**FORMATO PARA EL DESARROLLO DE COMPONENTE FORMATIVO**

|  |  |
| --- | --- |
| PROGRAMA DE FORMACIÓN | Visualización de datos con Python |

|  |  |  |  |
| --- | --- | --- | --- |
| COMPETENCIA | 220501115 - Integrar datos según técnicas de visualización y metodologías de análisis. | RESULTADOS DE APRENDIZAJE | 220501115-02 - Aplicar metodologías de visualización de datos de acuerdo con requerimientos. |

|  |  |
| --- | --- |
| NÚMERO DEL COMPONENTE FORMATIVO | CF002 |
| NOMBRE DEL COMPONENTE FORMATIVO | Herramientas de visualización de datos |
| BREVE DESCRIPCIÓN | Este componente formativo aborda aspectos generales y claves de las herramientas de visualización, mismas que han ayudado a interpretar los datos que pretenden guiar tendencias y que, realizando una serie de métodos, permiten sacarles el mejor provecho definiendo objetivos específicos. Estas herramientas se trasladan al lenguaje de programación preferido por la comunidad de desarrolladores (Python), gracias también a las librerías que permiten hacer análisis científico de datos. |
| PALABRAS CLAVE | Ciencia de datos, gráficas, Matplotlib, NumPy, *scripts*. |

|  |  |
| --- | --- |
| ÁREA OCUPACIONAL | Ciencias naturales, aplicadas y relacionadas |
| IDIOMA | Español |

1. **TABLA DE CONTENIDOS**

**Introducción**

1. **Herramientas a utilizar**
2. **Implementación con Matplotlib**
   1. Componentes de una figura
   2. Definición de Pandas
   3. Tipos de entradas para graficar funciones
3. **Simulación de la herramienta mediante caso práctico**
4. **INTRODUCCIÓN**

A medida que la ciencia de datos se introduce cada vez más en la cotidianidad de las predicciones en las empresas, se torna indispensable para el futuro de estas, dándoles sentido a los millones de datos que circulan diariamente por la red, pero, para esto, es necesario el buen uso de las herramientas de visualización de datos.

A continuación, le invitamos a ver explorar el siguiente video, el cual introduce los temas de este componente formativo:

|  |
| --- |
| **DI\_CF02\_0\_Video\_Introduccion** |

1. **DESARROLLO DE CONTENIDOS**
2. **Herramientas a utilizar**

La comunidad de Python es muy grande, por lo tanto, es posible encontrar un gran número de librerías para visualizar datos. Al tener tanta variedad de opciones, a veces se hace realmente difícil determinar cuándo utilizar cada una de ellas.

Por consiguiente, a continuación, se relacionan las más utilizadas por la comunidad Python y que han demostrado ser las librerías más poderosas frente a los temas de interpretación de gráficas:

|  |
| --- |
| **DI\_CF02\_1\_Petañas\_HerramientasDeVisualizacion** |

Cuando se empieza un proyecto de visualización, la duda que se viene a la cabeza es cuál herramienta utilizar para poder realizar un trabajo que satisfaga las necesidades del usuario final. En ese orden de ideas, es bueno tener en cuenta la siguiente guía, que ayudará a la escogencia de la herramienta o librería indicada; no es obligatorio seguirla al pie de la letra, pero será de gran valor a la hora de elegir:

|  |
| --- |
| **DI\_CF02\_1\_InfografiaInteractiva\_HerramientasVisualizacion** |

Como se observa, hay una división en cuatro partes principales, y estas, después, se subdividen en métodos de visualización, que representan las categorías iniciales, como se puede ver a continuación:

* **Distribuciones:** en esta parte, se intenta comprender la distribución de los datos, por lo tanto, se encuentran variables cuantitativas y categóricas; solo depende de las variables y la cantidad de ellas.
* **Comparaciones:** se comparan valores a través de diferentes categorías; los gráficos que se identifican con esta etapa son las barras, cuando se realiza una comparación de elementos y categorías, y los diagramas de puntos y líneas, para las variables cuantitativas.
* **Relaciones:** lo importante en este modelo es comprender la relación que hay entre las variables; su representación gráfica son las gráficas de dispersión.
* **Composiciones:** en este modelo, lo importante es comprender cómo está compuesta una variable y sus relaciones frente a otras, ya sea en una proyección de tiempo o de manera estática. Las gráficas que las representan son las gráficas de barras y las tortas.

1. **Implementación con Matplotlib**

Esta librería se destaca por ser la más utilizada en la comunidad desarrolladora, particularmente, para los científicos de datos. Matplotlib, como ya se ha dicho, no es más que una biblioteca de Python para la visualización de gráficas dinámicas o estáticas. En este apartado, se tratará de explicar cómo es su implementación y cuáles son las gráficas resultantes.

Para instalar e implementar la biblioteca, se corre el comando **pip,** a través de los siguientes pasos:

|  |
| --- |
| **DI\_CF02\_2\_Pasos\_ImplementacionConMatplotlib** |

**2.1 Componentes de una figura**

La forma en que se puede realizar una gráfica puede variar según lo que se requiera representar, por lo tanto, existen diferentes formas con las que se puede representar una situación, siendo muy amplias y útiles, gracias a las librerías especializadas como Matplotlib.

Las gráficas que aportan estas librerías ofrecen información contundente frente a los datos ingresados, es posible categorizarlas bajo la premisa que se tenga para la organización de datos, tal y como se expone a continuación:

* **Al comparar datos**

**Figura 1**

*Gráfica de comparación de datos*

**Gráfico

Descripción generada automáticamente con confianza media**

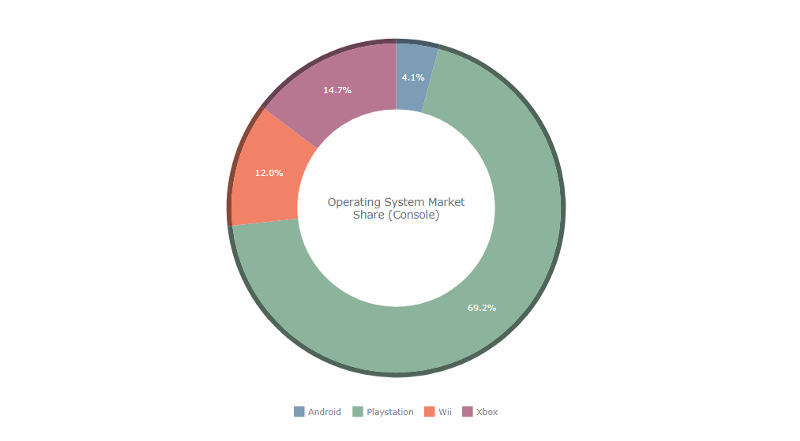
Los siguientes tipos de gráficos son muy útiles y suelen ser, también, frecuentemente usados:

|  |
| --- |
| **DI\_CF02\_2-1\_Slider\_AlCompararDatos** |

* **‌Explorar relaciones / *Part-To-Whole* (de la parte al todo)**

**Figura 2**

*Gráfica de exploración de relaciones*

****

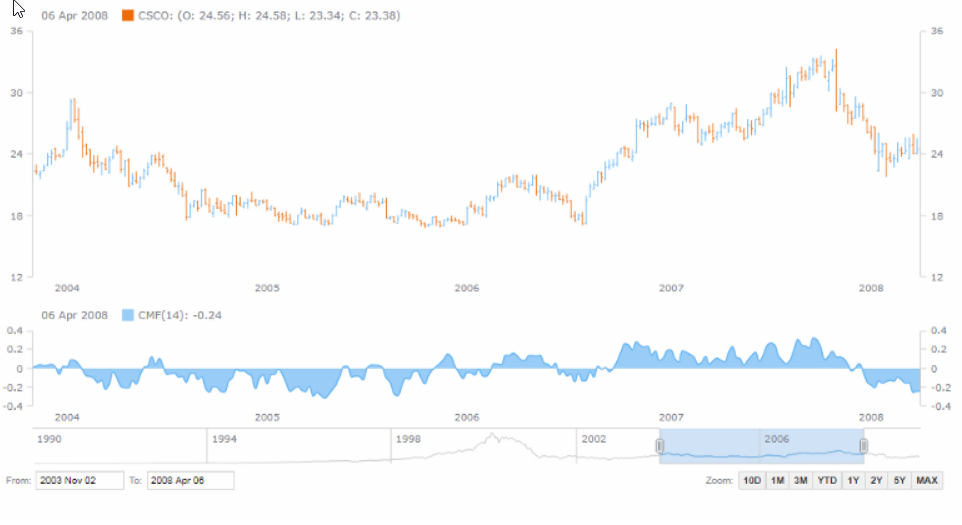
Cuando lo que se quiere es explorar la composición, por ejemplo, de una variable categórica y analizar relaciones del tipo *part-to-whole*, los siguientes tipos de visualizaciones son muy útiles:

|  |
| --- |
| **DI\_CF02\_2-1\_Slider\_ExplorarRelaciones** |

* **Análisis en el tiempo**

**Figura 3**

*Gráfica de análisis en el tiempo*



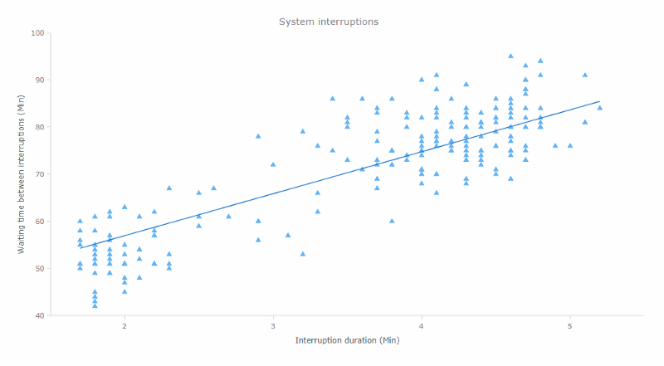
Cuando se busca analizar la relación entre una o más variables con un componente temporal como, por ejemplo, el analizar la evolución de los salarios de Uruguay, respecto del tiempo, los siguientes tipos de visualizaciones son los indicados:

|  |
| --- |
| **DI\_CF02\_2-1\_Slide\_AnálisisEnElTiempo** |

* **Analizar distribución de un *dataset***

**Figura 4**

*Representación gráfica de datos estructurados (dataset)*

****

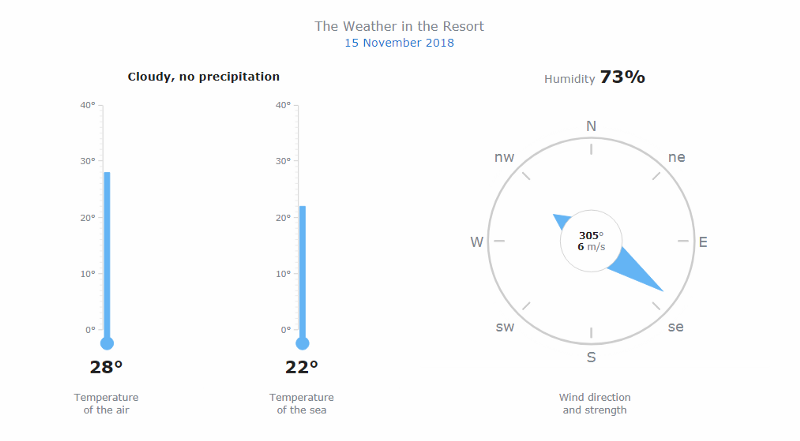
Para analizar la distribución de una variable en un determinado *dataset* y comparar, por ejemplo, con otros *datasets*, encontrando relaciones y correlaciones, los siguientes tipos de gráficos son muy útiles:

|  |
| --- |
| **DI\_CF02\_2-1\_Slide\_AnalizarDistribucionDeUnDataset** |

* **Evaluar *performance* de un dato**

**Figura 5**

*Gráfica de performance*

****

Cuando se quiere evaluar el rendimiento de una variable y comparar el valor real con valores KPI, se suele recurrir a los siguientes tipos de gráficos:

|  |
| --- |
| **DI\_CF02\_2-1\_Slide\_EvaluarPerformance** |

* **Datos geográficos**

**Figura 6**

*Gráfica de performance geográfico*



Muchas veces, la mejor forma de darle sentido a datos geográficos (georreferenciados) es utilizando visualizaciones en un mapa, teniendo en cuenta:

**Mapa de colores (*choropleth map*):** permite identificar diferencias entre áreas geográficas.

**Mapa de puntos (*dot map*):** Proporciona una visualización accesible de la distribución geográfica de los fenómenos en las áreas geográficas en las que se encuentran.

**Mapa de burbujas (*bubble map*):** similar a los gráficos de burbujas, permite agregar una tercera dimensión a la posición de un fenómeno, utilizando el tamaño de una burbuja. Son muy útiles para agregar la información de un mapa de puntos cuando se cuenta con tantos puntos que se hace difícil entender la visualización.

**Mapa de conexiones (*connector map*):** permite representar relaciones entre los distintos puntos del mapa mediante líneas que unen los puntos.

**Mapa de flujos (*flow map*):** útiles para explorar cómo los objetos se mueven entre ubicaciones, en especial cuando interesa mostrar la dirección del flujo.

Hasta este punto, se han presentado diferentes maneras para realizar una gráfica, según la necesidad que se quiera representar, sabiendo que se tiene una gran cantidad de opciones, según el fin que se busque.

|  |
| --- |
| Explore el **Anexo\_1\_EjemplosDeGraficas**, el cual contiene algunos ejemplos, para su afianzamiento en el tema. |

**2.2 Definición de Pandas**

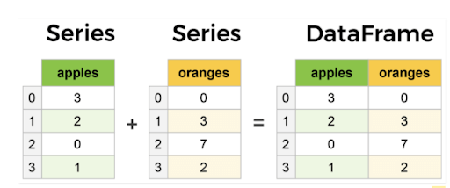
Según Google Colaboratory (2022), “Es una API de análisis de datos en columnas, ideal para manipular y analizar datos de entrada”. Además, muchos marcos de trabajo de AA admiten las estructuras de datos Pandas como entradas. Pandas es muy popular, es una librería por preferencia dentro de la comunidad desarrolladora, es una de las más versátiles entre las de código abierto y más aún cuando se habla de temas como ciencia de datos y *machine learning* por su poderío en las estructuras que ofrece y la facilidad para manejar grandes cantidades de datos.

Sobre pandas, se puede afirmar que:

|  |  |  |
| --- | --- | --- |
| Esta herramienta nace con la necesidad, de los analistas de datos, de unificar esfuerzos en una sola herramienta poderosa y facilitar la labor: analizar, depurar y seleccionar datos. | Hay dos estructuras poderosas dentro de la librería de datos Pandas: Series *y DataFrame*. Las Series hacen referencia a un *array* unidimensional que puede almacenar cualquier tipo de dato. | Los *DataFrame* son bidimensionales, con estructuras que pueden albergar cualquier tipo de dato; a su vez, se pueden interpretar como series. |

**Figura 7**

*Series y DataFrames*



¿Qué se necesita para trabajar con Pandas a partir de ahora? Analice la información presentada en el siguiente video, la cual le indicará lo que necesitará:

|  |
| --- |
| **DI\_CF02\_2-2\_Video\_DefinicionDePandas** |

**2.3 Tipos de entradas para graficar funciones**

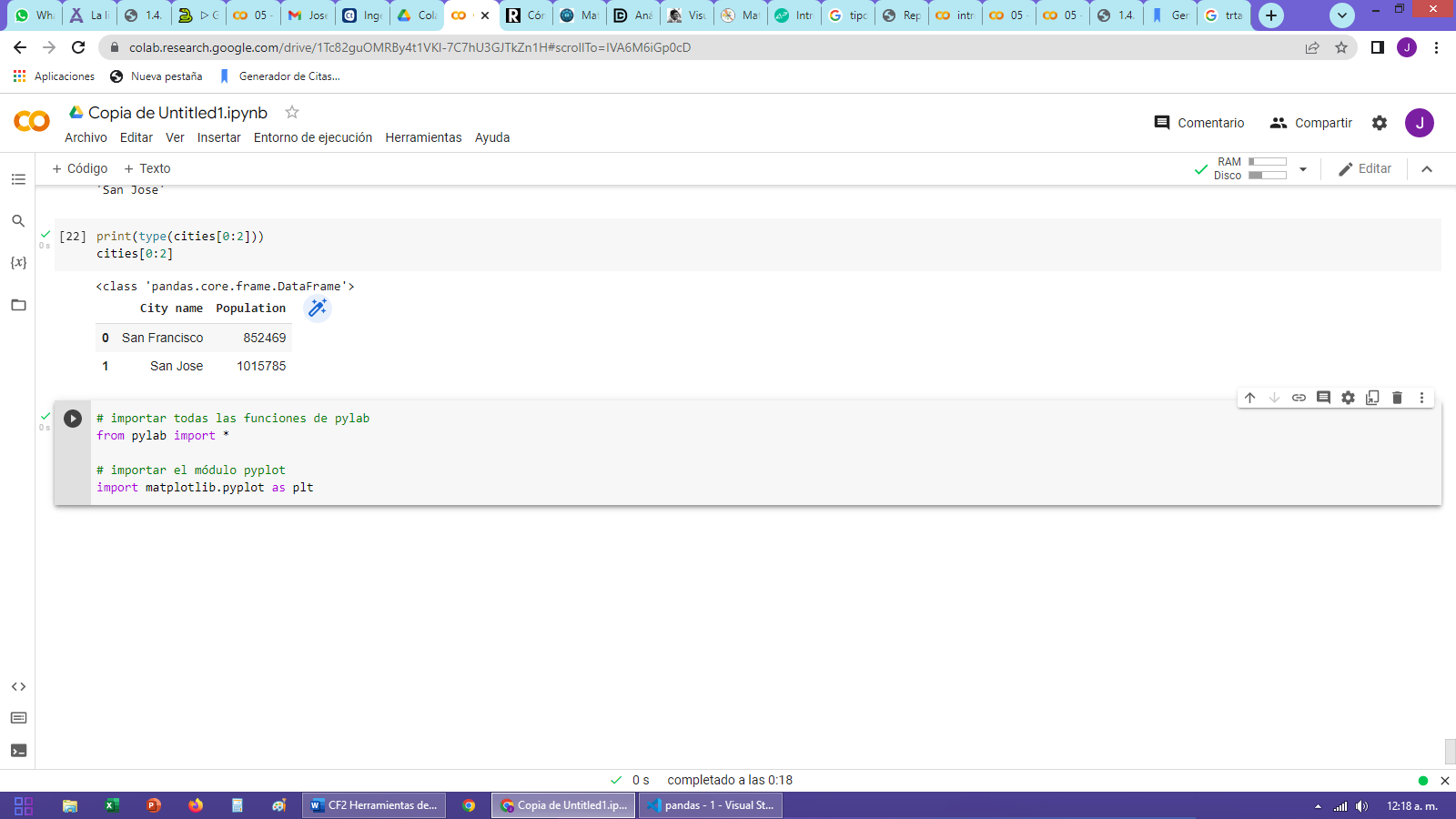
Como se ha venido estudiando en este componente formativo, se observa que existen diversas gráficas que pueden ayudar a la proyección estadística de algo que se quiera visualizar de la mejor manera, utilizando las herramientas correspondientes y las técnicas adecuadas para conseguir el objetivo propuesto; no obstante, las librerías que suministra Python son la mejor opción en el mercado de la ciencia de datos. Por ese motivo, la librería Matplotlib es la más utilizada en la comunidad del *Big Data*, se trata de la librería más grande y más completa de Python, que además de eso contiene dos módulos principales: *pyplot* y *pylab,* ambos con diferencias puntuales.

*Pyplot* es muy versátil y permite crear gráficas automatizando sus ejes cuando son creados. Por otro lado, *pylab* no se queda atrás con sus bondades, ya que trae algo de *pyplot* para la creación de gráficos con la versatilidad de *Numpy* y realizar cálculos con *arrays* con una etiqueta muy particular de *Matlab*.

A continuación, se ilustrará cómo es posible trabajar con estas funciones que marcan la pauta para una estructuración correcta de las gráficas de visualización:

**Figura 8**

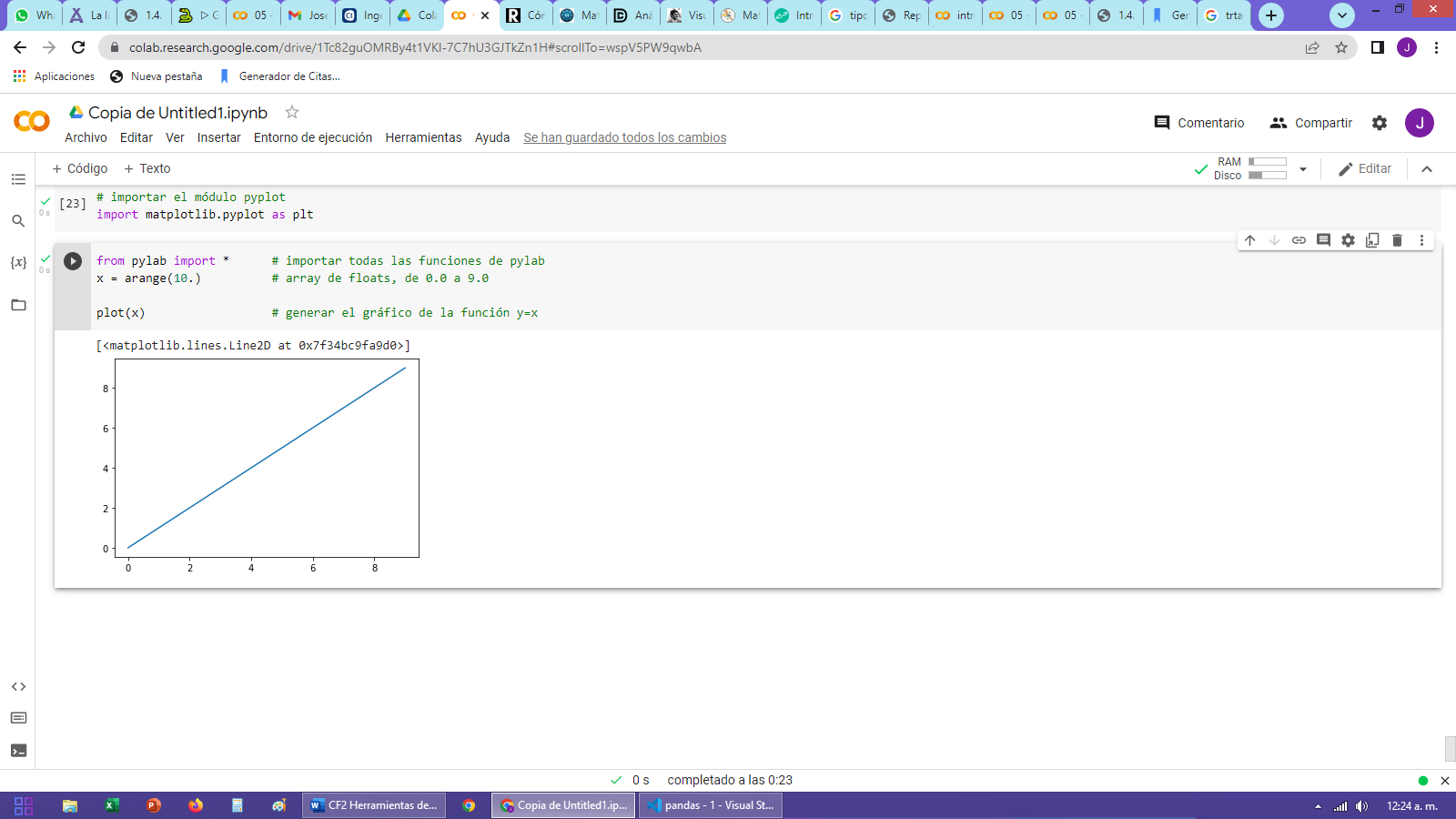
*Comenzando a graficar funciones*



* Luego, se procede a crear un gráfico de la manera más sencilla posible:

**Figura 8**

*Graficación*



1. **Simulación de la herramienta mediante caso práctico**

Para poder trabajar y poner en práctica los conceptos adquiridos, se verán dos ejemplos a continuación, para que, a través de la visualización de estos, pueda identificar lo visto hasta este momento y que sirva, a su vez, para su aplicación, teniendo en cuenta todo lo aprendido.

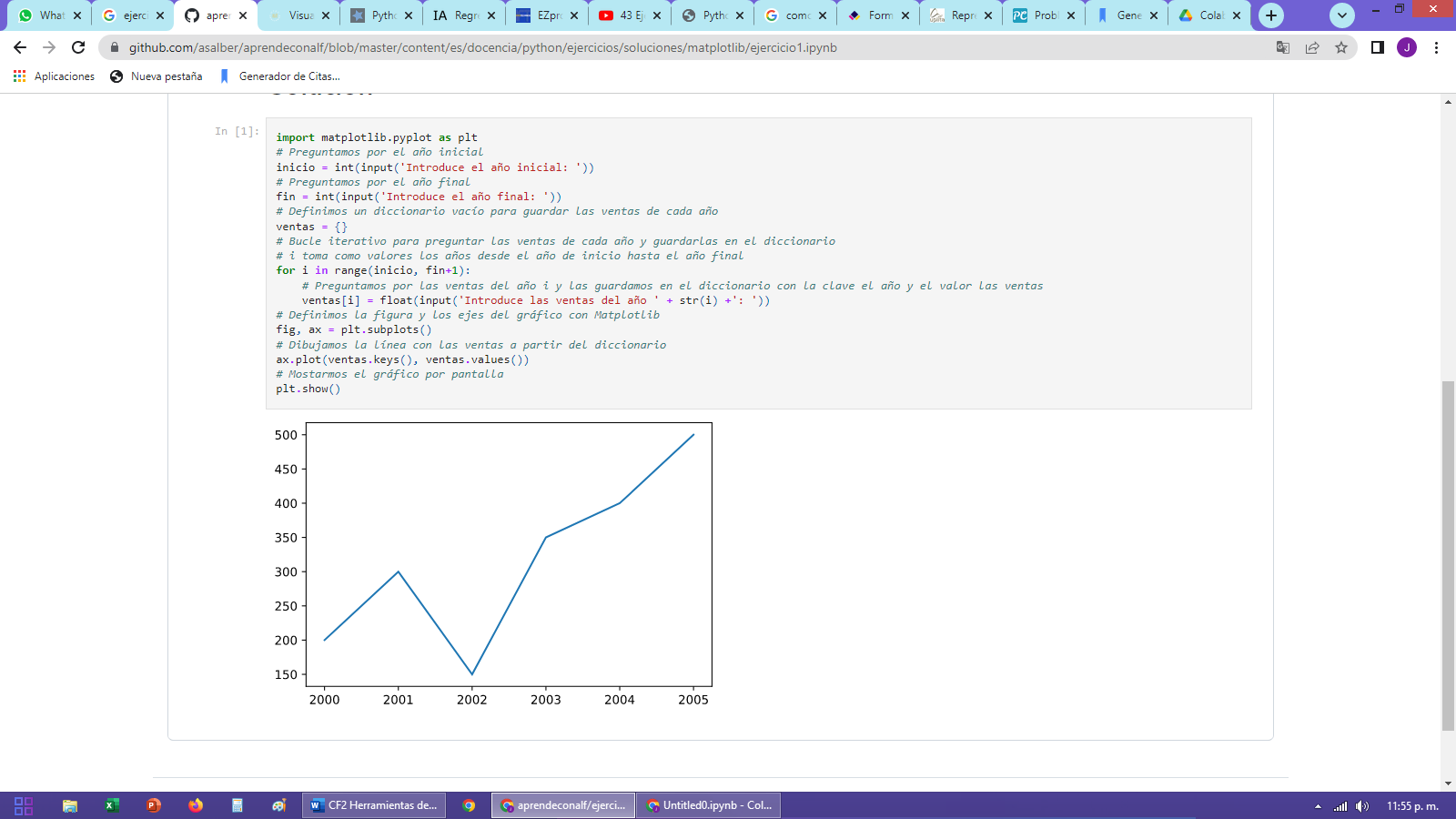
Las aplicaciones son muchas, pero vea las siguientes simulaciones donde es posible ver el código utilizado y el resultado:

* **Ejemplo 1**

**Se digita un código donde se le pregunta a un usuario por las ventas en un intervalo de años y que muestra, mediante un diagrama, el incremento o decremento, dado el caso, de las ventas de la empresa:**

**Figura 9**

*Simulación de caso 1*

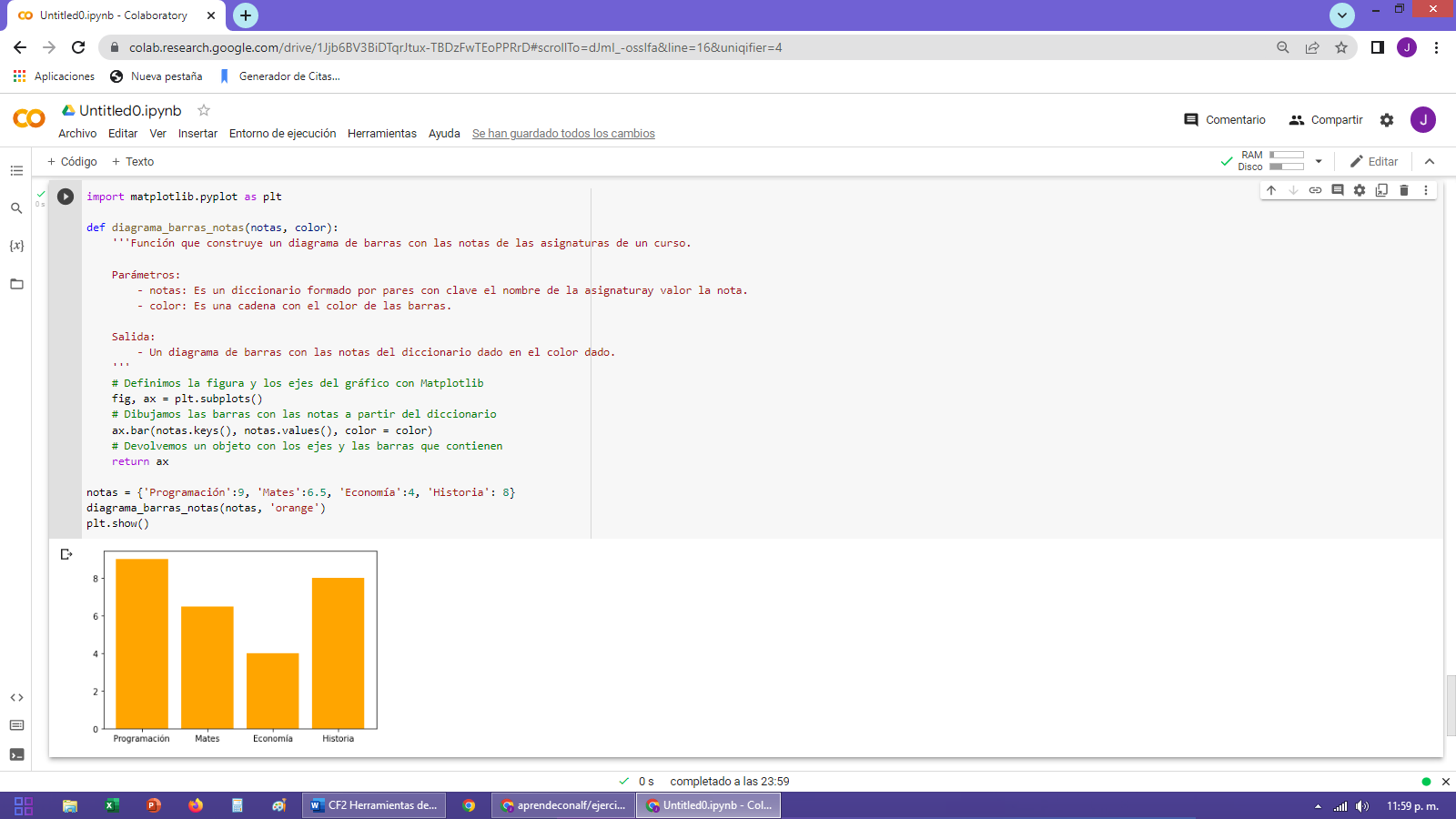


* **Ejemplo 2**

**Se digita un código para obtener un diccionario de notas con las materias de un salón de clase y además una cadena con la palabra de algún color y que retorne un diagrama de barras con las notas y también en el color que se indicó.**

**Figura 10**

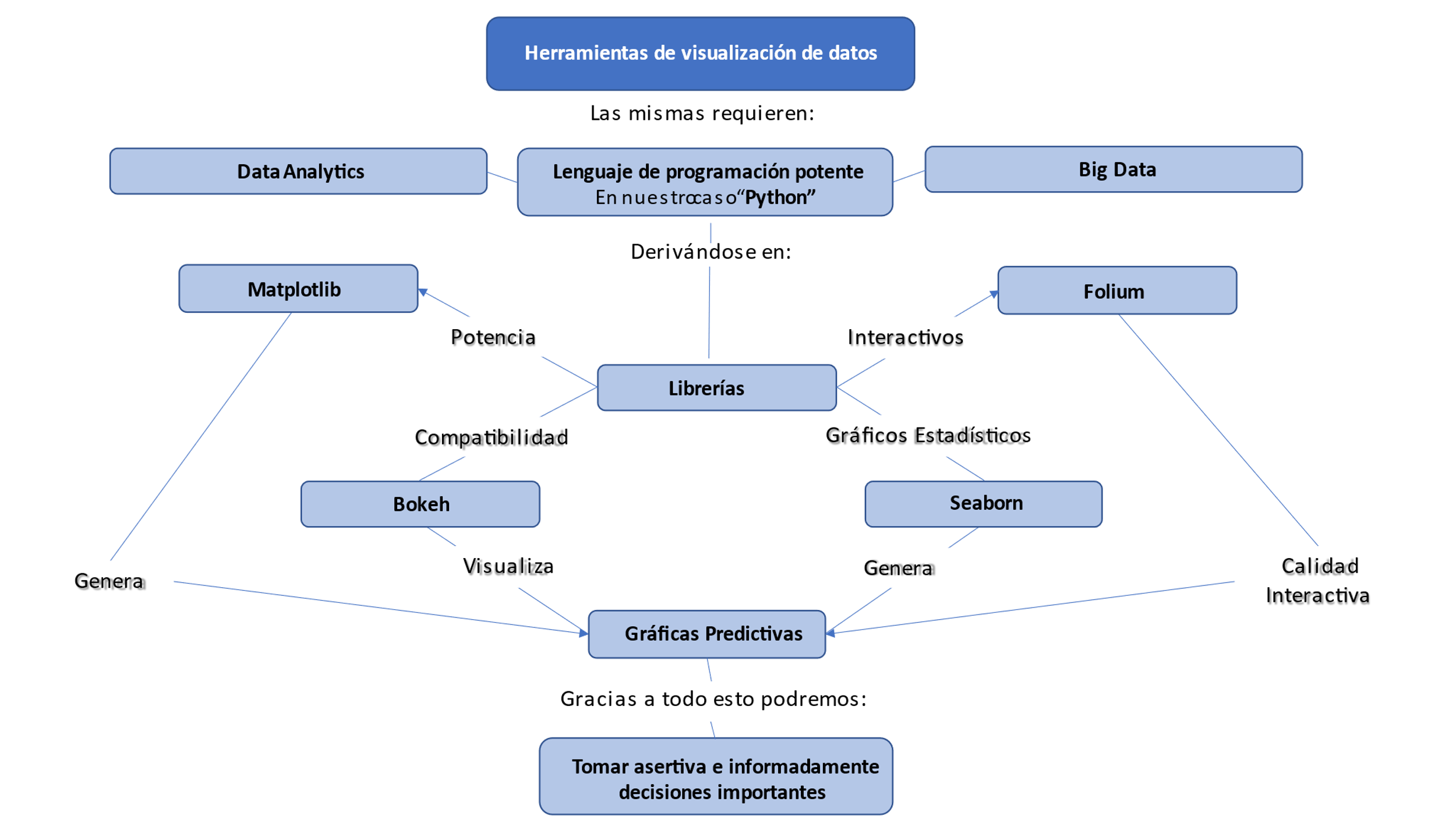
*Simulación de caso 2*



Los datos están cambiando la forma de ver los problemas y las decisiones, las empresas que están en la cúspide del mercado emplean sus habilidades para utilizar todo lo que esté a su alcance para la recolección de estos datos y así poder incursionar con nuevas tendencias en todos los campos; los datos están cambiando la base de las competencias.

1. **SÍNTESIS**

Aquí finaliza el estudio de los temas de este componente formativo; en este punto, analice el esquema que se muestra enseguida y realice su propia síntesis de los contenidos estudiados. ¡**Adelante**!



1. **ACTIVIDAD DIDÁCTICA**

|  |  |
| --- | --- |
| DESCRIPCIÓN DE ACTIVIDAD DIDÁCTICA | |
| Nombre de la Actividad | Visualizando datos |
| Objetivo de la actividad | Identificar especificaciones técnicas logrando la identificación relacionada con las herramientas de visualización de datos con Python. |
| Tipo de actividad sugerida | Relacionar conceptos |
| Archivo de la actividad  (Anexo donde se describe la actividad propuesta) | Anexos:  Actividad\_Didactica\_1 |

1. **MATERIAL COMPLEMENTARIO**

|  |  |  |  |
| --- | --- | --- | --- |
| Tema | Referencia APA del Material | Tipo de material  (Video, capítulo de libro, artículo, otro) | Enlace del Recurso o  Archivo del documento o material |
| 1. Herramientas a utilizar. | Credenciales Alternativas TEC. (2020). *Las gráficas más populares para visualización de datos* [Video]. YouTube. | Video | <https://www.youtube.com/watch?v=OTtN0gu1-UA> |
| 1. Herramientas a utilizar. | Comunicación Numérica. (2021). *Crea Reportes con Impacto, con visualización de datos*  [Video]. YouTube. | Video | <https://www.youtube.com/watch?v=TYp2lJhuE6A> |
| 1. Herramientas a utilizar. | Comunicación Numérica. (2020). *Fundamentos del Análisis de Datos para Toma de Decisiones* [Video]. YouTube. | Video | <https://www.youtube.com/watch?v=qvZxvMWMvDo> |

1. **GLOSARIO**

|  |  |
| --- | --- |
| TÉRMINO | SIGNIFICADO |
| Algoritmo: | conjunto ordenado de operaciones sistemáticas que permite hacer cálculos y hallar soluciones a diferentes tipos de problemas. |
| Centroide: | es la ubicación real o imaginaria que representa el centro del grupo. Cada punto de datos se asigna a cada uno de los grupos mediante la reducción de la suma de cuadrados en el grupo. |
| *Dataset:* | un conjunto de datos tabulados en cualquier sistema de almacenamiento de datos estructurados. |
| *Google Colaboratory:* | *Colaboratory* o "*Colab*", para abreviar, es un producto de *Google Research*. Permite a cualquier usuario escribir y ejecutar código arbitrario de Python en el navegador. Es especialmente adecuado para tareas de aprendizaje automático, análisis de datos y educación. |
| *IPython:* | es un *shell* interactivo que añade funcionalidades extra al modo interactivo incluido con *Python*, como resaltado de líneas y errores mediante colores, una sintaxis adicional para el *shell*, autocompletado mediante tabulador de variables, módulos y atributos. |
| *Jupyter:* | *Jupyter Notebook* es una aplicación cliente-servidor lanzada en 2015 por la organización sin ánimo de lucro Proyecto *Jupyter*. Permite crear y compartir documentos web en formato JSON que siguen un esquema versionado y una lista ordenada de celdas de entrada y de salida.  Estas celdas albergan, entre otras cosas, código, texto (en formato Markdown), fórmulas matemáticas y ecuaciones, o también contenido multimedia (*Rich Media*). El programa se ejecuta desde la aplicación web cliente que funciona en cualquier navegador estándar. |
| *Machine Learning:* | es una forma de IA que permite a un sistema aprender de los datos en lugar de aprender mediante la programación explícita. Sin embargo, *machine learning* no es un proceso sencillo.  Conforme el algoritmo ingiere datos de entrenamiento, es posible producir modelos más precisos basados en datos. Un modelo de *machine learning* es la salida de información que se genera cuando entrena su algoritmo de *machine learning* con datos. Después del entrenamiento, al proporcionar un modelo con una entrada, se le dará una salida. |

1. **REFERENCIAS BIBLIOGRÁFICAS**

Aprende IA. (2020). *Aprendizaje no Supervisado*. <https://aprendeia.com/aprendizaje-no-supervisado-machine-learning/>

Chacón, J. (2021). *Introducción a Pandas, la librería de Python para trabajar con datos.* Profile.<https://profile.es/blog/pandas-python/>

De la Fuente, Ó. (2019). *Google Colab: Python y Machine Learning en la nube.* Adictos al trabajo*.* <https://www.adictosaltrabajo.com/2019/06/04/google-colab-python-y-machine-learning-en-la-nube/>

IDATHA. (s. f.). *05 - DS - Visualizacion de Datos - Python.ipynb.* <https://colab.research.google.com/github/efviodo/idatha-data-science-course/blob/master/notebooks/05%20-%20DS%20-%20Visualizacion%20de%20Datos%20-%20Python.ipynb#scrollTo=RQftWD90aHmb>

Kumar, A. (2022). *Tensor Explained with Python Numpy Examples.* Data Analytics*.* <https://vitalflux.com/tensor-explained-with-python-numpy-examples/>

Mariños, J.(2015). *Listas en Python.* DevCode. <https://devcode.la/tutoriales/listas-python/>

Na8. (2019). *Análisis Exploratorio de Datos con Pandas en Python*. Aprende Machine Learning. <https://www.aprendemachinelearning.com/analisis-exploratorio-de-datos-pandas-python/>

Python Software Foundation. (s. f.). *5. Estructuras de datos.* Python.<https://docs.python.org/es/3/tutorial/datastructures.html#more-on-lists>

Rodríguez, D. (2020). *Formatos condicionales en Pandas.* Analytics Lane. [*https://www.analyticslane.com/2020/04/17/formatos-condicionales-en-pandas/*](https://www.analyticslane.com/2020/04/17/formatos-condicionales-en-pandas/)

Tableau Software. (2019). *¿Qué es la visualización de datos?* SalesForce. <https://www.tableau.com/es-mx/learn/articles/data-visualization>

1. **CONTROL DEL DOCUMENTO**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Nombre | Cargo | Dependencia  *(Para el SENA indicar Regional y Centro de Formación)* | Fecha |
| Autor (es) | José Luis Bastidas Pérez | Experto Temático | Regional Cauca - Centro de Teleinformática y Producción Industrial | Julio de 2022 |
| Zvi Daniel Grosman Landáez | Diseñador Instruccional | Regional Distrito Capital – Centro de Gestión Industrial | Julio de 2022 |
| Andrés Felipe Velandia Espitia | Asesor Metodológico | Regional Distrito Capital - Centro de Diseño y Metrología | Julio de 2022 |
| Darío González | Corrector de Estilo | Regional Distrito Capital - Centro de Diseño y Metrología | Julio de 2022 |
| Fabián Leonardo Correa Díaz | Diseñador Instruccional | Regional Santander – Centro Industrial del Diseño y la Manufactura | Agosto de 2023 |
| Rafael Neftalí Lizcano Reyes | Responsable Equipo Desarrollo Curricular | Regional Santander - Centro Industrial del Diseño y la Manufactura | Julio de2022 |

1. **CONTROL DE CAMBIOS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Nombre | Cargo | Dependencia | Fecha | Razón del cambio |
| Autor(es) |  |  |  |  |  |