codigo
0	0	3606
1	1	3615
2	2	3607
3	3	3603
4	4	3608

In [136]: `sales = pd.read_csv("Sales.SalesOrderHeader.csv", sep = ";")`
`sales.head()`

Out[136]:

OrderID	AccountNumber	CustomerID	SalesPersonID	TerritoryID	BillToAddressID	ShipToAddressID	ShipMethodID
37	10-4020-000676	29825	279.0	5	985	985	
30	10-4020-000117	29672	279.0	5	921	921	
20	10-4020-000442	29734	282.0	6	517	517	
44	10-4020-000227	29994	282.0	6	482	482	
70	10-4020-000510	29565	276.0	4	1073	1073	

```
In [162]: sales2 = sales[["SalesOrderID", "OrderDate", "CustomerID", "TotalDue", "SalesPersonID", "TerritoryID"]]
sales2["OrderDate"] = pd.to_datetime(sales2['OrderDate'].astype(str), format='%Y-%m-%d')
```

C:\Users\angel\anaconda3\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
In [163]: sales2.head()
```

Out[163]:

	SalesOrderID	OrderDate	CustomerID	TotalDue	SalesPersonID	TerritoryID
0	43659	2011-05-31	29825	23153.2339	279.0	5
1	43660	2011-05-31	29672	1457.3288	279.0	5
2	43661	2011-05-31	29734	36865.8012	282.0	6
3	43662	2011-05-31	29994	32474.9324	282.0	6
4	43663	2011-05-31	29565	472.3108	276.0	4

```
In [166]: sales2.groupby("OrderDate").sum()
```

Out[166]:

	SalesOrderID	CustomerID	TotalDue	SalesPersonID	TerritoryID
OrderDate					
2011-05-31	1878240	1232178	567020.9498	10596.0	173
2011-06-01	174814	66285	15394.3298	0.0	31
2011-06-02	218540	108147	16588.4572	0.0	36
2011-06-03	87423	41119	7907.9768	0.0	9
2011-06-04	218575	99453	16588.4572	0.0	41
...
2014-06-26	2174507	568392	1660.6501	0.0	174
2014-06-27	2400432	628627	1931.1761	0.0	207
2014-06-28	2326395	564428	2041.4440	0.0	190
2014-06-29	1726656	444102	1632.7596	0.0	116
2014-06-30	3004140	719521	2921.1901	0.0	234

1124 rows × 5 columns

```
In [142]: sales2.isna().any()
```

```
Out[142]: SalesOrderID    False
          OrderDate       False
          CustomerID      False
          SalesPersonID   False
          TerritoryID     False
          TotalDue        False
          dtype: bool
```

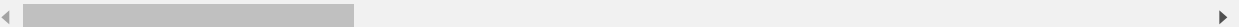
```
In [164]: sales2.to_csv("01sales.csv",index=False)
```

```
In [56]: sales.columns
```

```
Out[56]: Index(['SalesOrderID', 'RevisionNumber', 'OrderDate', 'DueDate', 'ShipDate',
               'Status', 'OnlineOrderFlag', 'SalesOrderNumber', 'PurchaseOrderNumber',
               'AccountNumber', 'CustomerID', 'SalesPersonID', 'TerritoryID',
               'BillToAddressID', 'ShipToAddressID', 'ShipMethodID', 'CreditCardID',
               'CreditCardApprovalCode', 'CurrencyRateID', 'SubTotal', 'TaxAmt',
               'Freight', 'TotalDue', 'Comment', 'ModifiedDate'],
              dtype='object')
```

```
In [58]: sales[sales.Comment.notnull()].head()
```

```
Out[58]:
```

SalesOrderID	RevisionNumber	OrderDate	DueDate	ShipDate	Status	OnlineOrderFlag	SalesOrd
							

```
In [59]: sales.shape
```

```
Out[59]: (31465, 25)
```

```
In [64]: sales.SalesPersonID.isna().sum()
```

```
Out[64]: 27659
```

```
In [69]: sales[sales["Status"] == 5].shape
```

```
Out[69]: (31195, 25)
```

```
In [73]: merge = pd.merge(sales, territory, on="TerritoryID")
         merge.columns
```

```
Out[73]: Index(['SalesOrderID', 'RevisionNumber', 'OrderDate', 'DueDate', 'ShipDate',
               'Status', 'OnlineOrderFlag', 'SalesOrderNumber', 'PurchaseOrderNumber',
               'AccountNumber', 'CustomerID', 'SalesPersonID', 'TerritoryID',
               'BillToAddressID', 'ShipToAddressID', 'ShipMethodID', 'CreditCardID',
               'CreditCardApprovalCode', 'CurrencyRateID', 'SubTotal', 'TaxAmt',
               'Freight', 'TotalDue', 'Comment', 'ModifiedDate_x', 'Name',
               'CountryRegionCode', 'Group', 'SalesYTD', 'SalesLastYear', 'CostYTD',
               'CostLastYear', 'rowguid', 'ModifiedDate_y'],
              dtype='object')
```

```
In [172]: a = merge.groupby("Group").sum()  
a = a["TotalDue"]/1000  
a
```

```
Out[172]: Group  
Europe      22173.617630  
North America  89228.792391  
Pacific      11814.376095  
Name: TotalDue, dtype: float64
```

```
In [174]: a[0]
```

```
Out[174]: 22173.617629699667
```

```
In [ ]:
```