

Data base creation project

Data base creation and automatization with .sql, .xml and .txt files

S O N I A R U I Z P É R E Z

OBJETIVO: El objetivo de este proyecto era devolverle al cliente (P. O.) una base de datos limpia y unificada a partir de tres archivos que él nos proporcionaba, así como automatizar la limpieza e introducción de futuros datos.

Archivo .SQL

```

/*!40103 SET @OLD_TIME_ZONE=@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
15 • /*!40101 SET @OLD_SQL_MODE=@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
16 • /*!40111 SET @OLD_SQL_NOTES=@SQL_NOTES, SQL_NOTES=0 */;

17 --
18 --
19 -- Dumping data for table `data_sql`
20 --
21 
22 • LOCK TABLES `data_sql` WRITE;
23 • /*!40000 ALTER TABLE `data_sql` DISABLE KEYS */;
24 • INSERT INTO `data_sql` VALUES (1,' Kaggle Notebooks','Colab Notebooks','ERROR','E
25 • INSERT INTO `data_sql` VALUES (6184,'ERROR','ERROR','ERROR','ERROR','ERR
26 • INSERT INTO `data_sql` VALUES (12373,' Kaggle Notebooks','Colab Notebooks','ERROR
27 • INSERT INTO `data_sql` VALUES (18553,'ERROR','ERROR','ERROR','ERROR','ERR
28 • INSERT INTO `data_sql` VALUES (24707,'ERROR','ERROR','ERROR','ERROR','ERR
29 • /*!40000 ALTER TABLE `data_sql` ENABLE KEYS */;
30 • UNLOCK TABLES;
31 • /*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;
32 
33 • /*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
34 • /*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
35 • /*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
36 • /*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
37 • /*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
38 • /*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

Archivo .XML

```

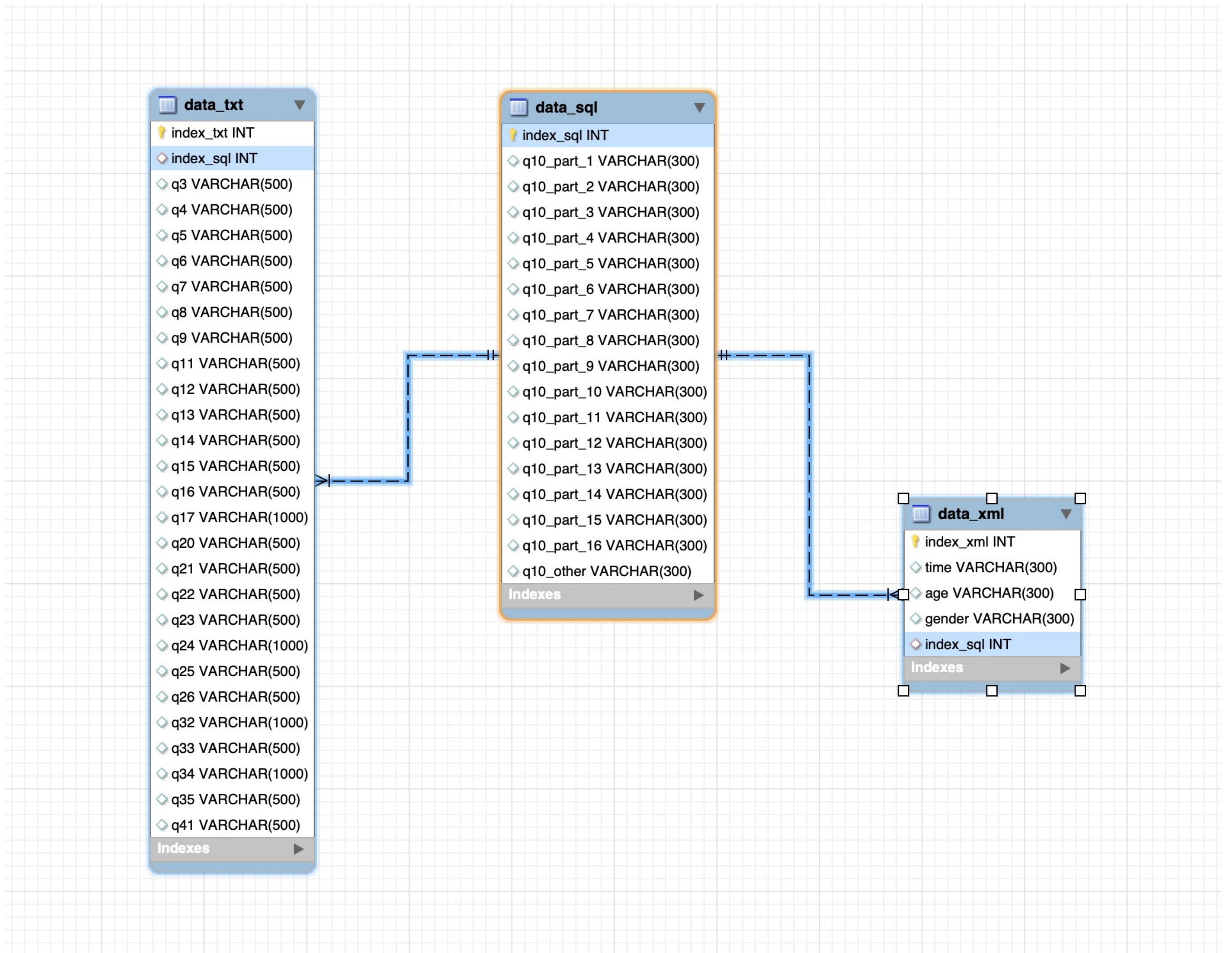
<?xml version="1.0" encoding="utf-8"?>
<data>
<row>
<level_0>1</level_0>
<index>1</index>
<time>784</time>
<age>50-54</age>
<gender>0</gender>
</row>
<row>
<level_0>2</level_0>
<index>2</index>
<time>924</time>
<age>22-24</age>
<gender>0</gender>
</row>
<row>
<level_0>3</level_0>
<index>3</index>
<time>575</time>
<age>45-49</age>
<gender>0</gender>
</row>
<row>
<level_0>4</level_0>
<index>4</index>
<time>781</time>
<age>45-49</age>
<gender>0</gender>
</row>
<row>
<level_0>5</level_0>
<index>5</index>
<time>1020</time>
<age>25-29</age>
<gender>1</gender>
</row>
<row>
<level_0>6</level_0>
<index>6</index>
<time>141</time>
<age>18-21</age>
<gender>1</gender>
</row>
<row>
<level_0>7</level_0>
<index>7</index>
<time>484</time>
<age>30-34</age>
<gender>0</gender>
</row>
<row>
<level_0>8</level_0>
<index>8</index>
<time>1744</time>
<age>22-24</age>
<gender>0</gender>
</row>
<row>
```

Archivo .XML

Archivo .TXT

```

index_sql;Q3;Q4;Q5;Q6;Q7;Q8;Q9;Q11;Q12;Q13;Q14;Q15;Q16;Q17;Q20;Q21;Q22;Q23;Q24;Q25;Q26;Q32;Q33;Q34;Q35;Q41
1;Indonesia;Master's degree;Program/Project Manager;20+ years;null, SQL, C, C++, Java;Python;null, Notepad++, Jupyter Notebook;A cloud computing platform (AWS, Azure, GCP, hosted notebooks, etc);null, None;Never;Matplotlib ;Under 1 year; Scikit-learn ;Linear or Logistic Regression, Decision Trees or Random Forests;Manufacturing/Fabrication;1000-9,999 employees;1-2;We are exploring ML methods (and may one day put a model into production);null, Build and/or run the data infrastructure that my business uses for storing, analyzing, and operationalizing data;60,000-69,999;$0 ($USD);;;Advanced statistical software (SPSS, SAS, etc.)
2;Pakistan;Master's degree;Software Engineer;1-3 years;Python, C++, Java;Python;null, PyCharm , Jupyter Notebook, Other;A laptop;null, Other;Never; Matplotlib ;I do not use machine learning methods;;;Academics/Education;1000-9,999 employees;0;I do not know;null, None of these activities are an important part of my role at work;$0-999;$0 ($USD);MySQL , MongoDB ;MySQL ;null, None;;Basic statistical software (Microsoft Excel, Google Sheets, etc.)
3;Mexico;Doctoral degree;Research Scientist;20+ years;Python;Python;null, Spyder , Jupyter Notebook;A cloud computing platform (AWS, Azure, GCP, hosted notebooks, etc); NVIDIA GPUs ;More than 25 times; Matplotlib ;5-10 years; Scikit-learn , TensorFlow , Keras ;null, Dense Neural Networks (MLPs, etc), Convolutional Neural Networks, Recurrent Neural Networks;Academics/Education;1000-9,999 employees;0;I do not know;null, Do research that advances the state of the art of machine learning;30,000-39,999;$0 ($USD);;;Local development environments (RStudio, JupyterLab, etc.)
4;India;Doctoral degree;Other;< 1 years;Python, C, MATLAB;Python;null, Spyder , MATLAB , Jupyter Notebook;A cloud computing platform (AWS, Azure, GCP, hosted notebooks, etc);null, None;Never; Matplotlib , Seaborn , Ggplot / ggplot2 ;10-20 years; Scikit-learn , PyTorch , LightGBM ;Linear or Logistic Regression, Decision Trees or Random Forests;xgboost, lightgbm, etc), Bayesian Approaches, Evolutionary Approaches;Academics/Education;50-249 employees;5-9;We use ML methods for generating insights (but do not put working models into production);Analyze and understand data to influence product or business decisions, Build prototypes to explore applying machine learning to new areas;30,000-39,999;$1000-$9,999;null, None;;null, Microsoft Power BI;;Local development environments (RStudio, JupyterLab, etc.)
5;India;I prefer not to answer;Currently not employed;< 1 years;Python;Python;JupyterLab, Jupyter Notebooks, etc , PyCharm , Spyder , Jupyter Notebook;A laptop;null, Google Cloud TPUs ;2-5 times; Matplotlib , Seaborn , Ggplot / ggplot2 ;Under 1 year; Scikit-learn , TensorFlow , Keras , PyTorch , Fast.ai ;Linear or Logistic Regression, Decision Trees or Random Forests;;;;;;Local development environments (RStudio, JupyterLab, etc.)
6;India;Some college/university study without earning a bachelor's degree;Student;1-3 years;null, C++, Java, Javascript;Python;null, Visual Studio , Visual Studio Code (VSCode) , Jupyter Notebook;A laptop;null, None;Never;null, Geoplotlib ;Under 1 year;null, Fast.ai ;;;;;;;
7;India;Bachelor's degree;Data Scientist;5-10 years;Python;Python;null, Jupyter Notebook;A personal computer / desktop;null, Google Cloud TPUs ;2-5 times; Matplotlib , Plotly / Plotly Express ;2-3 years; Scikit-learn , TensorFlow , Keras ;null, Decision Trees or Random Forests, Dense Neural Networks (MLPs, etc), Convolutional Neural Networks, Generative Adversarial Networks;Computers/Technology;10,000 or more employees;3-4;We have well established ML methods (i.e., models in production for more than 2 years);null, Build and/or run the data infrastructure that my business uses for storing, analyzing, and operationalizing data, Build prototypes to explore applying machine learning to new areas;15,000-19,999;$1-$99;MySQL ;null, None;;Basic statistical software (Microsoft Excel, Google Sheets, etc.)
8;Russia;Bachelor's degree;Currently not employed;3-5 years;Python, SQL;Python;null, Other;A cloud computing platform (AWS, Azure, GCP, hosted notebooks, etc);null, None;Never; Matplotlib ;Under 1 year; Scikit-learn , Xgboost ;null, Decision Trees or Random Forests, Gradient Boosting Machines (xgboost, lightgbm, etc);;;;;;;Basic statistical software (Microsoft Excel, Google Sheets, etc.)
9;Turkey;I prefer not to answer;Other;1-3 years;Python, SQL;SQL;null, Spyder , Jupyter Notebook;A laptop;null, None;Never; Matplotlib , Seaborn ;Under 1 year; Scikit-learn , TensorFlow ;Linear or Logistic Regression;Manufacturing/Fabrication;50-249 employees;1-2;I do not know;Analyze and understand data to influence product or business decisions;$0-999;$0 ($USD);;;Business intelligence software (Salesforce, Tableau, Spotfire, etc.)
10;Australia;Doctoral degree;Other;1-3 years;Python, R, SQL;R;null, RStudio , Jupyter Notebook;A personal computer / desktop;null, None;Never; Matplotlib , Seaborn , Ggplot / ggplot2 ;I do not use machine learning methods;;;Other;0-49 employees;0;No (we do not use ML methods);null, None of these activities are an important part of my role at work;70,000-79,999;$1-$99;MySQL ;null, Microsoft Power BI, Tableau, Alteryx ;Tableau;Local development environments (RStudio, JupyterLab, etc.)
11;India;Master's degree;Student;< 1 years;Python, R, C++;R;null, RStudio , Jupyter Notebook;A laptop;null, None;Never; Matplotlib , Seaborn , Ggplot / ggplot2 ;Under 1 year; Scikit-learn ;Linear or Logistic Regression;;;;;;Advanced statistical software (SPSS, SAS, etc.)
12;India;Master's degree;Student;< 1 years;Python, MATLAB;Python;null, MATLAB , Jupyter Notebook;A laptop;;;;;;;
13;Nigeria;Master's degree;Program/Project Manager;5-10 years;Python, SQL;Python;null, Spyder , Jupyter Notebook;A laptop;null, None;Never; Matplotlib , Seaborn ;1-2 years; Scikit-learn , Xgboost ;Linear or Logistic Regression, Decision Trees or Random Forests;Shipping/Transportation;1000-9,999 employees;10-14;We are exploring ML methods (and
```



Características de la Base de Datos:

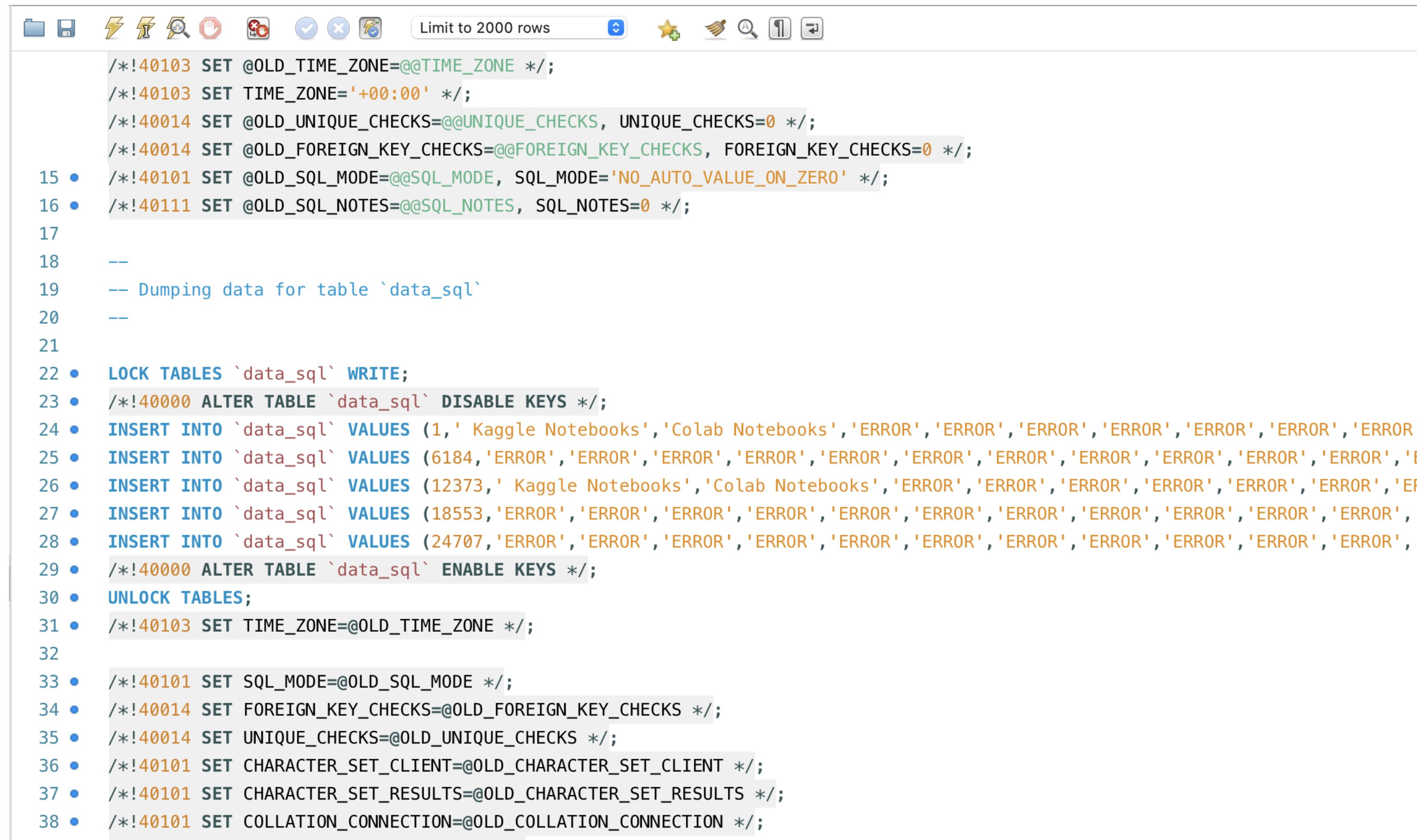
- Nombre de la Base de Datos: "project1":
- Nombre de las Tablas: "data_sql", "data_xml", "data_txt".

Las especificaciones que deben cumplir las tablas son:

- **data_sql**: tabla madre. Clave primaria **index_sql** (INT).
- **data_xml**: Su clave primaria es **index_xml** y su clave foránea es la columna **index_sql**, ambas de tipo numérico.
- **data_txt**: Su clave primaria es **index_txt** y su clave foránea es la columna **index_sql**, ambas de tipo numérico.

Las tres tablas se relacionarán mediante la columna **index_sql**.

Características del archivo .sql



The screenshot shows a MySQL Workbench interface with a SQL editor tab open. The editor displays a series of SQL commands, likely from a dump file, with line numbers on the left. The code includes various MySQL system variables being set (e.g., @OLD_TIME_ZONE, @OLD_SQL_MODE), table locking (LOCK TABLES), data insertion into a 'data_sql' table, and system variable restoration at the end. The code is color-coded for syntax highlighting.

```
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
15 • /*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
16 • /*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

17
18 --
19 -- Dumping data for table `data_sql`
20 --
21
22 • LOCK TABLES `data_sql` WRITE;
23 • /*!40000 ALTER TABLE `data_sql` DISABLE KEYS */;
24 • INSERT INTO `data_sql` VALUES (1,' Kaggle Notebooks','Colab Notebooks','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR'
25 • INSERT INTO `data_sql` VALUES (6184,'ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','E
26 • INSERT INTO `data_sql` VALUES (12373,' Kaggle Notebooks','Colab Notebooks','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','E
27 • INSERT INTO `data_sql` VALUES (18553,'ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','E
28 • INSERT INTO `data_sql` VALUES (24707,'ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','ERROR','E
29 • /*!40000 ALTER TABLE `data_sql` ENABLE KEYS */;
30 • UNLOCK TABLES;
31 • /*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

32
33 • /*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
34 • /*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
35 • /*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
36 • /*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
37 • /*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
38 • /*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

MySQL Workbench

Local instance 3306

Administration Schemas sql_code*

SCHEMAS

Filter objects

leccion-1-sql
leccion-2-sql
leccion-3-sql
leccion-4-sql
leccion-5-sql
leccion-6-sql
leccion-7-sql
leccion-8-sql
leccion-9-sql
mi_primerita_BBDD_Python
northwind
prueba_txt
sakila
spotify
sys

59 /*Para empezar con el proyecto, primero creamos la base de datos*/
60 • CREATE SCHEMA IF NOT EXISTS project1
61 ;
62
63 /*A continuación, entramos en la base de datos que acabamos de crear*/
64 • USE project1
65 ;
66
67 /*El tercer paso es crear las tres tablas que contendrán los datos que nos ha aportado el cliente.*/
68 • CREATE TABLE IF NOT EXISTS data_sql(
69 index_sql INT PRIMARY KEY,
70 q10_part_1 VARCHAR (300),
71 q10_part_2 VARCHAR (300),
72 q10_part_3 VARCHAR (300),
73 q10_part_4 VARCHAR (300),
74 q10_part_5 VARCHAR (300),
75 q10_part_6 VARCHAR (300),
76 q10_part_7 VARCHAR (300),
77 q10_part_8 VARCHAR (300),

Object Info Session 100% 1:60

No object selected

Action Output

	Time	Action	Response	Duration / Fetch Time
16	09:23:14	UPDATE data_sql SET q10_part_6=NULL WHERE q10_part_6='ERROR'	24542 row(s) affected Rows matched: 24542 Changed: 24542 warnings: 0	0.429 sec
17	09:23:15	UPDATE data_sql SET q10_part_7=NULL WHERE q10_part_7='ERROR'	23701 row(s) affected Rows matched: 23701 Changed: 23701 Warnings: 0	0.469 sec
18	09:23:15	UPDATE data_sql SET q10_part_8=NULL WHERE q10_part_8='ERROR'	23990 row(s) affected Rows matched: 23990 Changed: 23990 Warnings: 0	0.435 sec
19	09:23:16	UPDATE data_sql SET q10_part_9=NULL WHERE q10_part_9='ERROR'	24380 row(s) affected Rows matched: 24380 Changed: 24380 Warnings: 0	0.411 sec
20	09:23:16	UPDATE data_sql SET q10_part_10=NULL WHERE q10_part_10='ERROR'	22767 row(s) affected Rows matched: 22767 Changed: 22767 Warnings: 0	0.372 sec
21	09:23:16	UPDATE data_sql SET q10_part_11=NULL WHERE q10_part_11='ERROR'	23017 row(s) affected Rows matched: 23017 Changed: 23017 Warnings: 0	0.358 sec
22	09:23:17	UPDATE data_sql SET q10_part_12=NULL WHERE q10_part_12='ERROR'	24144 row(s) affected Rows matched: 24144 Changed: 24144 Warnings: 0	0.427 sec
23	09:23:17	UPDATE data_sql SET q10_part_13=NULL WHERE q10_part_13='ERROR'	24470 row(s) affected Rows matched: 24470 Changed: 24470 Warnings: 0	0.640 sec
24	09:23:18	UPDATE data_sql SET q10_part_14=NULL WHERE q10_part_14='ERROR'	24460 row(s) affected Rows matched: 24460 Changed: 24460 Warnings: 0	0.387 sec
25	09:23:18	UPDATE data_sql SET q10_part_15=NULL WHERE q10_part_15='ERROR'	24567 row(s) affected Rows matched: 24567 Changed: 24567 Warnings: 0	0.394 sec
26	09:23:19	UPDATE data_sql SET q10_part_16=NULL WHERE q10_part_16='ERROR'	17871 row(s) affected Rows matched: 17871 Changed: 17871 Warnings: 0	0.316 sec
27	09:23:19	UPDATE data_sql SET q10_other=NULL WHERE q10_other='ERROR'	24000 row(s) affected Rows matched: 24000 Changed: 24000 Warnings: 0	0.409 sec

```
<?xml version="1.0" encoding="utf-8"?>
<data>
<row>
<level_0>1</level_0>
<index>1</index>
<time>784</time>
<age>50-54</age>
<gender>0</gender>
</row>
<row>
<level_0>2</level_0>
<index>2</index>
<time>924</time>
<age>22-24</age>
<gender>0</gender>
</row>
<row>
<level_0>3</level_0>
<index>3</index>
<time>575</time>
<age>45-49</age>
<gender>0</gender>
</row>
<row>
<level_0>4</level_0>
<index>4</index>
<time>781</time>
<age>45-49</age>
<gender>0</gender>
</row>
<row>
<level_0>5</level_0>
<index>5</index>
<time>1020</time>
<age>25-29</age>
<gender>1</gender>
</row>
<row>
<level_0>6</level_0>
<index>6</index>
<time>141</time>
<age>18-21</age>
<gender>1</gender>
</row>
<row>
<level_0>7</level_0>
<index>7</index>
<time>484</time>
<age>30-34</age>
<gender>0</gender>
</row>
<row>
<level_0>8</level_0>
<index>8</index>
<time>1744</time>
<age>22-24</age>
<gender>0</gender>
</row>
```

Características del archivo .xml

Objetivos en la depuración del fichero XML:

- Omitir columnas con contenido redundante.
- Recodificar la variable de género ('Man', 'Woman', 'Non-binary', etc.)

Novedades respecto a la depuración:

- Se ha omitido la columna con contenido redundante (level_0)
- Variable género: Numérica → String (conjunto de caracteres)
- Relacionada con la tabla 'data_sql' mediante la columna 'index_sql'

ANTES

level_0
index_xml
time
age
gender

DESPUÉS

index_xml	INT AI PK
time	VARCHAR(300)
age	VARCHAR(300)
gender	VARCHAR (300)
index_sql	INT FK

DEF_limpieza_xml.ipynb

+ Código + Markdown | ⚡ Interrumpir ✖ Borrar resultados de todas las celdas ⏪ Ir a ⏴ Reiniciar ↗ Execute Group 1 ↘ Execute Group 2 | 📄 Variables 📑 Esquema ... base (Python 3.9.13)

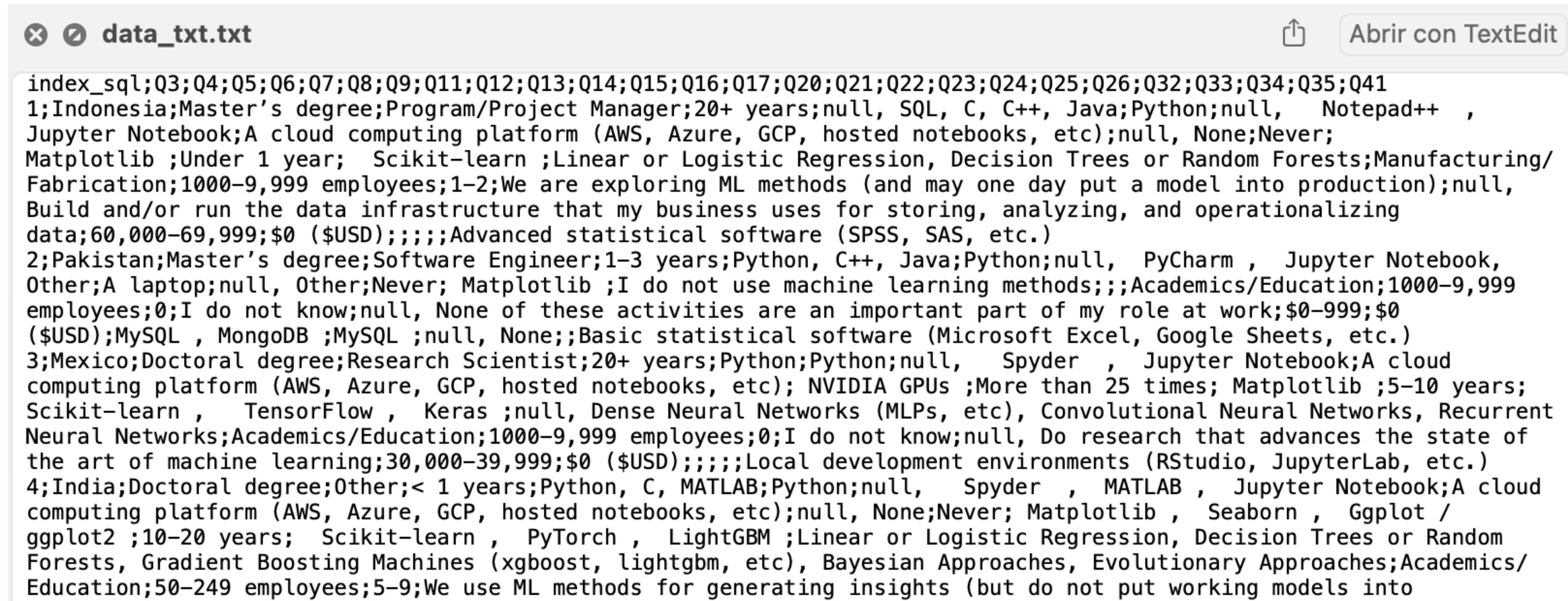
```
def limpia_xml(self):
    def clean_gender(gender: str):
        if (gender == '0'):
            return 'Man'
        if (gender == '1'):
            return 'Woman'
        if (gender == '2'):
            return 'Nonbinary'
        if (gender == '3'):
            return 'Prefer not to say'
        if (gender == '4'):
            return 'Prefer to self-describe'

        for row in self.root.iter('row'):
            level_column = row.find('level_0')
            row.remove(level_column)
            gender_column = row.find('gender')
            gender_column.text = clean_gender(gender_column.text)

    def insertar_xml_sql(self):
        # Para evitar que nuestro código se pare utilizamos un try y except al hacer la conexión con
        # la base de datos de "project1".
        try:
            cnx = mysql.connector.connect(user='root', password='AlumnaAdalab',
                                          host='127.0.0.1',
                                          database='project1')
            print('Conexión con la base de datos realizada')
        except mysql.connector.Error as err:
            if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
                print(
                    "Ha habido un error al introducir el nombre de usuario o la contraseña")
            elif err.errno == errorcode.ER_BAD_DB_ERROR:
                print("La base de datos no existe")
            else:
                print(err)
        # Comprobamos que tenemos las tablas correctamente creadas en nuestra base de datos.
        # creamos el servidor
        cursor = cnx.cursor()
        # ejecutamos la query para ver las tablas de la BBDD que hemos indicado en la conexión
        cursor.execute("SHOW TABLES")
        print(cursor.fetchall())

```

Características del archivo .txt



The screenshot shows a Mac OS X TextEdit window with the following details:

- Title Bar:** Shows the file name "data_txt.txt".
- Toolbar:** Includes standard OS X icons for close, minimize, maximize, and "Abrir con TextEdit".
- Content Area:** Displays a list of strings separated by semicolons. Each string contains various pieces of information, such as country, degree, role, experience, and tools used.

Sample content from the file:

```
index_sql;Q3;Q4;Q5;Q6;Q7;Q8;Q9;Q11;Q12;Q13;Q14;Q15;Q16;Q17;Q20;Q21;Q22;Q23;Q24;Q25;Q26;Q32;Q33;Q34;Q35;Q41
1;Indonesia;Master's degree;Program/Project Manager;20+ years;null, SQL, C, C++, Java;Python;null, Notepad++ ,
Jupyter Notebook;A cloud computing platform (AWS, Azure, GCP, hosted notebooks, etc);null, None;Never;
Matplotlib ;Under 1 year; Scikit-learn ;Linear or Logistic Regression, Decision Trees or Random Forests;Manufacturing/
Fabrication;1000-9,999 employees;1-2;We are exploring ML methods (and may one day put a model into production);null,
Build and/or run the data infrastructure that my business uses for storing, analyzing, and operationalizing
data;60,000-69,999;$0 ($USD);;;Advanced statistical software (SPSS, SAS, etc.)
2;Pakistan;Master's degree;Software Engineer;1-3 years;Python, C++, Java;Python;null, PyCharm , Jupyter Notebook,
Other;A laptop;null, Other;Never; Matplotlib ;I do not use machine learning methods;;Academics/Education;1000-9,999
employees;0;I do not know;null, None of these activities are an important part of my role at work;$0-999;$0
($USD);MySQL , MongoDB ;MySQL ;null, None;;Basic statistical software (Microsoft Excel, Google Sheets, etc.)
3;Mexico;Doctoral degree;Research Scientist;20+ years;Python;Python;null, Spyder , Jupyter Notebook;A cloud
computing platform (AWS, Azure, GCP, hosted notebooks, etc); NVIDIA GPUs ;More than 25 times; Matplotlib ;5-10 years;
Scikit-learn , TensorFlow , Keras ;null, Dense Neural Networks (MLPs, etc), Convolutional Neural Networks, Recurrent
Neural Networks;Academics/Education;1000-9,999 employees;0;I do not know;null, Do research that advances the state of
the art of machine learning;30,000-39,999;$0 ($USD);;;Local development environments (RStudio, JupyterLab, etc.)
4;India;Doctoral degree;Other;< 1 years;Python, C, MATLAB;Python;null, Spyder , MATLAB , Jupyter Notebook;A cloud
computing platform (AWS, Azure, GCP, hosted notebooks, etc);null, None;Never; Matplotlib , Seaborn , Ggplot /
ggplot2 ;10-20 years; Scikit-learn , PyTorch , LightGBM ;Linear or Logistic Regression, Decision Trees or Random
Forests, Gradient Boosting Machines (xgboost, lightgbm, etc), Bayesian Approaches, Evolutionary Approaches;Academics/
Education;50-249 employees;5-9;We use ML methods for generating insights (but do not put working models into
```

- Una lista de strings.
- Presenta un indice .sql para cada fila.
- Limpieza:
 - Eliminar espacios innecesarios.
 - Cambiar "<" por "under".
 - Cambiar "null" por "NULL".
 - Eliminar "\n " al final de cada elemento.

¿Que apariencia tiene el archivo que nos llega?

The screenshot shows a Jupyter Notebook interface with two code cells and their corresponding outputs.

Code Cell 26:

```
# En primer lugar guardamos el contenido del archivo .txt en la variable "file_source".
with open('data_txt.txt', 'r') as file:
    file_source = file.readlines()
```

Output for Cell 26:

```
[26] ✓ 1.1s
```

Code Cell 27:

```
# Leemos el contenido de la variable. Y vemos que es una lista, donde cada elemento
# es una fila de la futura tabla.
file_source
```

Output for Cell 27:

```
[27] ✓ 0.1s
```

Output for both cells:

```
... Output exceeds the size limit. Open the full output data in a text editor
['index_sql;Q3;Q4;Q5;Q6;Q7;Q8;Q9;Q11;Q12;Q13;Q14;Q15;Q16;Q17;Q20;Q21;Q22;Q23;Q24;Q25;Q26;Q32;Q33;Q34;Q35;Q41\n',
 '1;Indonesia;Master's degree;Program/Project Manager;20+ years;null, SQL, C, C++, Java;Python;null, Notepad++, Jupyter
Notebook;A cloud computing platform (AWS, Azure, GCP, hosted notebooks, etc);null, None;Never; Matplotlib ;Under 1 year;
Scikit-learn ;Linear or Logistic Regression, Decision Trees or Random Forests;Manufacturing/Fabrication;1000-9,999 employees;1-
2;We are exploring ML methods (and may one day put a model into production);null, Build and/or run the data infrastructure that
my business uses for storing, analyzing, and operationalizing data;60,000-69,999;$0 ($USD);;;;Advanced statistical software
(SPSS, SAS, etc. )\n',
 '2;Pakistan;Master's degree;Software Engineer;1-3 years;Python, C++, Java;Python;null, PyCharm , Jupyter Notebook, Other;A
laptop;null, Other;Never; Matplotlib ;I do not use machine learning methods;;;Academics/Education;1000-9,999 employees;0;I do
not know;null, None of these activities are an important part of my role at work;$0-999;$0 ($USD);MySQL , MongoDB ;MySQL ;null,
None;;Basic statistical software (Microsoft Excel, Google Sheets, etc.)\n',
 '3;Mexico;Doctoral degree;Research Scientist;20+ years;Python;Python;null, Spyder , Jupyter Notebook;A cloud computing
platform (AWS, Azure, GCP, hosted notebooks, etc); NVIDIA GPUs ;More than 25 times; Matplotlib ;5-10 years; Scikit-learn ,
TensorFlow , Keras ;null, Dense Neural Networks (MLPs, etc), Convolutional Neural Networks, Recurrent Neural
Networks;Academics/Education;1000-9,999 employees;0;I do not know;null, Do research that advances the state of the art of
machine learning;30,000-39,999;$0 ($USD);;;;Local development environments (RStudio, JupyterLab, etc.)\n',
 '4;India;Doctoral degree;Other;< 1 years;Python, C, MATLAB;Python;null, Spyder , MATLAB , Jupyter Notebook;A cloud
computing platform (AWS, Azure, GCP, hosted notebooks, etc);null, None;Never; Matplotlib , Seaborn , Gephi , ggplot2 ;10-20
```

At the bottom of the interface, there are several status indicators: main*, 12 ▲ 22, Live Share, Git Graph, Servidor de Jupyter: local, Celda 4 de 35, and a few other icons.

Desarrollo de funciones.

The screenshot shows a Jupyter Notebook environment with the following details:

- EXPLORADOR**: Shows files and checkpoints in the current project, including `txt_code.ipynb`, `productowner_code.ipynb`, `.ipynb_checkpoints`, `.gitignore`, `actas_daily_meetings`, `data_sql.sql`, `data_txt.csv`, `data_txt.txt`, `data_xml.xml`, `explicacion_creacion...`, `explicacion_sql_code`, `funcion-xml-intento...`, `intento-xml-tree_3.1...`, `productowner_... U`, `read_xml_txt.ipynb`, `README.md`, `sql_code.sql`, `txt_code.ipynb` (selected), and `txt_code3.ipynb`.
- EDITORES ... 2 sin guardar**: Displays two code cells:

 - Cell 31**: Python code defining a function `sustituir` that takes a list `lista_txt` and replaces '\n' with '' (empty string), '<' with 'under', and 'null' with 'NULL'. It returns the modified list `lista_sustituida`. Execution time: 0.6s.
 - Cell 32**: Python code executing the `sustituir` function on a variable `file_source` and printing the result. Execution time: 1.2s.

- Código + Markdown | Ejecutar todo | Borrar resultados de todas las celdas | Reiniciar | Variables | Esquema | ...**: Notebook navigation and execution controls.
- base (Python 3.9.7)**: Notebook kernel information.
- Python**: Cell type indicator for both cells.
- Servidor de Jupyter: local Celda 12 de 35**: Notebook status bar.

Desarrollo de funciones.

The screenshot shows a Jupyter Notebook environment with two code cells. The left sidebar lists files in the project directory, including `txt_code.ipynb`, `productowner_code.ipynb`, and `txt_code3.ipynb`. The top cell (cell 33) contains Python code to clean extra spaces from a text file:

```
# Tras la limpieza previa, procedemos a eliminar los espacios extra. Para ello creamos
# esta funcion para automatizar la eliminación de espacios extra en un archivo .txt.
def limpiar(archivo_txt):
    # Esta función sirve para eliminar los espacios innecesarios al principio y al final del
    # contenido de cada columna. El parámetro será una variable que almacene, en formato lista,
    # el contenido de un .txt.
    lista_espaciol=[]
    for entrada in archivo_txt:
        espacio1=entrada.split(";")
        lista_espaciol.append(espacio1)

    lista_espacio2=[]
    lista_espacio3=[]
    for columna in lista_espaciol:
        for elemento in columna:
            espacio2=elemento.strip()
            lista_espacio2.append(espacio2)
            if len(lista_espacio2)==len(columna):
                lista_espacio3.append(lista_espacio2)
                lista_espacio2=[]
            else:
                pass
    return lista_espacio3
```

The bottom cell (cell 34) contains the execution of the function on a variable named `file_source_sustituido`:

```
# Ejecutamos la función creada sobre la variable que tiene el contenido del .txt, con
# la limpieza previa ya hecha en este caso.
file_source_limpio=limpiar(file_source_sustituido)
print(file_source_limpio)
```

The notebook is running on a local server, and the current cell is 15 of 35.

Resultado preliminar.

The screenshot shows a Jupyter Notebook interface with the following details:

- EXPLORADOR**: Shows the project structure with files like `txt_code.ipynb`, `productowner_code.ipynb`, `.ipynb_checkpoints`, `.gitignore`, `actas_daily_meetings`, `data_sql.sql`, `data_txt.csv`, `data_txt.txt`, `data_xml.xml`, `explicacion_creacion...`, `explicacion_sql_code`, `funcion-xml-intento...`, `intento-xml-tree_3.1...`, `productowner_... u`, `read_xml_txt.ipynb`, `README.md`, `sql_code.sql`, `txt_code.ipynb` (selected), and `txt_code3.ipynb`.
- EDITORES ... 2 sin guardar**: Displays two code cells. The first cell contains the following Python code:

```
# Ejecutamos la funcion creada sobre la variable que tiene el contenido del .txt, con
# la limpieza previa ya hecha en este caso.
file_source_limpio=limpiar(file_source_sustituido)
print(file_source_limpio)
```

The cell has a green checkmark and a duration of `1.6s`. The second cell's content is partially visible as `...`.
- Variables**: Shows variables defined in the notebook.
- ESQUEMA**: Shows the schema of the data.
- LÍNEA DE TIEMPO**: Shows the timeline of changes.
- BOTONES**: Includes buttons for Código, Markdown, Ejecutar todo, Borrar resultados de todas las celdas, Reiniciar, Variables, Esquema, and base (Python 3.9.7).
- ESTADÍSTICAS**: Shows main*, 12, 22, Live Share, Git Graph, Servidor de Jupyter: local, Celda 15 de 35, and other status indicators.

Desarrollo de clase.

The screenshot shows a Jupyter Notebook interface with the following details:

- EXPLORADOR**: Shows the project structure with files like `txt_code.ipynb`, `productowner_code.ipynb`, `.ipynb_checkpoints`, `.gitignore`, `actas_daily_meetings`, `data_sql.sql`, `data_txt.csv`, `data_txt.txt`, `data_xml.xml`, `explicacion_creacion...`, `explicacion_sql_code`, `funcion-xml-intento...`, `intento-xml-tree_3.1...`, `productowner_...`, `read_xml_txt.ipynb`, `README.md`, `sql_code.sql`, `txt_code.ipynb` (selected), and `txt_code3.ipynb`.
- EDTORES**: Displays two notebooks: `txt_code.ipynb` and `productowner_code.ipynb`.
- Barra superior**: Includes tabs for `txt_code.ipynb` and `productowner_code.ipynb`, and a status bar indicating the current cell is 34 of 35.
- Contenido**: The main area contains Python code for developing a class `Txt` to clean text files. The code defines methods to replace newlines and underscores, and to trim whitespace from columns.

```
# Finalmente, metemos todas las funciones dentro de la clase Txt anteriormente creada.
# Primero definimos el nombre de nuestra clase.
class Txt:
    # Creamos el método constructor donde definimos los parámetros que estamos interesados.
    def __init__(self, archivo_txt):
        # Definimos la variable que almacena el contenido del archivo que queremos limpiar.
        self.archivo_txt= archivo_txt

    # Definimos el primer método para quitar saltos de línea(\n), cambiar '<' por 'under' y
    # 'null' por 'NULL', cuyo parámetro será una variable que almacene, en formato lista,
    # el contenido de un .txt.
    def sustituir(self):
        lista_sustituida=[]
        for i in self.archivo_txt:
            saltos=i.replace("\n","")
            menores=saltos.replace("<"," under")
            nules=menores.replace("null","NULL")
            lista_sustituida.append(nules)
        return lista_sustituida

    # Definimos el segundo método para eliminar los espacios innecesarios al principio y al final del
    # contenido de cada columna. El parámetro será una variable que almacene, en formato lista,
    # el contenido de un .txt.
    def limpiar(self):
        lista_espaciol=[]
        for entrada in self.archivo_txt:
            espacio1=entrada.split(";")
            lista_espaciol.append(espacio1)
        lista_espacio2=[]
        lista_espacio3=[]
        for columna in lista_espaciol:
            for elemento in columna:
                espacio2=elemento.strip()
                lista_espacio2.append(espacio2)
        return lista_espacio2
```

- Barra inferior**: Includes icons for main, refresh, back, forward, Live Share, Git Graph, and a status bar indicating "Servidor de Jupyter: local" and "Celda 34 de 35".

Desarrollo de clase

Desarrollo de clase

Desarrollo de clase.

The screenshot shows a Jupyter Notebook interface with the following details:

- EXPLORADOR**: Shows the file structure of the project. The current file is `txt_code.ipynb`.
- EDITORES**: Shows two notebooks: `txt_code.ipynb` and `productowner_code.ipynb`.
- CELLS**: Shows two cells: one for reading and explaining the .txt file structure, and another for cleaning the .txt file.
- CODIGO**: The active cell contains Python code for connecting to a MySQL database and creating a DataFrame from a .txt file.
- MARKDOWN**: A cell for writing explanations.
- EJECUTAR TODO**: A button to run all cells.
- BORRAR RESULTADOS**: A button to clear all results.
- REINICIAR**: A button to restart the kernel.
- VARIABLES**: A button to view variables.
- ESQUEMA**: A button to view the schema.
- BASE (Python 3.9.7)**: The kernel configuration.
- CELLS**: Shows 31 cells in total.
- Servidor de Jupyter: local**: The notebook server information.

```
txt_code.ipynb ● productowner_code.ipynb u ●
# Lectura y explicación de la estructura del archivo .txt > Limpieza del archivo .txt > # En caso de no tener creada la tabla para insertar los datos del fichero txt, procedemos
return

# Definimos el quito método para crear un dataframe de pandas de nuestros datos
# del .txt. y después almacenarlo como un archivo externo.

def exportar_txt_csv (self):
    import os
    print('-----')
    print('La ruta en la que se va a guardar el archivo es:')
    print(os.getcwd())
    import mysql.connector
    import pandas as pd
    from mysql.connector import errorcode
    try:
        cnx = mysql.connector.connect(user='root', password='AlumnaAdalab',
                                      host='127.0.0.1',
                                      database='project1')
        print('-----')
        print('Conexión con la base de datos realizada')
    except mysql.connector.Error as err:
        if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
            print("Ha habido un error al introducir el nombre de usuario o la contraseña")
        elif err.errno == errorcode.ER_BAD_DB_ERROR:
            print("La base de datos no existe")
        else:
            print(err)
    # Realizamos la consulta a la tabla alumnos mediante pandas y creamos el dataframe.
    query = """SELECT * FROM data_txt"""
    df = pd.read_sql_query(query, cnx)

    # Guardamos los datos registrados del dataframe en un fichero csv (separado por comas).
    df.to_csv("data_txt.csv")

    # Cerramos la conexión
    cnx.close()
```

Código en ejecución.

The screenshot shows a Jupyter Notebook interface with a dark theme. The top bar displays the file name "productowner_code.ipynb" and a timer showing "00:00:02". The left sidebar includes sections for "EXPLORADOR", "EDITORES AB...", "ESQUEMA", and a message stating "No se encontró ningún símbolo en el documento 'productowner_code.ipynb'". The main area contains a code cell with the following Python code:

```
# En primer lugar guardamos el contenido del archivo .txt en la variable "file_source".
with open('data_txt.txt','r') as file:
    file_source = file.readlines()

# Finalmente, metemos todas las funciones dentro de la clase Txt anteriormente creada.
# Primero definimos el nombre de nuestra clase.
class Txt:
    # Creamos el método constructor donde definimos los parámetros que estamos interesados.
    def __init__(self, archivo_txt):
        # Definimos la variable que almacena el contenido del archivo que queremos limpiar.
        self.archivo_txt= archivo_txt

    # Definimos el primer método para quitar saltos de línea(\n), cambiar '<' por 'under y
    # 'null' por 'NULL', cuyo parámetro será una variable que almacene, en formato lista,
    # el contenido de un .txt.
    def sustuir(self):
        lista_sustituida=[]
        for i in self.archivo_txt:
            saltos=i.replace("\n","",)
            menores=saltos.replace("<", "under")
            nules=menores.replace("null","NULL")
            lista_sustituida.append(nules)
        return lista_sustituida

    # Definimos el segundo método para eliminar los espacios innecesarios al principio y al final del
    # contenido de cada columna. El parámetro será una variable que almacene, en formato lista,
    # el contenido de un .txt.
    def limpiar(self):
        lista_espacio1=[]
        for entrada in self.archivo_txt:
            espacio1=entrada.split(";")
            lista_espacio1.append(espacio1)
        lista_espacio2=[]
        lista_espacio3=[]
        for columna in lista_espacio1:
            for elemento in columna:
                espacio2=elemento.strip()
```

Datos TXT en CSV

	index_txt	index_sql	q3	q4	q5	q6	q7	q8
0	1	1	Indonesia	Master's degree	Program/Project Manager	20+ years	NULL, SQL, C, C++, Java	Python
1	2	2	Pakistan	Master's degree	Software Engineer	1-3 years	Python, C++, Java	Python
2	3	3	Mexico	Doctoral degree	Research Scientist	20+ years	Python	Python
3	4	4	India	Doctoral degree	Other	under 1 years	Python, C, MATLAB	Python
4	5	5	India	I prefer not to answer	Currently not employed	under 1 years	Python	Python
5	6	6	India	Some college/university study without earning a bachelor's degree	Student	1-3 years	NULL, C++, Java, Javascript	Python
6	7	7	India	Bachelor's degree	Data Scientist	5-10 years	Python	Python
7	8	8	Russia	Bachelor's degree	Currently not employed	3-5 years	Python, SQL	Python
8	9	9	Turkey	I prefer not to answer	Other	1-3 years	Python, SQL	SQL
9	10	10	Australia	Doctoral degree	Other	1-3 years	Python, R, SQL	R
10	11	11	India	Master's degree	Student	under 1 years	Python, R, C++	R
11	12	12	India	Master's degree	Student	under 1 years	Python, MATLAB	Python
12	13	13	Nigeria	Master's degree	Program/Project Manager	5-10 years	Python, SQL	Python
13	14	14	Nigeria	Bachelor's degree	Other	under 1 years	Python	Python
14	15	15	Greece	Doctoral degree	Research Scientist	10-20 years	Python, C, C++, MATLAB	Python
15	16	16	Belgium	Bachelor's degree	Data Analyst	20+ years	Python, SQL	Python
16	17	17	Pakistan	Bachelor's degree	Data Scientist	1-3 years	Python, SQL	Python
17	18	18	Japan	Master's degree	Software Engineer	3-5 years	Python, SQL, C, Java, Javascript	Python
18	19	19	Egypt	Bachelor's degree	Other	under 1 years	NULL, None	R
19	20	20	Singapore	Bachelor's degree	Other	under 1 years	Python	Python
20	21	21	Turkey	Bachelor's degree	Data Scientist	3-5 years	Python, R, SQL, C++	R
21	22	22	Indonesia	Master's degree	Student	1-3 years	NULL, R	R
22	23	23	Brazil	Master's degree	Machine Learning Engineer	20+ years	Python, SQL, C++	SQL
23	24	24	India	Bachelor's degree	Student	1-3 years	Python, R, SQL, C, C++, MATLAB	Python
24	25	25	Poland	Master's degree	Machine Learning Engineer	3-5 years	Python, C++	Python
25	26	26	Brazil	Doctoral degree	Research Scientist	under 1 years	Python, R	Python

Archivo SQL en SQL

Adalab Server

File Edit View Query Database Server Tools Scripting Help

Schemas

SCHEMAS

Filter objects

project1

Tables

data_sql

data_txt

data_xml

Views

Stored Proced

Functions

prueba1

prueba_z

sakila

sys

Object Info

Table: data_sql

Columns:

- index_sql int(11) PK
- q10_part_1 varchar(300)
- q10_part_2 varchar(300)

data_sql 1

data_sql

1 • SELECT * FROM project1.data_sql;

Result Grid

Filter Rows:

Edit: Export/Import: Wrap Cell Content: Fetch rows:

#	index_sql	q10_part_1	q10_part_2	q10_part_3	q10_part_4	q10_part_5	q10_part_6	q10_part_7	q10_part_8	q10_part_9	q10_part_10	q10_part_11
1	1	Kaggle Notebooks	Colab Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
2	2	Kaggle Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
3	3	NULL	Colab Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
4	4	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	Google Col...
5	5	NULL	Colab Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
6	6	Kaggle Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	Google Cloud Notebooks (AI Platform / Ver...	NULL	NULL
7	7	Kaggle Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
8	8	Kaggle Notebooks	Colab Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
9	9	Kaggle Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
10	10	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
11	11	Kaggle Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
12	12	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
13	13	Kaggle Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
14	14	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
15	15	Kaggle Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
16	16	Kaggle Notebooks	Colab Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
17	17	NULL	Colab Notebooks	NULL	NULL	Binder / J...	NULL	NULL	NULL	NULL	NULL	NULL
18	18	Kaggle Notebooks	Colab Notebooks	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
19	19	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Query Completed

Apply Revert

Result Grid

Form Editor

Field Types

Query Stats

Execution Plan

Archivo XML en SQL

The screenshot shows the MySQL Workbench interface with the title "Archivo XML en SQL". The left sidebar displays the database schema for "project1", including tables like "data_sql", "data_txt", and "data_xml". The "data_xml" table is selected. The main area shows the results of the query `SELECT * FROM project1.data_xml;`. The results are presented in a "Result Grid" with the following columns: #, index_xml, time, age, gender, and index_sql. The data consists of 19 rows of XML index information.

#	index_xml	time	age	gender	index_sql
1	1	784	50-54	Man	1
2	2	924	22-24	Man	2
3	3	575	45-49	Man	3
4	4	781	45-49	Man	4
5	5	1020	25-29	Woman	5
6	6	141	18-21	Woman	6
7	7	484	30-34	Man	7
8	8	1744	22-24	Man	8
9	9	655	30-34	Man	9
10	10	1777	40-44	Man	10
11	11	3081	18-21	Woman	11
12	12	1922	18-21	Woman	12
13	13	852	45-49	Man	13
14	14	838	22-24	Man	14
15	15	563	35-39	Man	15
16	16	1315	50-54	Man	16
17	17	479	18-21	Man	17
18	18	249	22-24	Man	18
19	19	650	30-34	Man	19

Table: data_xml
Columns:
index_xml
int(11) AI PK
time
varchar(300)
age
varchar(300)

data_xml 1

Query Completed

Archivo TXT en SQL

Adalab Server

File Edit View Query Database Server Tools Scripting Help

sql_code data_txt SQL File 3* data_txt

Limit to 200 rows

```
1 • SELECT * FROM project1.data_txt;
```

Result Grid Filter Rows: Export/Import: Wrap Cell Content: Fetch rows:

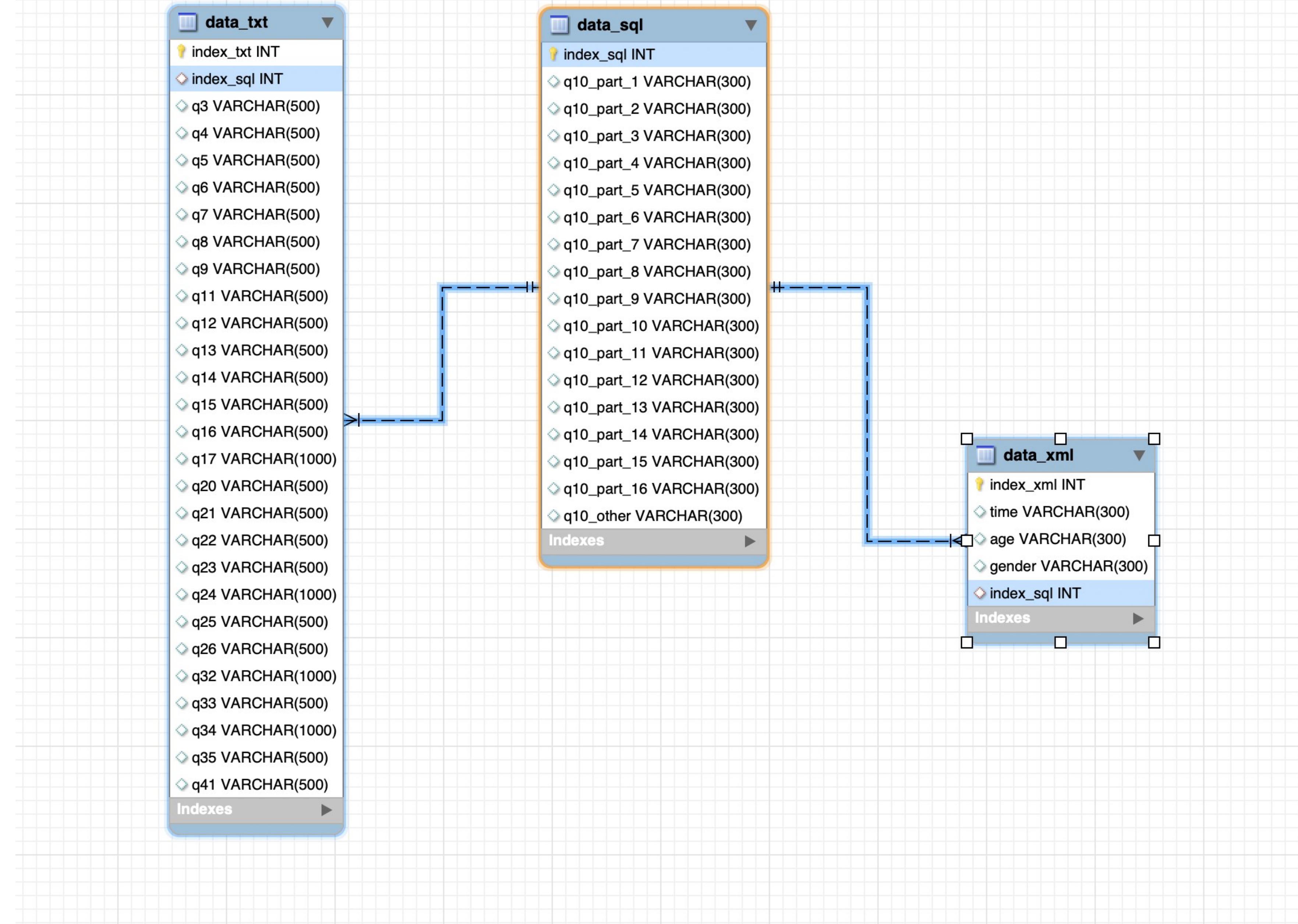
#	index_txt	index_sql	q3	q4	q5	q6	q7	q8	q9	q11
1	1	1	Indonesia	Master's degree	Program/Project Manager	20+ years	null, SQL, C, C++, Java	Python	null, Notepad++, Jupyter Notebook	A cloud computing pl...
2	2	2	Pakistan	Master's degree	Software Engineer	1-3 years	Python, C++, Java	Python	null, PyCharm, Jupyter Notebook, Other	A laptop
3	3	3	Mexico	Doctoral degree	Research Scientist	20+ years	Python	Python	null, Spyder, Jupyter Notebook	A cloud computing pl...
4	4	4	India	Doctoral degree	Other	< 1 years	Python, C, MATLAB	Python	null, Spyder, MATLAB, Jupyter Notebook	A cloud computing pl...
5	5	5	India	I prefer not to answer	Currently not employed	< 1 years	Python	Python	Jupyter (JupyterLab, Jupyter Notebooks, etc...)	A laptop
6	6	6	India	Some college/university study without ear...	Student	1-3 years	null, C++, Java, Javascript	Python	null, Visual Studio, Visual Studio Code (V...	A laptop
7	7	7	India	Bachelor's degree	Data Scientist	5-10 years	Python	Python	null, Jupyter Notebook	A personal computer
8	8	8	Russia	Bachelor's degree	Currently not employed	3-5 years	Python, SQL	Python	null, Other	A cloud computing pl...
9	9	9	Turkey	I prefer not to answer	Other	1-3 years	Python, SQL	SQL	null, Spyder, Jupyter Notebook	A laptop
10	10	10	Australia	Doctoral degree	Other	1-3 years	Python, R, SQL	R	null, RStudio, Jupyter Notebook	A personal computer
11	11	11	India	Master's degree	Student	< 1 years	Python, R, C++	R	null, RStudio, Jupyter Notebook	A laptop
12	12	12	India	Master's degree	Student	< 1 years	Python, MATLAB	Python	null, RStudio, MATLAB, Jupyter Notebook	A laptop
13	13	13	Nigeria	Master's degree	Program/Project Manager	5-10 years	Python, SQL	Python	null, Spyder, Jupyter Notebook	A laptop
14	14	14	Nigeria	Bachelor's degree	Other	< 1 years	Python	Python	null, Visual Studio Code (VSCode)	A laptop
15	15	15	Greece	Doctoral degree	Research Scientist	10-20 ye...	Python, C, C++, MATLAB	Python	null, Spyder	A laptop
16	16	16	Belgium	Bachelor's degree	Data Analyst	20+ years	Python, SQL	Python	Jupyter (JupyterLab, Jupyter Notebooks, etc...)	A laptop
17	17	17	Pakistan	Bachelor's degree	Data Scientist	1-3 years	Python, SQL	Python	Jupyter (JupyterLab, Jupyter Notebooks, etc...)	A laptop
18	18	18	Japan	Master's degree	Software Engineer	3-5 years	Python, SQL, C, Java, Jav...	Python	Jupyter (JupyterLab, Jupyter Notebooks, etc...)	A laptop

data_txt 1

Apply Revert

Action Output

Query Completed



Thank you!

FOLLOW ME ON:

e - m a i l : sonia.ruiz.p31@gmail.com

L i n k e d I n : <https://www.linkedin.com/in/sonia-ruiz-perez/>

G i t H u b : <https://github.com/solkiria>

T a b l e a u : <https://public.tableau.com/app/profile/solkiria>