

Universidad Nacional Abierta y a Distancia
Vicerrectoría Académica y de Investigación
Curso: Programación para el Análisis de Datos
Código: 203008069

Guía de actividades y rúbrica de evaluación – Tarea 1 -
Principios básicos de programación científica.

1. Descripción de la actividad

Tipo de actividad: Individual	
Momento de la evaluación: Intermedia en la unidad 1	
Puntaje máximo de la actividad: 115 puntos	
La actividad inicia el: jueves, 22 de febrero de 2024	La actividad finaliza el: domingo, 17 de marzo de 2024
Con esta actividad se espera conseguir el siguiente resultado de aprendizaje: Resultado de aprendizaje 2: Implementar diferentes funciones, paquetes y módulos para la computación matemática y ciencia de datos.	
La actividad consiste en: Realizar los siguientes tres (3) ejercicios en el entorno dinámico de programación Jupyter Notebook: <u>Ejercicio 1: Estructuras de Datos y DataFrames.</u> Para el desarrollo de este ejercicio es necesario que revise en el Entorno de Aprendizaje (Unidad 1 - Fundamentos de programación científica con Python - Contenidos y referentes bibliográficos), las siguientes referencias: <ul style="list-style-type: none"> • Alex Galea. (2018). <u><i>Applied Data Science with Python and Jupyter: Use Powerful Industry-standard Tools to Unlock New, Actionable Insights From Your Data: Vol. 1st edition.</i></u> Packt Publishing. (pp. 22-28). • Boschetti, A., & Massaron, L. (2016). <i>Python Data Science Essentials - Second Edition (Vol. 0002)</i>. Packt Publishing. (pp. 63-78). 	

- Samir Madhavan. (2015). *Mastering Python for Data Science: Explore the World of Data Science Through Python and Learn How to Make Sense of Data*. Packt Publishing. (pp. 7-18).
- Vargas, M. (2022). *Carga de datos en Python desde un archivo Excel*. UNAD. [OVI]

Descripción del ejercicio:

Desarrolle códigos para ejecutar las siguientes tareas:

- Cargue en la variable `df1` el conjunto de datos que se presentan en el archivo Excel, Anexo 1 - Base de Datos Excel.xlsx, que se encuentra en la carpeta de la Unidad 1 - Tarea 1 - Fundamentos de programación científica con Python. Tenga en cuenta la hoja, encabezado y celdas en las que comienzan los datos. Los índices del dataframe deben ser los valores en primera columna de la tabla de datos.
- Cargue en la variable `df2` el conjunto de datos que se presentan en el archivo CSV, Anexo 2 - Base de Datos CSV.csv, que se encuentra en la carpeta de la Unidad 1 - Tarea 1 - Fundamentos de programación científica con Python. Tenga en cuenta el tipo de separador y los encabezados de las columnas.
- Cargue los datos *Breast Cancer Wisconsin Dataset* que se encuentran en la variable `load_breast_cancer` del paquete `sklearn`, en un dataframe de pandas asignado a la variable `df3`.
- Construya **un Panel** con los tres (3) dataframes cargados anteriormente: `df1`, `df2` y `df3`. Para construir este panel, cree un diccionario llamado `panel` que en sus keys tenga los nombres de los conjuntos de datos y los valores sean los dataframes.
- Programe una función que extraiga todas las listas de los encabezados de columnas que se encuentran en un Panel y úsela para extraer las listas de columnas del Panel del punto anterior. Imprima los resultados de forma que se diferencien los encabezados entre los dataframes en el Panel.
- Programe una función que imprima la cantidad de elementos que tienen las bases de datos en un Panel y úsela con el Panel del punto anterior. Imprima los resultados de forma que se diferencien los tamaños de los diferentes dataframes en el Panel.

- Filtre el dataframe df1 para obtener las muestras de la empresa TAC. Determine cuantos elementos tiene dicho filtro.
- Determine cuantos elementos nulos hay en cada una de las columnas del dataframe df2.

El archivo de Jupyter notebook (.ipynb) presentado debe contar con la siguiente estructura:

- Portada como se diseñó en la Tarea 0.
- Presentación del ejercicio en una celda markdown.
- Una revisión teórica de Estructuras de Datos y DataFrames. en celdas markdown. Se debe referenciar con normas APA las fuentes de consulta. En esta revisión debe presentar las definiciones de Series, DataFrame y Panel.
- El desarrollo de los códigos solicitados con sus resultados.
- Conclusiones del ejercicio en una celda markdown
- La bibliografía en formato APA en una celda markdown.

Condiciones de entrega:

- Debe ser entregado un archivo en extensión .ipynb nombrado de la siguiente manera en el foro colaborativo:

G##_NombreEstudiante_Ej1.ipynb

Donde:

- *G##_*: es el número del grupo al que pertenecen
- *NombreEstudiante* es el nombre del estudiante tal como aparece en el foro.

Por ejemplo:

G21_MiguelAngelVargasValencia_Ej1.ipynb

- No se acepta el enlace de Google Colab u otros como Word o PDF.

Ejercicio 2: Arreglos, vectores, matices y operadores matemáticos

Para el desarrollo de este ejercicio, es necesario que revise en el Entorno de Aprendizaje (Unidad 1 - Fundamentos de programación científica con Python - Contenidos y referentes bibliográficos), las siguientes referencias:

- Thakur, A. (2016). Python: Real-World Data Science. Packt Publishing. (pp. 491 - 508).

Descripción del ejercicio:

Desarrolle códigos para ejecutar las siguientes tareas:

- Construya la siguiente matriz como un arreglo de numpy:

$$A = \begin{bmatrix} 13 & 1 \\ 14 & 1 \\ 15 & 1 \\ 16 & 1 \\ 17 & 1 \\ 18 & 1 \end{bmatrix}$$

- Construya el siguiente vector como arreglo de numpy:

$$b = \begin{bmatrix} 16 \\ 24 \\ 45 \\ 57 \\ 75 \\ 102 \end{bmatrix}$$

- Calcule el siguiente vector x , usando operaciones de matrices (*dot*, *transpose*, *inv*) de *numpy.linalg*:

$$x = (A^T A)^{-1} A^T b$$

y calcule $\|Ax - b\|$ utilizando la función *numpy.linalg.norm*.

- Solucione el siguiente sistema de ecuaciones, mediante la función *numpy.linalg.solve* (de ser necesario, investigue la función en la documentación de numpy):

$$(A^T A)x = A^T b$$

y calcule $\|Ax - b\|$ utilizando la función *numpy.linalg.norm*.

- Solucione el siguiente sistema de ecuaciones, mediante la función *numpy.linalg.lstsq* (de ser necesario, investigue la

función en la documentación de numpy):

$$Ax = b$$

y calcule $\|Ax - b\|$ utilizando la función `numpy.linalg.norm`.

- Compare todos los resultados obtenidos y anexe una discusión en el análisis de resultados.

Construya un archivo Jupyter notebook (.ipynb) con la siguiente estructura:

- Portada como se diseñó en la Tarea 0.
- Presentación del ejercicio en una celda markdown.
- Una revisión teórica de Arreglos, vectores, matrices y operadores matemáticos en celdas markdown y utilizando el código Latex para las ecuaciones o fórmulas. Se debe referenciar con normas APA las fuentes de consulta.
- El desarrollo de los códigos solicitados con sus resultados.
- Análisis de los resultados.
- Realice un análisis de resultados obtenidos.
- Conclusiones del ejercicio en una celda markdown
- La bibliografía en formato APA en una celda markdown.

Condiciones de entrega:

- Debe ser entregado un archivo en extensión .ipynb nombrado de la siguiente manera en el foro colaborativo:

G##_NombreEstudiante_Ej2.ipynb

Donde:

- *G##_*: es el número del grupo al que pertenecen
- *NombreEstudiante* es el nombre del estudiante tal como aparece en el foro.

Por ejemplo:

G21_MiguelAngelVargasValencia_Ej2.ipynb

- No se acepta el enlace de Google Colab u otros como Word o PDF.

Ejercicio 3: Gráficos científicos

Para el desarrollo de este ejercicio, es necesario que revise en el Entorno de Aprendizaje (Unidad 1 - Fundamentos de programación científica con Python - Contenidos y referentes bibliográficos), las siguientes referencias:

- Kane, F. (2017). Hands-On Data Science and Python Machine Learning. Packt Publishing. (pp. 56-62).
- Subramanian, G. (2015). *Python Data Science Cookbook*. Packt Publishing. (pp. 65-74).

Descripción del ejercicio.

Desarrolle códigos para generar y visualizar los siguientes gráficos:

Gráfico 1: Esta visualización debe cumplir con las siguientes condiciones:

- Mostrar los puntos de dispersión del emparejamiento las siguientes listas:
 $x = [13, 14, 15, 16, 17, 18]$, $y = [16, 24, 45, 57, 75, 102]$
- Mostrar la función lineal $f(x) = 17x - 210.3333$ en el dominio $[13, 18]$.
- Mostrar la función exponencial $g(x) = 0.1452 e^{(0.369x)}$ en el dominio $[13, 18]$.
- Mostrar la función logarítmica $h(x) = 259.49 \ln(x) - 656.46$ en el dominio $[13, 18]$.
- Los objetos mostrados deben tener colores diferentes.
- La línea de la función logarítmica debe ser *punteada*.
- En los ejes deben visualizarse los números.
- Asigne un nombre (label) a cada objeto e imprima la legenda que los identifique en la gráfica.
- Asigne nombres a los ejes de la gráfica.
- Asigne un título a la gráfica.
- Tenga en cuenta que los nombres que decida utilizar le serán de ayuda para presentar las conclusiones que usted obtenga del ejercicio.

Gráfico 2: Construya los códigos necesarios para:

- Generar 4 listas aleatorias de 700 muestras (revise la documentación de *numpy.random*):
 - x1 con distribución normal de media 5 y desviación estándar 0.1.
 - y1 con distribución normal de media 7 y desviación estándar 0.1.

- x_2 con distribución uniforme entre el valor mínimo de x_1 y el valor máximo de x_1 .
- y_2 con distribución uniforme entre el valor mínimo de y_1 y el valor máximo de y_1 .
- Graficar en un mismo plot nubes de dispersión de los pares de puntos que se forman con (x_1, y_1) y (x_2, y_2) .
- La nube de puntos (x_2, y_2) debe visualizarse sobre la nube de (x_1, y_1) .
- Anexe los títulos y legendas necesarios para poder citar y explicar realizar el análisis del gráfico obtenido y las conclusiones.

Gráficos 3: Genere las siguientes dos visualizaciones que se proponen a continuación:

- Cargue en la variable `df1` el conjunto de datos que se presentan en el archivo Excel, Anexo 1 - Base de Datos Excel.xlsx, que se encuentra en la carpeta de la Tarea 1 - Fundamentos de programación científica con Python.
- En una misma gráfica presente 2 histogramas de las variables *Tiempo motor ON* y *Tiempo en Movimiento*, respectivamente. Deben ser lo suficientemente transparentes, para poder ver ambas al tiempo. (Revise la documentación de la función `matplotlib.pyplot.hist`).
- En otra gráfica, genere el gráfico de dispersión que se obtiene de emparejar los valores de *Tiempo motor ON* y *Tiempo en Movimiento*.
- Anexe los títulos y legendas necesarios en ambos gráficos para poder citar y explicar el análisis de los gráficos obtenidos y las conclusiones.

Genere un archivo Jupyter notebook (.ipynb) que cuente con la siguiente estructura:

- Portada como se diseñó en la Tarea 0.
- Presentación del ejercicio en una celda markdown.
- Una revisión teórica de Gráficos científicos en celdas markdown y utilizando el código Latex para las ecuaciones o fórmulas. Se debe referenciar con normas APA las fuentes de consulta.

- Para cada gráfico:
 - El desarrollo de los códigos para generar los gráficos solicitados.
 - Descripción y análisis de los resultados obtenidos.
- Conclusiones del ejercicio en una celda markdown
- La bibliografía en formato APA en una celda markdown.

Condiciones de entrega:

- Debe ser entregado un archivo en extensión .ipynb nombrado de la siguiente manera en el foro colaborativo:

G##_NombreEstudiante_Ej3.ipynb

Donde:

- *G##_*: es el número del grupo al que pertenecen
- *NombreEstudiante* es el nombre del estudiante tal como aparece en el foro.

Por ejemplo:

G21_MiguelAngelVargasValencia_Ej3.ipynb

- No se acepta el enlace de Google Colab u otros como Word o PDF.
- Finalmente, después de realizar las correcciones atendidas en el foro de discusión, genere un archivo comprimiendo los tres (3) documentos. Dicho archivo debe ser nombrado con la siguiente estructura: *G##_NombreEstudiante_Tarea1.zip*.

Para el desarrollo de la actividad tenga en cuenta que:

En el entorno de Información inicial debe: Revisar la agenda del curso y debe tener en cuenta las fechas de inicio y finalización de la actividad.

En el entorno de Aprendizaje debe: Realizar la lectura de las referencias bibliográficas correspondientes a la Unidad 1 y participar semanalmente con aportes significativos en el foro de la Tarea 1 – Fundamentos de programación científica con Python.

En el entorno de Evaluación debe: Entregar un archivo comprimido en formato .zip que contendrá los archivos en formato (*.ipynb) con cada uno de los ejercicios desarrollados el cual debe ser nombrado con la siguiente estructura: *G##_NombreEstudiante_Tarea1.zip*.

Evidencias individuales:

Las evidencias individuales para entregar son:

- El desarrollo de los dos ejercicios de la actividad; los cuales tendrá que realizar de forma individual en documentos con extensión ipynb, como se solicita en las condiciones de entrega y los aportes significativos que realizará en el foro sobre su propio avance y apoyo a sus compañeros.
- En el Entorno de Evaluación - Tarea 1 - Principios básicos de programación científica, subir un único archivo que cumpla con las condiciones de entrega y que contenga las correcciones y sugerencias que el tutor realizó previamente en el foro.

Evidencias grupales:

En esta actividad no se requieren evidencias grupales.

2. Lineamientos generales para la elaboración de las evidencias a entregar.

Para evidencias elaboradas **independientemente**, tenga en cuenta las siguientes orientaciones

1. Realice un reconocimiento general del curso y de cada uno de los entornos antes de abordar el desarrollo de las actividades.
2. Identifique los recursos y referentes de la unidad a la que corresponde la actividad.
3. Intervenga en el foro de discusión aplicando las normas de netiqueta Virtual, evidenciando siempre respeto por las ideas de sus compañeros y del cuerpo docente.
4. Antes de entregar el producto solicitado revise que cumpla con todos los requerimientos que se señalaron en esta guía de actividades, rúbrica de evaluación y por parte del tutor en el foro de discusión:
 - Ejercicios desarrollados haciendo uso correcto de la edición de texto en las celdas markdown con correcta escritura en la sintaxis matemática.
 - Estructura del documento de cada ejercicio:
 - Portada.
 - Introducción.

- Desarrollo de Ejercicios.
- Conclusiones.
- Referencias.
- Cada ejercicio tendrá un documento que deberá ser nombrado con la siguiente estructura:
G##_NombreEstudiante_Ej#.ipynb
- Entregar en el entorno de evaluación un archivo final comprimiendo los tres (3) archivos de los ejercicios en formato .zip nombrado con la siguiente estructura:
G##_NombreEstudiante_Tarea1.zip.

5. No cometa fraudes, ni plagios ni actos que atenten contra el normal desarrollo académico de las actividades.

Tenga en cuenta que todos los productos escritos individuales o grupales deben cumplir con las normas de ortografía y con las condiciones de presentación que se hayan definido.

En cuanto al uso de referencias considere que el producto de esta actividad debe cumplir con las normas **APA**

En cualquier caso, cumpla con las normas de referenciación y evite el plagio académico, para ello puede apoyarse revisando sus productos escritos mediante la herramienta Turnitin que encuentra en el campus virtual.

Tenga presente que en el acuerdo 029 del 13 de diciembre de 2013, artículo 99, se considera como faltas que atentan contra el orden académico, entre otras, las siguientes: literal e) "El plagiar, es decir, presentar como de su propia autoría la totalidad o parte de una obra, trabajo, documento o invención realizado por otra persona. Implica también el uso de citas o referencias faltas, o proponer citad donde no haya coincidencia entre ella y la referencia" y liberal f) "El reproducir, o copiar con fines de lucro, materiales educativos o resultados de productos de investigación, que cuentan con derechos intelectuales reservados para la Universidad"

Las sanciones académicas a las que se enfrentará el estudiante son las siguientes:

- a) En los casos de fraude académico demostrado en el trabajo académico o evaluación respectiva, la calificación que se impondrá será de cero puntos sin perjuicio de la sanción disciplinaria correspondiente.
- b) En los casos relacionados con plagio demostrado en el trabajo académico cualquiera sea su naturaleza, la calificación que se impondrá

será de cero puntos, sin perjuicio de la sanción disciplinaria correspondiente.

3. Formato de Rúbrica de evaluación

Tipo de actividad: Individual	
Momento de la evaluación: Intermedia en la unidad 1	
La máxima puntuación posible es de 115 puntos	
Primer criterio de evaluación: Del procedimiento: Desarrollo del Ejercicio 1 de Estructuras de Datos y DataFrames.	Nivel alto: El estudiante realiza adecuadamente el ejercicio 1 de forma correcta y como fue solicitado. Si su trabajo se encuentra en este nivel puede obtener entre 20 puntos y 25 puntos Nivel Medio: El estudiante realiza de forma parcial, o con errores, el ejercicio de programación, o no se ciñe a lo solicitado Si su trabajo se encuentra en este nivel puede obtener entre 10 puntos y 19 puntos Nivel bajo: El estudiante no desarrolla, o desarrolla de forma incorrecta, el ejercicio de programación solicitado. Si su trabajo se encuentra en este nivel puede obtener entre 0 puntos y 9 puntos
Segundo criterio de evaluación: Del procedimiento: Desarrollo del Ejercicio 2 de Arreglos, vectores, matrices y operadores matemáticos.	Nivel alto: El estudiante realiza adecuadamente el ejercicio 2 de forma correcta y como fue solicitado. Si su trabajo se encuentra en este nivel puede obtener entre 20 puntos y 25 puntos Nivel Medio: El estudiante realiza de forma parcial, o con errores, el ejercicio de programación, o no se ciñe a lo solicitado Si su trabajo se encuentra en este nivel puede obtener entre 10 puntos y 19 puntos Nivel bajo: El estudiante no desarrolla, o desarrolla de forma incorrecta, el ejercicio de programación solicitado. Si su trabajo se encuentra en este nivel puede obtener entre 0 puntos y 9 puntos

<p>Tercer criterio de evaluación:</p> <p>Del procedimiento: Desarrollo del Ejercicio 3 de Gráficos científicos.</p> <p>Este criterio representa 25 puntos del total de 115 puntos de la actividad</p>	<p>Nivel alto: El estudiante realiza adecuadamente el ejercicio 3 de forma correcta y como fue solicitado. Si su trabajo se encuentra en este nivel puede obtener entre 20 puntos y 25 puntos</p> <p>Nivel Medio: El estudiante realiza de forma parcial, o con errores, el ejercicio de programación, o no se ciñe a lo solicitado Si su trabajo se encuentra en este nivel puede obtener entre 10 puntos y 19 puntos</p> <p>Nivel bajo: El estudiante no desarrolla, o desarrolla de forma incorrecta, el ejercicio de programación solicitado. Si su trabajo se encuentra en este nivel puede obtener entre 0 puntos y 9 puntos</p>
<p>Cuarto criterio de evaluación:</p> <p>De la participación: Intervención semanal en el foro con aportes significativos</p> <p>Este criterio representa 20 puntos del total de 115 puntos de la actividad</p>	<p>Nivel alto: El estudiante interactúa de forma oportuna, adecuada y respetuosa en el foro. Realizando aportes significativos de su trabajo en archivos .ipynb, realiza las correcciones a las que haya lugar, de acuerdo con las indicaciones del tutor. Aporta apoyando al aprendizaje de sus compañeros. Si su trabajo se encuentra en este nivel puede obtener entre 10 puntos y 20 puntos</p> <p>Nivel Medio: Aunque el estudiante interactúa en el foro, no responde las participaciones académicas con aportes significativos y su desarrollo no responde con suficiente argumentación frente al referente consultado o no presenta las correcciones indicadas por el tutor. Si su trabajo se encuentra en este nivel puede obtener entre 5 puntos y 9 puntos</p> <p>Nivel bajo: El estudiante no presenta aportes individuales y no participa activamente en el foro. Si su trabajo se encuentra en este nivel puede obtener entre 0 puntos y 4 puntos</p>
<p>Quinto criterio de evaluación:</p> <p>Del desempeño: Cumplimiento de las</p>	<p>Nivel alto: Entrega el trabajo con al menos el 90% de las condiciones exigidas para entrega del documento. Si su trabajo se encuentra en este nivel puede obtener entre 10 puntos y 20 puntos</p>

condiciones para
realización y entrega
del documento final.

**Este criterio
representa 20
puntos del total
de 115 puntos de
la actividad**

Nivel Medio: Entrega el trabajo con el 70% a 90% de las condiciones exigidas para entrega del documento.

Si su trabajo se encuentra en este nivel puede obtener entre 5 puntos y 9 puntos

Nivel bajo: Entrega el trabajo con menos del 70% de las condiciones exigidas para entrega del documento.

Si su trabajo se encuentra en este nivel puede obtener entre 0 puntos y 4 puntos