In [ ]:

```
import pandas as pd
import numpy as np
import requests
import plotly.express as px
```

Function definitions

In [ ]:

```
def preparationOfDataframesWithSerie(serie):
    api_url = "http://127.0.0.1:3000/men_and_women_in_government?series_id_women=fts."+serie
    response = requests.get(api_url)
    db_json = response.json()
    df = pd.json_normalize(db_json)
    return df
```

In [ ]:

```
def preparationOfDataframesFull(table):
    api_url = "http://127.0.0.1:3000/"+table
    response = requests.get(api_url)
    db_json = response.json()
    df = pd.json_normalize(db_json)
    return df
```

In [ ]:

```
def dataCharConstructionProduction(df,title_def):
    fig = px.line(df_full, x=df["year"], y=df['series_id_prod'],hover_data={"year": "|%B %d, %Y"}, title=title_def)
    fig.update_xaxes(dtick="M1",tickformat="%b\n%Y")
    fig.show()
```

In [ ]:

```
def dataCharConstructorMonth(df_full,title_def):
    df_par = df_full.copy()
    del df_par['date']
    fig = px.line(df_full, x="date", y=df_par.columns,hover_data={"date": "|%B %d, %Y"}, title=title_def)
    fig.update_xaxes(dtick="M1",tickformat="%b\n%Y")
    fig.show()
```

In [ ]:

```
def dataCharConstructorYear(df_full,title_def):
    df_par = df_full.copy()
    del df_par['year']
    fig = px.line(df_full, x="year", y=df_par.columns,hover_data={"year": "|%B %d, %Y"}, title=title_def)
    fig.update_xaxes(dtick="M1",tickformat="%b\n%Y")
    fig.show()
```

1. Extractions for Women in government employment

The objective of this first practical section is the preparation of the info extracted from the U.S. Bureau of Labor Statistics (BLS) for presenting a brief study about the question **How was the evolution of women in goverment during time?**

As an important note, series are differenciated on seasonally adjusted and no seasonally adjusted. The first presents less variation due to the extraction of calendar effects and other data outliers. For this experiment it will be used the seasonally adjusted data (CES series).

1.1 Views constructed for the analysis

1.1.1 Total government employment tables and charts

For the main analysis, table **men_and_women_in_government** was prepared for concentrating the principal information related to the year performance of the series related to the women. Men employment data were infered using the total employment data minus women data.

In [ ]:

```
df_full_tot = preparationOfDataframesFull("men_and_women_in_government")
df_full_tot.head()
```

Out[ ]:

	series_id_tot	series_id_women	year	avg_value_tot	title	avg_value_women	avg_value_men	women_gov_ratio	men_gov_ratic
0	CES9000000001	CES9000000010	1964	9712.750000	Women employees, thousands, government, season...	3771.333333	5941.416667	0.388287	0.611713
1	CES9000000001	CES9000000010	1965	10193.083333	Women employees, thousands, government, season...	4023.833333	6169.250000	0.394761	0.605239
2	CES9000000001	CES9000000010	1966	10912.916667	Women employees, thousands, government, season...	4434.000000	6478.916667	0.406308	0.593692
3	CES9000000001	CES9000000010	1967	11528.083333	Women employees, thousands, government, season...	4765.666667	6762.416667	0.413396	0.586604
4	CES9000000001	CES9000000010	1968	11975.416667	Women employees, thousands, government, season...	5042.500000	6932.916667	0.421071	0.578929

Table **all_amount_total_in_government** contains the anual value for each series. This table allows to construct the chart with the behavior of each serie from 1964 to 2023, with the exceptions of the series "CES9092299910","CES9093000010","CES9093161110", "CES9093200010","CES9093222110","CES9093248010","CES9093262210","CES9093292010" and "CES9093299910" which information begins from 1972.

In [ ]:
```
df_full_tot = preparationOfDataframesFull("all_amount_total_in_government")
dataCharConstructorYear(df_full_tot,"Total of employments during time")
```

1.1.2 Men government employment tables and charts

As it was commented in paragraph 1.1.1, men government employment information was infered using the total employment series minus the info of the women government employment series. This information allowed to prepare the parallel info for a better contrast. Table all_amount_men_in_government contains the info related to the series id of women statistics.

In [ ]:
```
df_full = preparationOfDataframesFull("all_amount_men_in_government")
dataCharConstructorYear(df_full,"Total of men employment during time")
```

1.1.3 Women government employment tables and charts

Table **all_amount_women_in_government** presents the info related to the women employment in government. Table and chart results to be interesting because it shows how there is an interesting delta that will be commented in brief.

In [ ]:
```
df_full = preparationOfDataframesFull("all_amount_women_in_government")
dataCharConstructorYear(df_full,"Total of women employment during time")
```

For a better comparative between the status of the men employment and women employment for each serie, it was prepared the table **all_percent_women_in_government** where the women government employment ratio was calculated by the relation (division) of the sum_value_women (women employment data for the year) and sum_value_tot (total employment data for the year).

In [ ]:
```
df_full = preparationOfDataframesFull("all_percent_women_in_government")
dataCharConstructorYear(df_full,"Ratios of women employment during time")
```

1.2 Brief interpretation of the numbers

1.2.1 Serie CES9000000001 (All employees, thousands, government, seasonally adjusted)

Serie CES9000000001 covers only civilian employees in government. This serie presents the best relation quantity-ratio. During the years of the analyisis, just women employment in federal hospitals presents a better ratio (principally after 1976). In 1984 ratio crossed the 0.5 value, which means that since that year there are more women civilian employees than men.

1.2.2 Serie CES9091000010 (Women employees, thousands, federal, seasonally adjusted)

Serie CES9091000010 presents women on federal employments. Being approximately 1M employees per year, ratio shows a fast advance in the women employment (from 0.22 in 1964 to 0.43 in 1990), but ratio is still moving between 0.40 and 0.45.

1.2.3 Serie CES9091100010 (Women employees, thousands, federal, except u.s. postal service, seasonally adjusted)

This serie presents a very similar behavior than the previous one (because the difference of the extraction of the postal service).

1.2.4 Serie CES9091622010 (Women employees, thousands, federal hospitals, seasonally adjusted)

With a very low employment amount (200K), women employment has the best ratio of all the series since 1977 (0.63 in 2022)

1.2.5 Serie CES9091911010 (Women employees, thousands, department of defense, seasonally adjusted)

From almost 3.7M of total employment in 1971 and a ratio of 0.49 of women employment, department of defense shows an important decrement of employment (approximately 200K per year during the las 40 years) and a low ratio for woman (between 0.3 and 0.35)

1.2.6 Serie CES9091912010 (Women employees, thousands, u.s. postal service, seasonally adjusted)

```
In 1966 presented a ratio of 0.62 of women employment but in 1976 hit the lower ratio (0.28).  During the
last years ratio is about 0.45 for an employment of 300K women.
```

1.2.7 Serie CES9091999910 (Women employees, thousands, other federal government, seasonally adjusted)

```
This "other federal government" presents a very similar behavior than u.s. postal service.
```

1.2.8 Serie CES9092000010 (Women employees, thousands, state government, seasonally adjusted)

```
With a very regular base (1.5M on total employment in 1976 increasing each year to 2.7M in 2022), woman
employment ratio got bigger than 0.5 since 1995.
```

1.2.9 Serie CES9092161110 (Women employees, thousands, state government education, seasonally adjusted)

```
With a regular increment of total employment from 16.4M in 1976 to 29.3M in 2022, the women employment
ratio present a regular growth (0.33 in 1972 to 0.56 in 2022).
```

1.2.10 Serie CES9092200010 (Women employees, thousands, state government, excluding education, seasonally adjusted)

```
With a line of employment around 1M, ratio is very similar to the previous serie.
```

1.2.11 Serie CES9092262210 (Women employees, thousands, state hospitals, seasonally adjusted)

```
Even when there is an important decrement of total employment from 4.3M in 1975 to 325K in 1976), the
women employment ratio has a regular increment year by year (from 0.38 in 1964 to 0.58 in 2022), crossing
the 0.5 in 1984.
```

1.2.12 Serie CES9092292010 (Women employees, thousands, state government general administration, seasonally adjusted)

A very interesting behavior. In 1966 women employment ratio (0.62) start falling to 0.40 in 1976. However, it kept falling to 0.34 in 1983. After this year ratio has moved to almost 0.49 in 2022.

1.2.13 Serie CES9092299910 (Women employees, thousands, other state government, seasonally adjusted)

With a regular 250K employees since 1976, women employment ratio has mantained in a band of 0.45 to 0.55 since 1987.

1.2.14 Serie CES9093000010 (Women employees, thousands, local government, seasonally adjusted)

Since 1976, local government has gone from 4.4M to 8.9M in 2022. However, ratio presents a very interesting behavior: since 1977 percentage is bigger than 0.5 but in some point of the 1972-1975 women ratio was regularly in 0.76M.

1.2.15 Serie CES9093161110 (Women employees, thousands, local government education, seasonally adjusted)

Local education present a 2.8M-5.8M increase from 1976 to 2022. However, the real jump is the women employment ratio, which in the same interval it moved from 0.44 to 0.74.

1.2.16 Serie CES9093200010 (Women employees, thousands, local government, excluding education, seasonally adjusted)

Local government present a poor ratio of women employment, due to value went from 0.27 in 1972 to 0.47 in 2022: it result a low percentage if we compare it with the local government ratio.

1.2.17 Serie CES9093222110 (Women employees, thousands, local government utilities, seasonally adjusted)

```
Government utilities present a continuous decrement of employment (from 195K in 1972 to 60K in 2022).
However the interesting part is that from 1976 to 1989 the ratio was around 0.8.  After that, ratio is
around 0.2.
```

1.2.18 Serie CES9093248010 (Women employees, thousands, local government transportation, seasonally adjusted)

```
With a goden age from 1976 to 1990 (from 0.44 to 0.53), ratio jumped 0.22 in 1990. Recuperation in 2022 is
about 0.30 but with just 86K employees.
```

1.2.19 Serie CES9093262210 (Women employees, thousands, local hospitals, seasonally adjusted)

```
In 1990, local hospitals total employment had fallen from 1.3M to 532K.  However, in the same year the
women employment ratio went from 0.42 to 0.82 with an 0.78 in 2023.
```

1.2.20 Serie CES9093292010 (Women employees, thousands, local government general administration, seasonally adjusted)

```
Local government general administration presented a slow increment from 495K in 1972 to 1.7M in 2022.
Ratio has changed during years but it comes back to 0.4 every time.
```

1.2.21 Serie CES9093299910 (Women employees, thousands, other local government, seasonally adjusted)

```
This series present a regular ratio bigger than 0.5 since 1977 with an average of 500K employees.
```

1.3 Last comments of the Woman Government Employment

As a resumen, numbers show a solid growth in the labor force participation for the last 50 years. There is an important number of government employments where women now are mayority and almost every activity shows an important increment of women as employee.

```
- Years 1976 and 1990 presented an important modification in the numbers; it would be interesting to
verify the possible situations that would affect the behavior.
- There are very specific areas in government where women has an important impact (like local hospitals),
but there still other areas where it seems that the door is still closed (for example, department of
defense).
- An interesting case is the local government utilities.  In 1976 ratio jumped to 0.76 and mantain that
level until 1990, when disminshed to 0.19 maintaining similar ratio during the next years.  It should be
interesting to detect the motivation of this drastic changes in the behavior of some departments.
```

. . .

. .

2. Evolution of the ratio production employees / supervisory employees

The objective of this second practical section is the preparation of the info extracted from the U.S. Bureau of Labor Statistics (BLS) for presenting a brief analysis about the question **How was the evolution of the ratio "production employees / supervisory employees" during time?**

As an important note, with the same idea mencioned in the first section, series are differenciated on seasonally adjusted and no seasonally adjusted. The first presents less variation due to the extraction of calendar effects and other data outliers. For this experiment it will be used the seasonally adjusted data (CES series).

2.1 Views constructed for the analysis

The construction of the views and tables were different for the previous analysis due to the arrange of the information. For this case the interpretation of the info will be presented according to the serie, defining the contrast among the variations of each case.

2.2 Calculation of the ratios

Due to the database only presents the information of the production employees and the total employment of each serie, it is necessary to infere the information of the supervisory employees. Verifying some other pages of the BLS it recomments the construction of the supervisory employees as the residual of the total once the production employees are extracted. However, in some areas the number could not be efficient due to some of the residuals could be related to sales or other departments related. According to this, supervisory employees will be calculated as

```
supervisory_employees = total_employees - production_employees
```

As a result of this definition, the production-supervisory ratio was calculated as

```
prod_superv_ratio = (production_employees / supervisory_employees) =  (production_employees /
(total_employees - production_employees))
```

As a complementary coefficient, it was calculated a second ratio related to the production employees and total employees. This ratio was calculated as

```
prod_vs_total_ratio = production_employees / total_employees
```

which results in an acotation between 0 and 1 of the previous ratio.

2.2.1 Total private employment

First case is he total private employment case. This serie present the total private employment (130M average in 2022) composed by the goods-producing (21M average in 2022) and the private service providing (109M average in 2022).

In [ ]:

```
df = preparationOfDataframesFull('d05_total_private_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_t
58	CES0500000006	CES0500000001	2022	130447.5	106259.833333	4.393141	0.814579	Production and nonsupervisory employees, thous...	All emp tho total
135	CES0600000006	CES0600000001	2022	21181.5	15185.166667	2.532409	0.716907	Production and nonsupervisory employees, thous...	All emp tho pro
195	CES0800000006	CES0800000001	2022	109266.0	91074.666667	5.006487	0.833513	Production and nonsupervisory employees, thous...	All emp tho private

In [ ]:

```
df = preparationOfDataframesFull('cross_d05_total_private_employment')
dataCharConstructorYear(df,"Ratios of total_private_employment")
```

Previous chart shows a 5:1 ratio on private service-providing versus a 2.53:1 ratio on goods-producing employees. Due to the amount of both cases, generality (total private) shows a 4.4:1 ratio. However, ratios in 1964 were 5:1 for total private; private service-providing had a 6.25:1 and private service-providing presented 3.8:1 ratio.

2.2.2 Mining and logging employment

In [ ]:

```
df = preparationOfDataframesFull('d10_miningandlogging_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_t
279	CES1021200006	CES1021200001	2022	182.950000	142.458333	3.518214	0.778674	Production and nonsupervisory employees, thous...	All emp tho mining
313	CES1021230006	CES1021230001	2022	99.633333	74.733333	3.001339	0.750084	Production and nonsupervisory employees, thous...	All emp tho non m
347	CES1021231006	CES1021231001	2022	43.650000	32.475000	2.906040	0.743986	Production and nonsupervisory employees, thous...	All emp tho stone ar
381	CES1021300006	CES1021300001	2022	260.058333	201.533333	3.443543	0.774954	Production and nonsupervisory employees, thous...	All emp tho acti
415	CES1021311206	CES1021311201	2022	200.183333	159.283333	3.894458	0.795687	Production and nonsupervisory employees, thous...	All emp tho acti

In [ ]:

```
df = preparationOfDataframesFull('cross_d10_miningandlogging_employment')
dataCharConstructorYear(df,"Ratios of mining_and_logging_employment")
```

Since 1972, oil and gas extraction shows a 1:1 with a size of 72K employees in 2022. Similar behaviors are mining quarring and mining and logging (ratio of 3:1 but sizes of 417K and 455K respectively).
With a caotic behavior jumping between 5:1 and 2.5:1 ratios there are support of activities (mining and oil and gas operations), stone mining and quarrying, non minerals and mining except oil and gas. The case of logging is interesting due to the small size of employees and the variation of the ratio during the years. It is better to see in the previous chart (case CES101133006).

2.2.3 Construction employment

In [ ]:

```
df = preparationOfDataframesFull('d20_construction_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_
1143	CES2023835006	CES2023835001	2022	163.275000	110.900000	2.117422	0.679222	Production and nonsupervisory employees, thous...	All em th carper
1177	CES2023839006	CES2023839001	2022	88.650000	66.008333	2.915348	0.744595	Production and nonsupervisory employees, thous...	All em th other
1211	CES2023890006	CES2023890001	2022	746.458333	588.691667	3.731407	0.788646	Production and nonsupervisory employees, thous...	All em th other
1245	CES2023891006	CES2023891001	2022	379.666667	306.800000	4.210430	0.808077	Production and nonsupervisory employees, thous...	All em thous pre
1279	CES2023899006	CES2023899001	2022	366.800000	282.016667	3.326322	0.768857	Production and nonsupervisory employees, thous...	All em thous other

In [ ]:

```
df = preparationOfDataframesFull('cross_d20_construction_employment')
dataCharConstructorYear(df,"Ratios of construction_employment")
```

For this case, the behavior of the ratio is similar for almost all the series. Series with differences are framing_contractors (CES2023813006) which ratio has jumped to 11:1 in 1994 and 2005; masonry contractors (CES2023814006) and oil and gas pipelines (CES2023712006) which ratio has been running in 8:1 in different years. The rest of the cases are related with ratios between 6:1 and 2:1.

2.2.4 Manufacturing employment

In [ ]:

```
df = preparationOfDataframesFull('d30_manufacturing_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_tit
83	CES3000000006	CES3000000001	2022	12827.25	9003.833333	2.354918	0.70193	Production and nonsupervisory employees, thous...	All empl thou manufac s

In [ ]:

```
df = preparationOfDataframesFull('cross_d30_manufacturing_employment')
dataCharConstructorYear(df,"Ratios of manufacturing_employment")
```

With a size of 9M in 2022, manufacturing employment shows an important decrement during the last 80 years. In 1943 ratio was on 8.2:1, falling to 2.3:1 in 2022.

2.2.5 Trade transportation and utilities employment

In [ ]:

```
df = preparationOfDataframesFull('d40_trade_transportation_and_utilities_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_tit
58	CES4000000006	CES4000000001	2022	28663.916667	24181.666667	5.394984	0.843627	Production and nonsupervisory employees, thous...	All empl thou transpor

In [ ]:

```
df = preparationOfDataframesFull('cross_d40_trade_transportation_and_utilities_employment')
dataCharConstructorYear(df,"Ratios of trade_transportation_and_utilities_employment")
```

Trade transportation shows a brief decrement from 1964 (7.5:1) to 1992 (5.1:1). After this year it has presented a recuperation (2004 started the improvement) to present a 5.4:1 in 2022. It is important to measure that employment size in 2022 is more than 24M.

2.2.6 Wholesale trade employment

In [ ]:

```
df = preparationOfDataframesFull('d41_wholesale_trade_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_
1316	CES4142470006	CES4142470001	2022	99.025000	80.916667	4.468477	0.817134	Production and nonsupervisory employees, thous...	All em th petrol
1350	CES4142480006	CES4142480001	2022	205.475000	170.425000	4.862340	0.829420	Production and nonsupervisory employees, thous...	All em th beer, v
1384	CES4142490006	CES4142490001	2022	327.241667	262.725000	4.072204	0.802847	Production and nonsupervisory employees, thous...	All em th misce
1418	CES4142491006	CES4142491001	2022	117.441667	92.925000	3.790279	0.791244	Production and nonsupervisory employees, thous...	All em th farm
1452	CES4142500006	CES4142500001	2022	516.733333	397.491667	3.333496	0.769239	Production and nonsupervisory employees, thous...	All em th w trac

In [ ]:

```
df = preparationOfDataframesFull('cross_d41_wholesale_trade_employment')
dataCharConstructorYear(df,"Ratios of wholesale_trade_employment")
```

Wholesale trade employment shows a very static ratio close to 5:1 for almost all the series since 2000. Previous to this year just sporting wholesalers present a jump to the 24.4:1 in 1997; the other two cases correspond to computer wholesalers and electrical equipment wholesalers stayed in 10:1 in different years. Series present low-size of employment in almost all the cases.

2.2.7 Retail trade employment

In [ ]:

```
df = preparationOfDataframesFull('d42_retail_trade_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_
1736	CES4245942006	CES4245942001	2022	134.733333	107.041667	3.865483	0.794471	Production and nonsupervisory employees, thous...	All em th gift
1770	CES4245950006	CES4245950001	2022	194.250000	163.991667	5.419719	0.844230	Production and nonsupervisory employees, thous...	All em th mer
1804	CES4245990006	CES4245990001	2022	483.108333	374.633333	3.453638	0.775464	Production and nonsupervisory employees, thous...	All em th misce
1838	CES4245991006	CES4245991001	2022	133.191667	105.308333	3.776748	0.790653	Production and nonsupervisory employees, thous...	All em thous s
1872	CES4245999106	CES4245999101	2022	107.950000	76.833333	2.469202	0.711749	Production and nonsupervisory employees, thous...	All em th ele

In [ ]:

```
df = preparationOfDataframesFull('cross_d42_retail_trade_employment')
dataCharConstructorYear(df,"Ratios of retail_trade_employment")
```

Very similar to Wholesale trade employment, this case present also a 5:1 ratio but with a bigger dispersion in 1990 and 2022 (small dispersion in 2006, at the middle). Also employment size is low.

2.2.8 Transportation warehousing utilities employment

In [ ]:

```
df = preparationOfDataframesFull('d43_transportation_warehousing_utilities_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_t
834	CES4422111006	CES4422111001	2022	141.791667	108.908333	3.311961	0.768087	Production and nonsupervisory employees, thous...	All emp tho electri
868	CES4422111206	CES4422111201	2022	74.241667	57.975000	3.564037	0.780896	Production and nonsupervisory employees, thous...	All emp tho fo e
902	CES4422112006	CES4422112001	2022	242.450000	199.800000	4.684642	0.824087	Production and nonsupervisory employees, thous...	All emp tho electri
936	CES4422120006	CES4422120001	2022	112.816667	90.766667	4.116402	0.804550	Production and nonsupervisory employees, thous...	All emp tho natu d
970	CES4422130006	CES4422130001	2022	56.500000	41.941667	2.880939	0.742330	Production and nonsupervisory employees, thous...	All emp tho water,

In [ ]:

```
df = preparationOfDataframesFull('cross_d43_transportation_warehousing_utilities_employment')
dataCharConstructorYear(df,"Ratios of transportation_warehousing_utilities_employment")
```

Similar to the previous four series, transportation warehousing utilities shows a tendency to the 5:1 in most of the cases. Principal deviations could be 2.5:1 and 7.5:1 ratios. School and employee bus transportation is always close to 15:1 and transit and ground passenger transportation does the same in 10:1 ratio.

2.2.9 Information employment

In [ ]:

```
df = preparationOfDataframesFull('d50_information_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	ser
534	CES5051711206	CES5051711201	2022	93.133333	75.891667	4.401643	0.814871	Production and nonsupervisory employees, thous...	Al thousa te
568	CES5051712006	CES5051712001	2022	42.233333	34.625000	4.550931	0.819850	Production and nonsupervisory employees, thous...	Al telecom
602	CES5051780006	CES5051780001	2022	52.708333	44.125000	5.140777	0.837154	Production and nonsupervisory employees, thous...	Al thousar te
636	CES5051800006	CES5051800001	2022	465.716667	371.233333	3.929088	0.797123	Production and nonsupervisory employees, thous...	Al
670	CES5051900006	CES5051900001	2022	189.250000	152.466667	4.144993	0.805636	Production and nonsupervisory employees, thous...	Al tho sear

In [ ]:

```
df = preparationOfDataframesFull('cross_d50_information_employment')
dataCharConstructorYear(df,"Ratios of information_employment")
```

With a very contained ratio, information employment present a ratio between 1:1 and 4:1 in 1990 (with two exceptions) and finishes in 2022 with a range of 2.7:1 to 5.5:1 in 2022. Low-size of employment per serie.

2.2.10 Financial activities employment

In [ ]:

```
df = preparationOfDataframesFull('d55_financial_activities_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_
1232	CES5553200006	CES5553200001	2022	530.516667	427.175000	4.133618	0.805206	Production and nonsupervisory employees, thous...	All em th re le
1266	CES5553228306	CES5553228301	2022	34.000000	26.825000	3.738676	0.788971	Production and nonsupervisory employees, thous...	All em th hon eq
1300	CES5553240006	CES5553240001	2022	172.058333	135.333333	3.685047	0.786555	Production and nonsupervisory employees, thous...	All em th co an
1334	CES5553241006	CES5553241001	2022	92.225000	73.758333	3.994134	0.799765	Production and nonsupervisory employees, thous...	All em th cons
1368	CES5553249006	CES5553249001	2022	79.666667	61.625000	3.415704	0.773536	Production and nonsupervisory employees, thous...	All em th comn

In [ ]:

```python
df = preparationOfDataframesFull('cross_d55_financial_activities_employment')
dataCharConstructorYear(df,"Ratios of financial_activities_employment")
```

With the exception of serie CES5552229106 (consumer lending) that achieved a 22.8:1 ratio in 1991 and then returned to the similar behavior in 1998, ratio of this series can be assumed in 4:1.

2.2.11 Professional business services employment

In [ ]:

```python
df = preparationOfDataframesFull('d55_financial_activities_employment')
df = df.query('year == 2022')
```

```
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_
1232	CES5553200006	CES5553200001	2022	530.516667	427.175000	4.133618	0.805206	Production and nonsupervisory employees, thous...	All em th re le
1266	CES5553228306	CES5553228301	2022	34.000000	26.825000	3.738676	0.788971	Production and nonsupervisory employees, thous...	All em th hon eq
1300	CES5553240006	CES5553240001	2022	172.058333	135.333333	3.685047	0.786555	Production and nonsupervisory employees, thous...	All em th co an
1334	CES5553241006	CES5553241001	2022	92.225000	73.758333	3.994134	0.799765	Production and nonsupervisory employees, thous...	All em th cons
1368	CES5553249006	CES5553249001	2022	79.666667	61.625000	3.415704	0.773536	Production and nonsupervisory employees, thous...	All em th comn

In [ ]:

```
df = preparationOfDataframesFull('cross_d60_professional_business_services_employment')
dataCharConstructorYear(df,"Ratios of professional_business_services_employment")
```

Business services present a 5:1 as the previous cases, with the exceptions of temporary help services, employment services and armored car services. This cases present a behavior outside the group.

2.2.12 Education and healthcare employment

In [ ]:

```
df = preparationOfDataframesFull('d65_education_and_healthcare_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_
1388	CES6562412006	CES6562412001	2022	2152.033333	1951.241667	9.717742	0.906697	Production and nonsupervisory employees, thous...	All em th service
1422	CES6562419006	CES6562419001	2022	467.791667	377.625000	4.188078	0.807250	Production and nonsupervisory employees, thous...	All em th i
1456	CES6562420006	CES6562420001	2022	202.866667	156.316667	3.358038	0.770539	Production and nonsupervisory employees, thous...	All em th co foo
1508	CES6562430006	CES6562430001	2022	270.483333	230.516667	5.767723	0.852240	Production and nonsupervisory employees, thous...	All em th v r
1542	CES6562440006	CES6562440001	2022	957.525000	830.141667	6.516878	0.866966	Production and nonsupervisory employees, thous...	All em th c s

In [ ]:

```
df = preparationOfDataframesFull('cross_d65_education_and_healthcare_employment')
dataCharConstructorYear(df,"Ratios of education_and_healthcare_employment")
```

Education and healthcare show a very compact behavior since 1990, maintaining all the series among the ratios 2.5:1 and 12.5:5. It is important to say that the size of employees is big according to the previous cases.

2.2.13 Leisure and hospitality employment

```
In [ ]: df = preparationOfDataframesFull('d70_leisure_and_hospitality_employment')
        df = df.query('year == 2022')
        df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_
1030	CES7072250006	CES7072250001	2022	10672.758333	9404.466667	7.415066	0.881166	Production and nonsupervisory employees, thous...	All em th res ar
1064	CES7072251106	CES7072251101	2022	5189.725000	4643.641667	8.503540	0.894776	Production and nonsupervisory employees, thous...	All em th fu
1098	CES7072251306	CES7072251301	2022	4499.600000	3919.850000	6.761276	0.871155	Production and nonsupervisory employees, thous...	All em th limite
1132	CES7072251406	CES7072251401	2022	70.791667	61.000000	6.229787	0.861683	Production and nonsupervisory employees, thous...	All em th cafete
1166	CES7072251506	CES7072251501	2022	912.033333	778.850000	5.847954	0.853971	Production and nonsupervisory employees, thous...	All em th s non

```
In [ ]: df = preparationOfDataframesFull('cross_d70_leisure_and_hospitality_employment')
        dataCharConstructorYear(df,"Ratios of leisure_and_hospitality_employment")
```

Also with a compact behavior between 10:1 and 3:1, leisure and hospitality employment average is close to 5:1 with the exception of the serie CES7072251406 related to cafeterias, grills and buffets.

2.2.14 Other services employment

In [ ]:

```
df = preparationOfDataframesFull('d80_other_services_employment')
df = df.query('year == 2022')
df.tail()
```

Out[ ]:

	series_id_prod	series_id_tot	year	avg_value_tot	avg_value_prod	prod_superv_ratio	prod_vs_total_ratio	series_title_prod	series_
1078	CES8081390006	CES8081390001	2022	457.791667	351.991667	3.326953	0.768891	Production and nonsupervisory employees, thous...	All em th prc
1120	CES8081391006	CES8081391001	2022	116.416667	82.966667	2.480319	0.712670	Production and nonsupervisory employees, thous...	All em th assc
1172	CES8081392006	CES8081392001	2022	85.433333	59.083333	2.242252	0.691572	Production and nonsupervisory employees, thous...	All em th pro
1214	CES8081393006	CES8081393001	2022	105.516667	85.425000	4.251763	0.809588	Production and nonsupervisory employees, thous...	All em th labc a
1248	CES8081399006	CES8081399001	2022	150.533333	124.583333	4.800899	0.827613	Production and nonsupervisory employees, thous...	All em th pol

In [ ]:

```
df = preparationOfDataframesFull('cross_d80_other_services_employment')
dataCharConstructorYear(df,"Ratios of other_services_employment")
```

Other services maintain a compact case according to the numbers: an interval of 8:1 to 2:1 but with a tendency of 4:1 as average.

2.3 Last comments of the Production Employees/ Supervisory Employees

According to the historical information there are some basic comments about this ratio:

Series with longer inormation shows a bigger ratio in early years, principally when activity is not related to sales or administrative employment. After 1990 behaviors of each of the supersectors trend to converge to a specific ratio (which use to be close to 5:1). Goods production employment and manufacturing employment trend to have a smaller ratio (aprox 2.5:1) Even when inside each supersector there are important differences in the size (amount of employees), there are a significative simetry when we see their ratios. It could be related to the general procedures of each activity.

3. Conclusion

Thanks for the opportunity to apply. These exercise has been interesting for me due to my new understanding of the employment in US. Always interesting the preparation for data. In this case there were no deeper statistical analysis due to the focus in Postgres and Postgrest. I tried no to use python more than presentation of the charts and maybe a brief glimplse of the dataframes.

Due to the Data Analyst Challenge 2022 mention the use of the posgrest as defined in the link proposed, I prepared the docker containers to run the procedure (.yml included to verify the instalation).

I will aggregate the images in png in case that ipynb would not display the charts.

Also I will include the queries used to prepare the views used for the analysis.

Thanks.

Carlos

.

4. Queries used for construction

```
--CREATE VIEW woman_in_government AS with step1 as( SELECT g.series_id AS series_id, g.year AS year, g.period AS period, g.value AS value, s.series_title AS title FROM public."ce_data_90a_Government_Employment" g LEFT JOIN PUBLIC.ce_series s ON g.series_id = TRIM(s.series_id) where right(g.series_id,2) = '10' GROUP BY g.year, g.period, g.series_id, g.value, s.series_title ORDER BY g.series_id, g.year, g.period ) SELECT year, series_id, sum(value) AS sum_value, title FROM step1 GROUP BY year, series_id, title ORDER BY series_id, year ;
```

```
--CREATE VIEW men_and_women_in_government AS with tot_data as( SELECT g.series_id AS series_id_tot, g.year AS year, g.period AS period, g.value AS value, s.series_title AS title FROM public."ce_data_90a_Government_Employment" g LEFT JOIN PUBLIC.ce_series s ON g.series_id = TRIM(s.series_id) where right(g.series_id,2) = '01' GROUP BY g.year, g.period, g.series_id, g.value, s.series_title ORDER BY g.series_id, g.year, g.period ) SELECT s.series_id_tot AS series_id_tot, w.series_id AS series_id_women, s.year, sum(s.value) AS sum_value_tot, w.title, w.sum_value AS sum_value_women, sum(s.value)- w.sum_value AS sum_value_men, (w.sum_value/sum(s.value)) AS women_gov_ratio, ((sum(s.value)- w.sum_value)/sum(s.value)) AS men_gov_ratio FROM tot_data s RIGHT JOIN public.woman_in_government w ON (s.year = w.year AND left(s.series_id_tot,11) = left(w.series_id,11)) GROUP BY s.series_id_tot, s.year, w.series_id, w.sum_value, w.title ORDER BY s.series_id_tot, s.year ;
```

```
--CREATE VIEW all_percent_women_in_government AS SELECT * FROM crosstab('select year, series_id_women, women_gov_ratio from men_and_women_in_government order by 1,2') AS ct ("year" int, "CES9000000010" float8, "CES9091000010" float8, "CES9091100010" float8, "CES9091622010" float8, "CES9091911010" float8, "CES9091912010" float8, "CES9091999910" float8, "CES9092000010" float8, "CES9092161110" float8, "CES9092200010" float8, "CES9092262210" float8, "CES9092292010" float8, "CES9092299910" float8, "CES9093000010" float8, "CES9093161110" float8, "CES9093200010" float8, "CES9093222110" float8, "CES9093248010" float8, "CES9093262210" float8, "CES9093292010" float8, "CES9093299910" float8);
```

```
--CREATE VIEW all_amount_women_in_government AS SELECT * FROM crosstab('select year, series_id_women, sum_value_women from men_and_women_in_government order by 1,2') AS ct ("year" int, "CES9000000010" float8, "CES9091000010" float8, "CES9091100010" float8, "CES9091622010" float8, "CES9091911010" float8, "CES9091912010" float8, "CES9091999910" float8, "CES9092000010" float8, "CES9092161110" float8, "CES9092200010" float8, "CES9092262210" float8, "CES9092292010" float8, "CES9092299910" float8, "CES9093000010" float8, "CES9093161110" float8, "CES9093200010" float8, "CES9093222110" float8, "CES9093248010" float8, "CES9093262210" float8, "CES9093292010" float8, "CES9093299910" float8);
```

```
--CREATE VIEW all_amount_total_in_government AS SELECT * FROM crosstab('select year, series_id_women, sum_value_tot from
men_and_women_in_government order by 1,2') AS ct ("year" int, "CES9000000010" float8, "CES9091000010" float8, "CES9091100010" float8,
"CES9091622010" float8, "CES9091911010" float8, "CES9091912010" float8, "CES9091999910" float8, "CES9092000010" float8,
"CES9092161110" float8, "CES9092200010" float8, "CES9092262210" float8, "CES9092292010" float8, "CES9092299910" float8,
"CES9093000010" float8, "CES9093161110" float8, "CES9093200010" float8, "CES9093222110" float8, "CES9093248010" float8,
"CES9093262210" float8, "CES9093292010" float8, "CES9093299910" float8);
```

```
CREATE VIEW all_amount_men_in_government AS SELECT * FROM crosstab('select year, series_id_women, sum_value_men from
men_and_women_in_government order by 1,2') AS ct ("year" int, "CES9000000010" float8, "CES9091000010" float8, "CES9091100010" float8,
"CES9091622010" float8, "CES9091911010" float8, "CES9091912010" float8, "CES9091999910" float8, "CES9092000010" float8,
"CES9092161110" float8, "CES9092200010" float8, "CES9092262210" float8, "CES9092292010" float8, "CES9092299910" float8,
"CES9093000010" float8, "CES9093161110" float8, "CES9093200010" float8, "CES9093222110" float8, "CES9093248010" float8,
"CES9093262210" float8, "CES9093292010" float8, "CES9093299910" float8);
```

```
CREATE VIEW woman_in_government_month AS with step1 as( SELECT g.series_id AS series_id, g.year AS year, g.period AS period, g.value AS
value, s.series_title AS title FROM public."ce_data_90a_Government_Employment" g LEFT JOIN PUBLIC.ce_series s ON g.series_id =
TRIM(s.series_id) where right(g.series_id,2) = '10' GROUP BY g.year, g.period, g.series_id, g.value, s.series_title ORDER BY g.series_id, g.year,
g.period ) SELECT year, series_id, period AS period, value AS value, title FROM step1 GROUP BY year, series_id, title,value, period ORDER BY
series_id, year, period
```

```
--CREATE VIEW men_and_women_in_government_month AS with tot_data as( SELECT g.series_id AS series_id_tot, g.year AS year, g.period AS
period, g.value AS value, s.series_title AS title FROM public."ce_data_90a_Government_Employment" g LEFT JOIN PUBLIC.ce_series s ON
g.series_id = TRIM(s.series_id) where right(g.series_id,2) = '01' GROUP BY g.year, g.period, g.series_id, g.value, s.series_title ORDER BY
g.series_id, g.year, g.period ) SELECT s.series_id_tot AS series_id_tot, w.series_id AS series_id_women, s.year, w.period,
TO_DATE(CONCAT(s.year,replace(w.period, 'M', '-')),'YYYY-MM') AS date, s.value AS value_tot, w.title, w.value AS value_women, s.value - w.value
AS value_men, (w.value/s.value) AS women_gov_ratio, ((s.value- w.value)/s.value) AS men_gov_ratio FROM tot_data s RIGHT JOIN
public.woman_in_government_month w ON (s.year = w.year AND left(s.series_id_tot,11) = left(w.series_id,11) AND w.period = s.period) WHERE
w.period != 'M13' GROUP BY s.series_id_tot, s.year, w.period, w.series_id, s.value, w.value, w.title ORDER BY s.series_id_tot, s.year, w.period
```

```
CREATE VIEW all_amount_women_in_government_month AS SELECT * FROM crosstab('select date, series_id_women, value_women from
men_and_women_in_government_month order by 1,2') AS ct (date date, "CES9000000010" float8, "CES9091000010" float8, "CES9091100010"
float8, "CES9091622010" float8, "CES9091911010" float8, "CES9091912010" float8, "CES9091999910" float8, "CES9092000010" float8,
"CES9092161110" float8, "CES9092200010" float8, "CES9092262210" float8, "CES9092292010" float8, "CES9092299910" float8,
"CES9093000010" float8, "CES9093161110" float8, "CES9093200010" float8, "CES9093222110" float8, "CES9093248010" float8,
"CES9093262210" float8, "CES9093292010" float8, "CES9093299910" float8);
```

```
CREATE VIEW all_amount_men_in_government_month AS SELECT * FROM crosstab('select date, series_id_women, value_men from
men_and_women_in_government_month order by 1,2') AS ct (date date, "CES9000000010" float8, "CES9091000010" float8, "CES9091100010"
float8, "CES9091622010" float8, "CES9091911010" float8, "CES9091912010" float8, "CES9091999910" float8, "CES9092000010" float8,
"CES9092161110" float8, "CES9092200010" float8, "CES9092262210" float8, "CES9092292010" float8, "CES9092299910" float8,
"CES9093000010" float8, "CES9093161110" float8, "CES9093200010" float8, "CES9093222110" float8, "CES9093248010" float8,
"CES9093262210" float8, "CES9093292010" float8, "CES9093299910" float8);
```

```
CREATE VIEW all_amount_total_in_government_month AS SELECT * FROM crosstab('select date, series_id_women, value_tot from
men_and_women_in_government_month order by 1,2') AS ct (date date, "CES9000000010" float8, "CES9091000010" float8, "CES9091100010"
float8, "CES9091622010" float8, "CES9091911010" float8, "CES9091912010" float8, "CES9091999910" float8, "CES9092000010" float8,
"CES9092161110" float8, "CES9092200010" float8, "CES9092262210" float8, "CES9092292010" float8, "CES9092299910" float8,
"CES9093000010" float8, "CES9093161110" float8, "CES9093200010" float8, "CES9093222110" float8, "CES9093248010" float8,
"CES9093262210" float8, "CES9093292010" float8, "CES9093299910" float8);
```

```
CREATE VIEW d05_total_private_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_05a_TotalPrivate_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),
```

```
production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_05a_TotalPrivate_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year )
```

```
SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
(p.sum_value_prod/(t.sum_value_tot-p.sum_value_prod)) AS prod_superv_ratio, (p.sum_value_prod/(t.sum_value_tot)) AS prod_vs_total_ratio,
p.series_title_prod, t.series_title_tot FROM total_data t RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) =
left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year
```

```
CREATE VIEW d10_miningandlogging_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_10a_MiningAndLogging_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),

production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_10a_MiningAndLogging_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year )

SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
(p.sum_value_prod/(t.sum_value_tot-p.sum_value_prod)) AS prod_superv_ratio, (p.sum_value_prod/(t.sum_value_tot)) AS prod_vs_total_ratio,
p.series_title_prod, t.series_title_tot FROM total_data t RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) =
left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year
```

```
CREATE VIEW d20_construction_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_20a_Construction_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),

production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_20a_Construction_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year )

SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
(p.sum_value_prod/(t.sum_value_tot-p.sum_value_prod)) AS prod_superv_ratio, (p.sum_value_prod/(t.sum_value_tot)) AS prod_vs_total_ratio,
p.series_title_prod, t.series_title_tot FROM total_data t RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) =
left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year
```

```
CREATE VIEW d30_manufacturing_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_30a_Manufacturing_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),

production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_30a_Manufacturing_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year )

SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
(p.sum_value_prod/(t.sum_value_tot-p.sum_value_prod)) AS prod_superv_ratio, (p.sum_value_prod/(t.sum_value_tot)) AS prod_vs_total_ratio,
p.series_title_prod, t.series_title_tot FROM total_data t RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) =
left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year
```

```
CREATE VIEW d40_trade_transportation_and_utilities_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_40a_TradeTransportationAndUtilities_Employment" t LEFT
JOIN public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13'
GROUP BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),

production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_40a_TradeTransportationAndUtilities_Employment" t
LEFT JOIN public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period !=
'M13' GROUP BY t.series_id, t.year, s.series_title order by t.series_id, t.year )

SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
(p.sum_value_prod/(t.sum_value_tot-p.sum_value_prod)) AS prod_superv_ratio, (p.sum_value_prod/(t.sum_value_tot)) AS prod_vs_total_ratio,
p.series_title_prod, t.series_title_tot FROM total_data t RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) =
left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year
```

```
CREATE VIEW d41_wholesale_trade_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_41a_WholesaleTrade_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),

production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_41a_WholesaleTrade_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year )

SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot-p.sum_value_prod,0)),0) AS prod_superv_ratio,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot,0)),0) AS prod_vs_total_ratio, p.series_title_prod, t.series_title_tot FROM total_data t
RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) = left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year
```

```
CREATE VIEW d42_retail_trade_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_42a_RetailTrade_Employment" t LEFT JOIN public.ce_series
s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP BY t.series_id,
t.year, s.series_title order by t.series_id, t.year ),

production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_42a_RetailTrade_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year )

SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot-p.sum_value_prod,0)),0) AS prod_superv_ratio,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot,0)),0) AS prod_vs_total_ratio, p.series_title_prod, t.series_title_tot FROM total_data t
RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) = left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year
```

```
--CREATE VIEW d43_transportation_warehousing_utilities_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM
public."ce_data_43a_TransportationAndWarehousingAndUtilities_Employment" t LEFT JOIN public.ce_series s ON TRIM(s.series_id) = t.series_id
where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP BY t.series_id, t.year, s.series_title order by t.series_id,
t.year ),
production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM
public."ce_data_43a_TransportationAndWarehousingAndUtilities_Employment" t LEFT JOIN public.ce_series s ON TRIM(s.series_id) = t.series_id
where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP BY t.series_id, t.year, s.series_title order by t.series_id,
t.year )
```

```
SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot-p.sum_value_prod,0)),0) AS prod_superv_ratio,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot,0)),0) AS prod_vs_total_ratio, p.series_title_prod, t.series_title_tot FROM total_data t
RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) = left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year
```

```
CREATE VIEW d50_information_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_50a_Information_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),
```

```
production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_50a_Information_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year )
```

```
SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot-p.sum_value_prod,0)),0) AS prod_superv_ratio,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot,0)),0) AS prod_vs_total_ratio, p.series_title_prod, t.series_title_tot FROM total_data t
RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) = left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year
```

```
CREATE VIEW d55_financial_activities_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_55a_FinancialActivities_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),
production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_55a_FinancialActivities_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year )

SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot-p.sum_value_prod,0)),0) AS prod_superv_ratio,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot,0)),0) AS prod_vs_total_ratio, p.series_title_prod, t.series_title_tot FROM total_data t
RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) = left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year

CREATE VIEW d60_professional_business_services_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_60a_ProfessionalBusinessServices_Employment" t LEFT
JOIN public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13'
GROUP BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),

production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_60a_ProfessionalBusinessServices_Employment" t LEFT
JOIN public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13'
GROUP BY t.series_id, t.year, s.series_title order by t.series_id, t.year )

SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot-p.sum_value_prod,0)),0) AS prod_superv_ratio,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot,0)),0) AS prod_vs_total_ratio, p.series_title_prod, t.series_title_tot FROM total_data t
RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) = left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year

CREATE VIEW d65_education_and_healthcare_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_65a_EducationAndHealthCare_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),

production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_65a_EducationAndHealthCare_Employment" t LEFT
JOIN public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13'
GROUP BY t.series_id, t.year, s.series_title order by t.series_id, t.year )
```

```
SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot-p.sum_value_prod,0)),0) AS prod_superv_ratio,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot,0)),0) AS prod_vs_total_ratio, p.series_title_prod, t.series_title_tot FROM total_data t
RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) = left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year
```

```
CREATE VIEW d70_leisure_and_hospitality_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_70a_LeisureAndHospitality_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),
```

```
production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_70a_LeisureAndHospitality_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year )
```

```
SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot-p.sum_value_prod,0)),0) AS prod_superv_ratio,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot,0)),0) AS prod_vs_total_ratio, p.series_title_prod, t.series_title_tot FROM total_data t
RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) = left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year
```

```
CREATE VIEW d80_other_services_employment AS WITH total_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_tot, s.series_title AS series_title_tot FROM public."ce_data_80a_OtherServices_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '01') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year ),
```

```
production_data AS( SELECT t.series_id, t.year,
sum(t.value) as sum_value_prod, s.series_title as series_title_prod FROM public."ce_data_80a_OtherServices_Employment" t LEFT JOIN
public.ce_series s ON TRIM(s.series_id) = t.series_id where (right(t.series_id,2) = '06') and left(t.series_id,3) = 'CES' and period != 'M13' GROUP
BY t.series_id, t.year, s.series_title order by t.series_id, t.year )
```

```
SELECT p.series_id AS series_id_prod, t.series_id AS series_id_tot, p.year, t.sum_value_tot, p.sum_value_prod,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot-p.sum_value_prod,0)),0) AS prod_superv_ratio,
COALESCE((p.sum_value_prod/NULLIF(t.sum_value_tot,0)),0) AS prod_vs_total_ratio, p.series_title_prod, t.series_title_tot FROM total_data t
RIGHT JOIN production_data p ON p.year = t.year where (left(t.series_id,11) = left(p.series_id,11)) ORDER BY t.series_id, p.series_id,t.year

CREATE VIEW cross_d05_total_private_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio from
public.d05_total_private_employment order by 1,2') AS ct ("year" int, "CES0500000006" float8,"CES0600000006" float8,"CES0800000006"
float8);

CREATE VIEW cross_d10_miningandlogging_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio from
public.d10_miningandlogging_employment order by 1,2') AS ct ("year" int, "CES1000000006" float8,"CES1011330006" float8,"CES1021000006"
float8,"CES1021100006" float8,"CES1021200006" float8,"CES1021230006" float8,"CES1021231006" float8,"CES1021300006"
float8,"CES1021311206" float8);

CREATE VIEW cross_d20_construction_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio from
public.d20_construction_employment order by 1,2') AS ct ("year" int, "CES2000000006" float8,"CES2023600006" float8,"CES2023610006"
float8,"CES2023611506" float8,"CES2023611806" float8,"CES2023620006" float8,"CES2023621006" float8,"CES2023622006"
float8,"CES2023700006" float8,"CES2023710006" float8,"CES2023711006" float8,"CES2023712006" float8,"CES2023713006"
float8,"CES2023730006" float8,"CES2023800006" float8,"CES2023810006" float8,"CES2023811006" float8,"CES2023812006"
float8,"CES2023813006" float8,"CES2023814006" float8,"CES2023815006" float8,"CES2023816006" float8,"CES2023820006"
float8,"CES2023821006" float8,"CES2023822006" float8,"CES2023829006" float8,"CES2023830006" float8,"CES2023831006"
float8,"CES2023832006" float8,"CES2023833006" float8,"CES2023834006" float8,"CES2023835006" float8,"CES2023839006"
float8,"CES2023890006" float8,"CES2023891006" float8,"CES2023899006" float8);

CREATE VIEW cross_d20_construction_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio from
public.d20_construction_employment order by 1,2') AS ct ("year" int, "CES3000000006" float8);

CREATE VIEW cross_d30_manufacturing_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio from
public.d30_manufacturing_employment order by 1,2') AS ct ("year" int, "CES3000000006" float8);
```

```
CREATE VIEW cross_d40_trade_transportation_and_utilities_employment AS SELECT * FROM crosstab('select year, series_id_prod,
prod_superv_ratio from public.d40_trade_transportation_and_utilities_employment order by 1,2') AS ct ("year" int, "CES4000000006" float8);

--CREATE VIEW cross_d41_wholesale_trade_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio from
public.d41_wholesale_trade_employment order by 1,2') AS ct ("year" int, "CES4142000006" float8,"CES4142300006" float8,"CES4142310006"
float8,"CES4142311006" float8,"CES4142312006" float8,"CES4142320006" float8,"CES4142330006" float8,"CES4142331006"
float8,"CES4142340006" float8,"CES4142343006" float8,"CES4142345006" float8,"CES4142350006" float8,"CES4142360006"
float8,"CES4142361006" float8,"CES4142369006" float8,"CES4142370006" float8,"CES4142371006" float8,"CES4142372006"
float8,"CES4142374006" float8,"CES4142380006" float8,"CES4142381006" float8,"CES4142382006" float8,"CES4142383006"
float8,"CES4142384006" float8,"CES4142390006" float8,"CES4142391006" float8,"CES4142393006" float8,"CES4142400006"
float8,"CES4142410006" float8,"CES4142420006" float8,"CES4142430006" float8,"CES4142435006" float8,"CES4142440006"
float8,"CES4142441006" float8,"CES4142448006" float8,"CES4142450006" float8,"CES4142460006" float8,"CES4142470006"
float8,"CES4142480006" float8,"CES4142490006" float8,"CES4142491006" float8,"CES4142500006" float8);

CREATE VIEW cross_d42_retail_trade_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio from
public.d42_retail_trade_employment order by 1,2') AS ct ("year" int, "CES4200000006" float8, "CES4244100006" float8,"CES4244110006"
float8,"CES4244111006" float8,"CES4244112006" float8,"CES4244120006" float8,"CES4244130006" float8,"CES4244133006"
float8,"CES4244134006" float8,"CES4244400006" float8,"CES4244410006" float8,"CES4244414006" float8,"CES4244418006"
float8,"CES4244420006" float8,"CES4244500006" float8,"CES4244510006" float8,"CES4244511006" float8,"CES4244513006"
float8,"CES4244520006" float8,"CES4244525006" float8,"CES4244530006" float8,"CES4244900006" float8,"CES4244910006"
float8,"CES4244911006" float8,"CES4244912006" float8,"CES4244912106" float8,"CES4244912906" float8,"CES4244920006"
float8,"CES4245500006" float8,"CES4245600006" float8,"CES4245611006" float8,"CES4245612006" float8,"CES4245700006"
float8,"CES4245710006" float8,"CES4245711006" float8,"CES4245712006" float8,"CES4245720006" float8,"CES4245800006"
float8,"CES4245810006" float8,"CES4245820006" float8,"CES4245830006" float8,"CES4245900006" float8,"CES4245910006"
float8,"CES4245911006" float8,"CES4245920006" float8,"CES4245930006" float8,"CES4245940006" float8,"CES4245941006"
float8,"CES4245942006" float8,"CES4245950006" float8,"CES4245990006" float8,"CES4245991006" float8,"CES4245999106" float8);

CREATE VIEW cross_d43_transportation_warehousing_utilities_employment AS SELECT * FROM crosstab('select year, series_id_prod,
prod_superv_ratio from public.d43_transportation_warehousing_utilities_employment order by 1,2') AS ct ("year" int, "CES4300000006"
float8,"CES4348100006" float8,"CES4348400006" float8,"CES4348410006" float8,"CES4348411006" float8,"CES4348412006"
float8,"CES4348420006" float8,"CES4348421006" float8,"CES4348422006" float8,"CES4348423006" float8,"CES4348500006"
float8,"CES4348540006" float8,"CES4348590006" float8,"CES4348800006" float8,"CES4348810006" float8,"CES4348840006"
float8,"CES4348850006" float8,"CES4349200006" float8,"CES4349300006" float8,"CES4349311006" float8,"CES4422000006"
float8,"CES4422110006" float8,"CES4422111006" float8,"CES4422111206" float8,"CES4422112006" float8,"CES4422120006"
float8,"CES4422130006" float8);
```

```
CREATE VIEW cross_d50_information_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio from
public.d50_information_employment order by 1,2') AS ct ("year" int, "CES5000000006" float8,"CES5051200006" float8,"CES5051300006"
float8,"CES5051310006" float8,"CES5051311006" float8,"CES5051312006" float8,"CES5051320006" float8,"CES5051600006"
float8,"CES5051610006" float8,"CES5051620006" float8,"CES5051700006" float8,"CES5051710006" float8,"CES5051711006"
float8,"CES5051711106" float8,"CES5051711206" float8,"CES5051712006" float8,"CES5051780006" float8,"CES5051800006"
float8,"CES5051900006" float8);
```

```
CREATE VIEW cross_d55_financial_activities_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio from
public.d55_financial_activities_employment order by 1,2') AS ct ("year" int, "CES5500000006" float8,"CES5552200006" float8,"CES5552210006"
float8,"CES5552211006" float8,"CES5552213006" float8,"CES5552218006" float8,"CES5552220006" float8,"CES5552229106"
float8,"CES5552229206" float8,"CES5552229906" float8,"CES5552230006" float8,"CES5552300006" float8,"CES5552315006"
float8,"CES5552320006" float8,"CES5552390006" float8,"CES5552394006" float8,"CES5552400006" float8,"CES5552410006"
float8,"CES5552411006" float8,"CES5552412006" float8,"CES5552413006" float8,"CES5552420006" float8,"CES5552421006"
float8,"CES5552429006" float8,"CES5552429206" float8,"CES5553100006" float8,"CES5553110006" float8,"CES5553111006"
float8,"CES5553112006" float8,"CES5553120006" float8,"CES5553130006" float8,"CES5553131006" float8,"CES5553131106"
float8,"CES5553131206" float8,"CES5553200006" float8,"CES5553228306" float8,"CES5553240006" float8,"CES5553241006"
float8,"CES5553249006" float8);
```

```
CREATE VIEW cross_d60_professional_business_services_employment AS SELECT * FROM crosstab('select year, series_id_prod,
prod_superv_ratio from public.d60_professional_business_services_employment order by 1,2') AS ct ("year" int, "CES6000000006"
float8,"CES6054000006" float8,"CES6054110006" float8,"CES6054111006" float8,"CES6054119006" float8,"CES6054120006"
float8,"CES6054121106" float8,"CES6054121406" float8,"CES6054130006" float8,"CES6054131006" float8,"CES6054132006"
float8,"CES6054134006" float8,"CES6054137006" float8,"CES6054138006" float8,"CES6054140006" float8,"CES6054143006"
float8,"CES6054150006" float8,"CES6054151106" float8,"CES6054151206" float8,"CES6054160006" float8,"CES6054161006"
float8,"CES6054161106" float8,"CES6054161306" float8,"CES6054161406" float8,"CES6054170006" float8,"CES6054171006"
float8,"CES6054171506" float8,"CES6054172006" float8,"CES6054180006" float8,"CES6054181006" float8,"CES6054182006"
float8,"CES6054189006" float8,"CES6054190006" float8,"CES6054191006" float8,"CES6054194006" float8,"CES6055000006"
float8,"CES6055111206" float8,"CES6055111406" float8,"CES6056000006" float8,"CES6056100006" float8,"CES6056110006"
float8,"CES6056120006" float8,"CES6056130006" float8,"CES6056131006" float8,"CES6056132006" float8,"CES6056133006"
float8,"CES6056140006" float8,"CES6056142006" float8,"CES6056143006" float8,"CES6056144006" float8,"CES6056150006"
float8,"CES6056151006" float8,"CES6056160006" float8,"CES6056161006" float8,"CES6056162006" float8,"CES6056170006"
float8,"CES6056171006" float8,"CES6056172006" float8,"CES6056173006" float8,"CES6056190006" float8,"CES6056199006"
float8,"CES6056200006" float8,"CES6056210006" float8);
```

```
CREATE VIEW cross_d65_education_and_healthcare_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio
from public.d65_education_and_healthcare_employment order by 1,2') AS ct ("year" int, "CES6500000006" float8,"CES6562000006"
float8,"CES6562000106" float8,"CES6562100006" float8,"CES6562110006" float8,"CES6562120006" float8,"CES6562130006"
float8,"CES6562131006" float8,"CES6562132006" float8,"CES6562133006" float8,"CES6562134006" float8,"CES6562139006"
float8,"CES6562140006" float8,"CES6562142006" float8,"CES6562149006" float8,"CES6562149306" float8,"CES6562149806"
float8,"CES6562150006" float8,"CES6562151106" float8,"CES6562151206" float8,"CES6562160006" float8,"CES6562190006"
float8,"CES6562191006" float8,"CES6562199006" float8,"CES6562200006" float8,"CES6562210006" float8,"CES6562300006"
float8,"CES6562310006" float8,"CES6562320006" float8,"CES6562321006" float8,"CES6562322006" float8,"CES6562330006"
float8,"CES6562331106" float8,"CES6562331206" float8,"CES6562390006" float8,"CES6562400006" float8,"CES6562410006"
float8,"CES6562411006" float8,"CES6562412006" float8,"CES6562419006" float8,"CES6562420006" float8,"CES6562430006"
float8,"CES6562440006" float8);
```

```
CREATE VIEW cross_d70_leisure_and_hospitality_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio from
public.d70_leisure_and_hospitality_employment order by 1,2') AS ct ("year" int, "CES7000000006" float8,"CES7071000006"
float8,"CES7071100006" float8,"CES7071110006" float8,"CES7071120006" float8,"CES7071130006" float8,"CES7071150006"
float8,"CES7071200006" float8,"CES7071211006" float8,"CES7071219006" float8,"CES7071300006" float8,"CES7071320006"
float8,"CES7071391006" float8,"CES7071392006" float8,"CES7071393006" float8,"CES7071394006" float8,"CES7071395006"
float8,"CES7072000006" float8,"CES7072100006" float8,"CES7072110006" float8,"CES7072111006" float8,"CES7072120006"
float8,"CES7072200006" float8,"CES7072230006" float8,"CES7072231006" float8,"CES7072233006" float8,"CES7072240006"
float8,"CES7072250006" float8,"CES7072251106" float8,"CES7072251306" float8,"CES7072251406" float8,"CES7072251506" float8);
```

```
--CREATE VIEW cross_d80_other_services_employment AS SELECT * FROM crosstab('select year, series_id_prod, prod_superv_ratio from
public.d80_other_services_employment order by 1,2') AS ct ("year" int, "CES8000000006" float8,"CES8081100006" float8,"CES8081110006"
float8,"CES8081111006" float8,"CES8081111106" float8,"CES8081111406" float8,"CES8081112006" float8,"CES8081119006"
float8,"CES8081119206" float8,"CES8081119806" float8,"CES8081120006" float8,"CES8081130006" float8,"CES8081140006"
float8,"CES8081200006" float8,"CES8081210006" float8,"CES8081211006" float8,"CES8081219006" float8,"CES8081220006"
float8,"CES8081230006" float8,"CES8081233006" float8,"CES8081290006" float8,"CES8081291006" float8,"CES8081293006"
float8,"CES8081300006" float8,"CES8081320006" float8,"CES8081321106" float8,"CES8081330006" float8,"CES8081331106"
float8,"CES8081331906" float8,"CES8081340006" float8,"CES8081390006" float8,"CES8081391006" float8,"CES8081392006"
float8,"CES8081393006" float8,"CES8081399006" float8);
```