

Desafío 14: Segunda Entrega del Proyecto Final

SEGUNDA ENTREGA DEL PROYECTO FINAL

Formato: Tablero en archivo ejecutable de Power BI, .pbix. La documentación debe ser formato pdf.

Sugerencia: Adjuntar la fuente de datos del tablero, archivo plano xls, csv o txt. En caso de ser formato google slide, hacer público el documento.

Proyecto
Final



>>Objetivos Generales:

1. Desarrollar un tablero de control.
2. Documentar la iniciativa de análisis de datos.

>>Objetivos Específicos:

1. Implementar transformación de datos.
2. Diseñar el modelo relacional en Power BI.
3. Desarrollar medidas y columnas calculadas.
4. Implementar storytelling.
5. Crear gráficos eficientes.

SEGUNDA ENTREGA DEL PROYECTO FINAL

Formato: Tablero en archivo ejecutable de Power BI, .pbix. La documentación debe ser formato pdf.

Sugerencia: Adjuntar la fuente de datos del tablero, archivo plano xls, csv o txt. En caso de ser formato google slide, hacer público el documento.

Proyecto
Final



>>Se debe entregar:

Cada archivo debe llevar por título el nombre del proyecto y los nombres de los integrantes. En el caso del tablero, debe estar en la solapa de portada.

En el archivo pdf:

- Debe incluir la documentación presentada en la primera entrega de proyecto final.
- Objetivo del proyecto.
- Alcance
- Usuario final y nivel de aplicación del análisis.
- Transformaciones realizadas.
- Medidas calculadas generadas y sus fórmulas.
- Segmentaciones elegidas.

SEGUNDA ENTREGA DEL PROYECTO FINAL

Formato: Tablero en archivo ejecutable de Power BI: .pbix. La documentación debe ser formato pdf.

Sugerencia: Adjuntar la fuente de datos del tablero, archivo plano xls, csv o txt. En caso de ser formato google slide, hacer público el documento.

Proyecto
Final



En el archivo .pbix:

- Al menos tres solapas definidas.
- Modelo de datos relacional.
- Tabla de calendario.
- Indicadores generados como medidas calculadas.
- Segmentaciones definidas.
- Gráficos específicos de acuerdo al tipo de variables.
- Diseño que permita una comunicación efectiva (orden de lectura de los objetos).
- Tipografía uniforme.
- Paleta de colores uniforme.
- Logos e imágenes.

SEGUNDA ENTREGA DEL PROYECTO FINAL

Formato: Tablero en archivo ejecutable de Power BI: .pbix. La documentación debe ser formato pdf.

Sugerencia: Adjuntar la fuente de datos del tablero, archivo plano xls, csv o txt. En caso de ser formato google slide, hacer público el documento.

Proyecto
Final



Ejemplo:

Les dejamos como ejemplo el trabajo de Matías Tello [PDF](#) [.pbix](#)

 Base de Datos - Tablas.xlsx 

 Vacunación CABA contra el COVID-19.pbix 

 Vacunación contra el COVID-19 en la Ciudad Autónoma de Buenos Aires - Pr... 

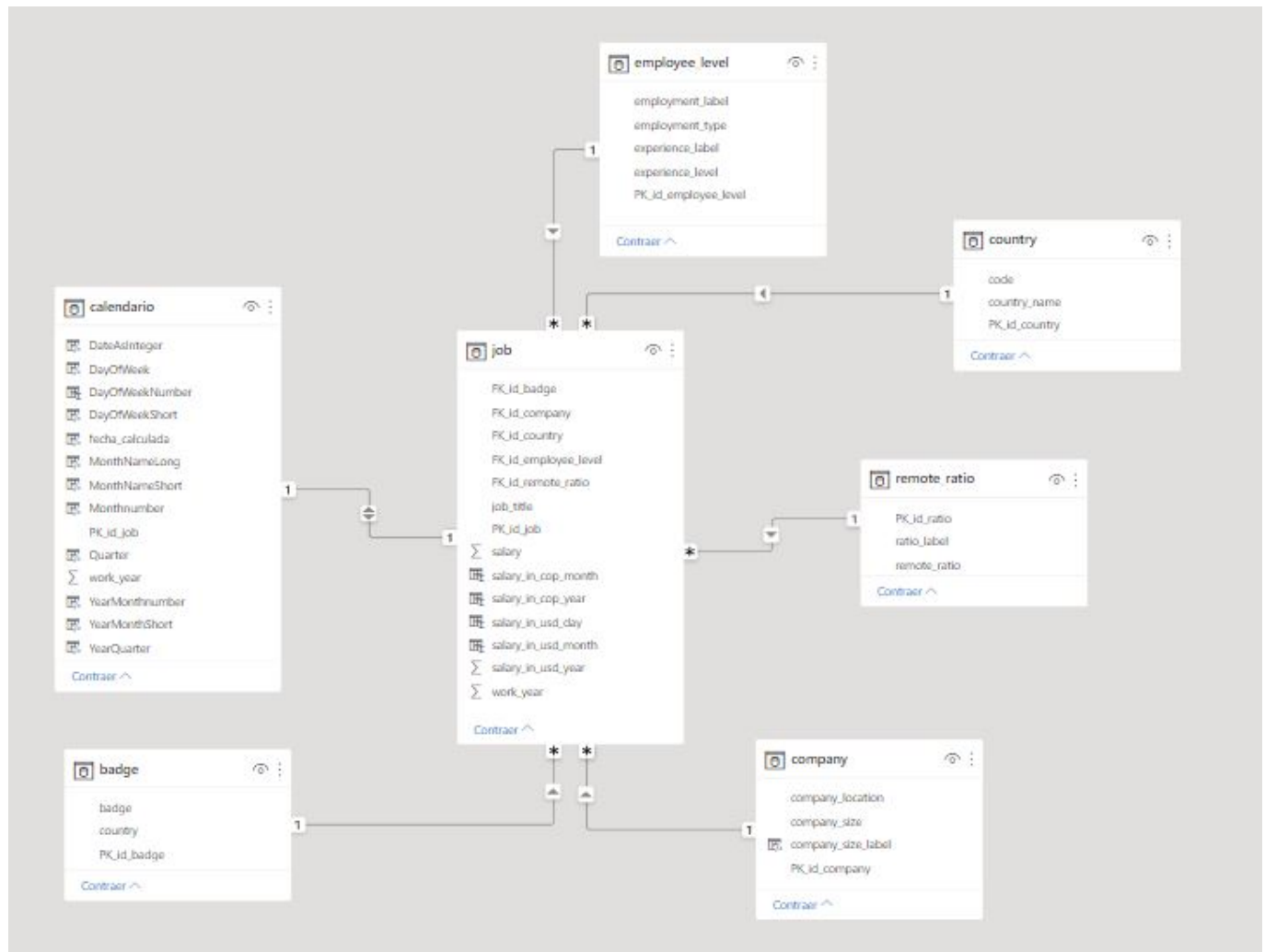
Tabla de contenido

- [Desafío 14: Segunda Entrega del Proyecto Final](#)
 - [Tabla de contenido](#)
 - [Objetivo del proyecto](#)
 - [Alcance](#)
 - [Usuario final y nivel de aplicación del análisis](#)
 - [Transformaciones realizadas](#)
 - [Medidas calculadas generadas y sus fórmulas](#)
 - [Segmentaciones elegidas](#)

Objetivo del proyecto

En el presente trabajo se realizó inicialmente una búsqueda de datasets que comprendían datos sobre salarios de los empleos en áreas de data science; seguido se realizó inicialmente el análisis de los datos que comprendían el dataset elegido, en este caso, "Data Science Jobs Salaries", el cual fue encontrado en el repositorio *Kaglee* al cual se puede acceder a través del siguiente [link](#)

Esquema en estrella: Según wikipedia, "*En las bases de datos usadas para data warehousing, un **esquema en estrella** es un modelo de datos que tiene una **tabla de hechos** (o tabla fact) que contiene los datos para el análisis, rodeada de las **tablas de dimensiones***". Dicho diagrama quedó de la siguiente forma:



Alcance

Usuario final y nivel de aplicación del análisis

Transformaciones realizadas

```
// Medidas de agregación
```

```
Promedio salario por mes en COP =
```

```
//Se analiza el promedio de salario de todos los tipos de empleos
```

```
AVERAGE('salary'[salary_in_cop_month])
```

```
// Medidas de recuento
```

```

Cantidad de países =
// Se distingue entre los países evaluados
DISTINCTCOUNT(employee[FK_id_country_code])

Cantidad empleados =
// Se cuentan los de registros de empleados
COUNT(employee[PK_id_employee])

Distintos tipos de empleos =
// Se analiza la cantidad de tipos de empleos
DISTINCTCOUNT(job[job_title])

```

Columnas calculadas:

▼	salary
	PK_id_salary
Σ	salary
	salary_currency
⌘	salary_in_cop_month
⌘	salary_in_cop_year
⌘	salary_in_usd_day
⌘	salary_in_usd_month
Σ	salary_in_usd_year
Σ	tipo_cambio

salary_currency ▼	salary_in_usd_year ▼	tipo_cambio ▼	salary_in_usd_month ▼	salary_in_usd_day ▼	salary_in_cop_year ▼	salary_in_cop_month ▼
USD	85000	1	7083,33	232,88	323000000	26916666,67
USD	230000	1	19166,67	630,14	874000000	72833333,33
USD	125000	1	10416,67	342,47	475000000	39583333,33
USD	120000	1	10000	328,77	456000000	38000000
USD	450000	1	37500	1232,88	1710000000	142500000
USD	144000	1	12000	394,52	547200000	45600000
USD	13400	1	1116,67	36,71	50920000	4243333,33
USD	150000	1	12500	410,96	570000000	47500000
USD	103000	1	8583,33	282,19	391400000	32616666,67

Para las anteriores columnas calculadas se usaron las siguientes expresiones de análisis de datos (DAX):

```

// Columnas calculadas

salary_in_usd_month = ROUND(salary[salary_in_usd_year]/12,2)

salary_in_usd_day = ROUND(salary[salary_in_usd_year]/365,2)

salary_in_cop_year = ROUND(salary[salary_in_usd_year]*3800,2)

salary_in_cop_month = ROUND(salary[salary_in_cop_year]/12,2)

```

Tabla calendario:

La presente tabla calendario fue generada automáticamente con la condición de que los rangos de fechas coincidieran con los indicados en la tabla jobs, dónde la columna *work_year* corresponde al año en el que se realizó el trabajo.

PK_id_job	work_year	fecha_calculada	DateAsInteger	DayOfWeek	DayOfWeekNumber	DayOfWeekShort	MonthNameLong	MonthNameShort	Monthnumber	Quarter	YearMonthnumb
1	2021	21/06/2021	20210621	lunes	2	lu.	junio	jun	06	Q2	2021/06
3	2021	14/09/2021	20210914	martes	3	ma.	septiembre	sep	09	Q3	2021/09
4	2021	10/05/2021	20210510	lunes	2	lu.	mayo	may	05	Q2	2021/05
5	2021	25/04/2021	20210425	domingo	1	do.	abril	abr	04	Q2	2021/04
6	2021	21/02/2021	20210221	domingo	1	do.	febrero	feb	02	Q1	2021/02
10	2021	14/02/2021	20210214	domingo	1	do.	febrero	feb	02	Q1	2021/02
11	2021	26/10/2021	20211026	martes	3	ma.	octubre	oct	10	Q4	2021/10
12	2021	19/03/2021	20210319	viernes	6	vi.	marzo	mar	03	Q1	2021/03

El código de fórmulas utilizado fue:

```

Calendario =
ADDCOLUMNS (
    CALENDAR("1/1/2019","31/12/2025"),
    "DateAsInteger", FORMAT ( [Date], "YYYYMMDD" ),
    "Year", YEAR ( [Date] ),
    "Monthnumber", FORMAT ( [Date], "MM" ),
    "YearMonthnumber", FORMAT ( [Date], "YYYY/MM" ),
    "YearMonthShort", FORMAT ( [Date], "YYYY/mmm" ),
    "MonthNameShort", FORMAT ( [Date], "mmm" ),
    "MonthNameLong", FORMAT ( [Date], "mmmm" ),
    "DayOfWeekNumber", WEEKDAY ( [Date] ),
    "DayOfWeek", FORMAT ( [Date], "dddd" ),
    "DayOfWeekShort", FORMAT ( [Date], "ddd" ),
    "Quarter", "Q" & FORMAT ( [Date], "Q" ),
    "YearQuarter", FORMAT ( [Date], "YYYY" ) & "/Q" & FORMAT ( [Date], "Q" )
)

```

Medidas calculadas generadas y sus fórmulas

-- Medidas calculada con una variable y al menos una medida de agregación

```

MCA Cantidad de Data Engineer con ratio codigo 3 = VAR ratio_de =
CALCULATE(DISTINCTCOUNT(job[PK_id_job]), remote_ratio[PK_id_ratio] = 3,
job[job_title] = "Data Engineer")
RETURN ratio_de

```

```

MCA Salario max de un Data engineer in usd year 2021 = VAR max_salary_dataengineer
= CALCULATE(MAX(job[salary_in_usd_year]), calendario[work_year] = 2021,
job[job_title] = "Data engineer")
RETURN max_salary_dataengineer

```

```

MCA Salario máximo en usd por año = VAR max_salary_usd =
MAX(job[salary_in_usd_year])
RETURN max_salary_usd

```

```

MCA Salario promedio en usd por año = VAR promedio_salario_usd_year =
AVERAGE(job[salary_in_usd_year])
RETURN promedio_salario_usd_year

```



```
-- Medida calculada con dos variables, al menos una función de agregación y una  
función de inteligencia de tiempo
```

```
MCA Diferencia del salario promedio por año de un Data Engineer =  
VAR promedio_2021 = CALCULATE(AVERAGE(job[salary_in_usd_year]), job[job_title] =  
"Data Engineer", job[work_year] = 2021)  
VAR promedio_2020 = CALCULATE(AVERAGE(job[salary_in_usd_year]), job[job_title] =  
"Data Engineer", job[work_year] = 2020)  
RETURN ((promedio_2020 - promedio_2021)/promedio_2020)
```

```
MCA Porcentaje de trabajo full remoto del total de registros =  
VAR ratio3 = CALCULATE(COUNT(job[job_title]),remote_ratio[PK_id_ratio] = 3)  
VAR cantidad_registros = COUNT(job[job_title])  
RETURN ratio3 / cantidad_registros
```

```
MCA Fecha de ultimo ingreso del mes = VAR fecha =  
ENDOFMONTH(calendario[fecha_calculada].[Date])  
RETURN fecha
```

```
-- Medida calculada con un parámetro con al menos una función de agregación
```

```
MCA Salario en cop por mes según parámetro dolar = SUM(job[salary_in_usd_month]) *  
[Valor precio_dolar]
```

Segmentaciones elegidas