

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 2
Desarrollo de código para el tratamiento y manejo de
información

1. Descripción de la actividad

Tipo de actividad: Individual	
Momento de la evaluación: Intermedia en la unidad 2	
Puntaje máximo de la actividad: 115 puntos	
La actividad inicia el: lunes, 25 de septiembre de 2023	La actividad finaliza el: domingo, 22 de octubre de 2023
Con esta actividad se espera conseguir el siguiente resultado de aprendizaje: Resultado de aprendizaje 3: Estructurar diferentes técnicas para la manipulación de datos en un lenguaje de programación.	
La actividad consiste en: Realizar los siguientes tres (3) ejercicios en el entorno dinámico de programación Jupyter Notebook: <u>Ejercicio 1: Datos en línea y tipos de datos.</u> Para el desarrollo de este ejercicio es necesario que revise en el Entorno de Aprendizaje (Unidad 2 - Contenidos y referentes bibliográficos), las siguientes referencias: <ul style="list-style-type: none"> • Idris, I. (2014). <i>Python Data Analysis</i>. Packt Publishing. (pp. 131-134). • Samir Madhavan. (2015). <i>Mastering Python for Data Science: Explore the World of Data Science Through Python and Learn How to Make Sense of Data</i>. Packt Publishing. (pp. 11-12). 	
<u>Descripción del ejercicio:</u> Desarrolle códigos y celdas markdown para desarrollar las siguientes tareas:	

- Responda las siguientes preguntas en la introducción:
 - ¿Qué es *REST web service* y cómo se utilizan?
 - ¿Qué es una RESTful API?
 - ¿Qué es JSON?
- Desarrolle un código de python que le permita obtener los datos de las naves de SpaceX, desde la API spacexdata v4: <https://api.spacexdata.com/v4/ships>, tenga en cuenta que debe usar el paquete *requests*, funciones:
 - *request.get*
 - *Response.json*
 - De ser necesario, puede revisar su documentación en: <https://requests.readthedocs.io/en/latest/api/?requests.Request>
- Cargue el json de la respuesta del request anterior como un DataFrame de pandas (ayuda: función `pd.DataFrame.from_dict()`).
- Determine cuantas columnas tiene el DataFrame anterior, los tipos de datos (Dtype) que tiene y los datos ausentes (ayuda: imprima el `info()` del DataFrame de pandas).
- Con la información anterior, en una celda markdown concluya cuales de las variables podrían ser utilizadas para un análisis y clasifíquelas entre variables cualitativas (categóricas) y cuantitativas (numéricas).
- Determine el total de naves y cuántas están activas.
- Visualice el histograma de los pesos de las naves en kg.

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 Datos en línea y tipos de datos en celdas markdown respondiendo las preguntas realizadas. Se debe referenciar con normas APA las fuentes de consulta.
- El desarrollo de los códigos que implementen las tareas solicitadas.
- 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: Limpieza y procesamiento de datos

Para el desarrollo de este ejercicio, es necesario que revise en el Entorno de Aprendizaje (Unidad 2 - Contenidos y referentes bibliográficos), las siguientes referencias:

- 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. 11-18).
- Thakur, A. (2016). *Python: Real-World Data Science*. Packt Publishing. (pp. 511 – 538).

Descripción del ejercicio:

Para este ejercicio, cargue de nuevo los datos del Ejercicio 1, pero genere un DataFrame llamado *df*, solamente con las columnas:

- *type*
- *mass_kg*
- *year_built*
- *home_port*
- *name*

Siguiendo las funciones mostradas en las referencias bibliográficas, desarrolle las siguientes tareas en el DataFrame generado anteriormente:

Revisando datos faltantes:

- Determine cuantos datos faltantes hay en la variable *mass_kg* del mediante *value_counts()*

- Crea un DataFrame df1 removiendo las filas que tienen **algunos** datos faltantes. Por otro lado crea un DataFrame df2 removiendo las filas que tienen **todos** los datos faltantes. Compáralos y comenta los resultados.
- En el DataFrame df completa los datos faltantes por los siguientes tres métodos (muestra y comenta los resultados):
 - Completar datos faltantes con el número 0.
 - Completar por método de propagación hacia adelante.
 - Completar los valores faltantes con el promedio de la columna.

Operaciones de strings:

Realice las siguientes operaciones en el DataFrame df:

- De la columna *home_port* extraiga la primera palabra de cada dato y muestre los resultados.
- Determine cuántos datos comienzan con la palabra Port y cuántos datos comienzan con una palabra diferente, ¿Cuáles son las palabras diferentes?
- Filtre el DataFrame con los datos que en la columna *name* contengan en el string la palabra "GO" y muestre los resultados.
- Convierta todos los strings de la columna *name* en palabras con todas sus letras mayúsculas y muestre los resultados.
- Convierta todos los strings de la columna *type* en palabras con todas sus letras minúsculas y muestre los resultados.
- Divida todos los strings de la columna *home_port* por el espacio en blanco y muestre los resultados.
- Reemplace todas las palabras 'Port' que se encuentran en la columna *home_port* por la abreviación 'P.' y muestre los resultados.
- Comente todos los procesos realizados y concluya con las posibilidades de uso de estas herramientas en su campo de acción.

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 Limpieza y procesamiento de datos en celdas markdown. Se debe referenciar con normas APA las fuentes de consulta.
- El desarrollo de los códigos con los comentarios solicitados.
- 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: Agregación, unión y concatenación de DataFrames

Para el desarrollo de este ejercicio, es necesario que revise en el Entorno de Aprendizaje (Unidad 2 - Contenidos y referentes bibliográficos), las siguientes referencias:

- 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. 19-25).

Descripción del ejercicio.

Para este ejercicio, cargue en la variable df el conjunto de datos que se presentan en el archivo CSV, Anexo 1 - Student Weight Status.csv, que se encuentra en la carpeta de la Unidad 2 - Tarea 2 - Programación para la Manipulación de Datos. Tenga en cuenta el tipo de separador y los encabezados de las columnas.

Realice paso a paso las siguientes tareas, siguiendo lo estudiando en las referencias:

Fusionando datos:

- Cree un DataFrame p1 con los datos del 0 al 1 con las columnas 'area_name' y 'county' de df.
- Creen un DataFrame p2 con los datos del 2 al 4 con las columnas 'area_name' y 'county' de df.
- Concatene p1 y p2 y muestre los resultados.
- Concatene p1 y p2 identificando las asignaciones por las llaves 'p1' y 'p2' y muestre los resultados y asígnelos a la variable *concatened*. Comente las diferencias con el DataFrame anterior y mencione desde su punto de vista las posibles utilidades de esta herramienta.
- Desde el DataFrame *concatened* rescate los datos de p1.
- Comente los resultados y realice las conclusiones del ejercicio.

Operaciones de agregación:

- Encuentre el número promedio de estudiantes que son obesos en escuelas de primaria (tipo 'ELEMENTARY').
- Encuentre el número total de estudiantes de primaria que son obesos.
- Encuentre el número máximo de estudiantes obesos que tiene una escuela.
- Encuentre el número mínimo de estudiantes obesos que tiene una escuela.
- Obtenga la desviación estándar del número de estudiantes obesos en las escuelas.
- Obtenga la cantidad de escuelas de primarias en el condado (county) de DELAWARE.
- Comente los resultados del ejercicio y concluya con las posibilidades de uso de estas herramientas en su campo de acción.

Uniones (Joins):

- Defina un DataFrame de búsqueda llamado *grade_lookup*, que tenga las columnas 'grade_level' y 'level' que asigne niveles a cada uno de los grados como sigue: 1 a 'ELEMENTARY', 2 a 'MIDDLE/HIGH', 3 a 'MISC'.
- Por otro lado, tome las primeras cinco filas de la columna de datos en la columna GRADE e imprímalas (Téngalas en cuenta pues las utilizaremos para estudiar los tipos de uniones).

- Describa los *inner joins* y realice un ejemplo con las primeras 5 filas de la columna `grade_level` y `grade_lookup`, muestre sus resultados y explíquelos.
- Describa los *left outer joins* y realice un ejemplo con las primeras 5 filas de la columna `grade_level` y `grade_lookup`, muestre sus resultados y explíquelos.
- Describa los *full outer joins* y realice un ejemplo con las primeras 5 filas de la columna `grade_level` y `grade_lookup`, muestre sus resultados y explíquelos.

Agrupaciones:

- Encontrar la suma del número de estudiantes obesos en cada uno de los grados mediante el uso de la función `groupby` de `pandas`. Muestre sus resultados y explíquelo.
- Muestre diferentes tipos de agregación por suma, media y desviación estándar de los estudiantes obesos en cada uno de los grados.
- Comente los resultados y concluya con las posibilidades de uso de estas herramientas en su campo de acción.

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 Combinación, unión y concatenación de DataFrames 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.
- 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_Tarea2.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 2 y participar semanalmente con aportes significativos en el foro de la Unidad 2 - Tarea 2 – Programación para la Manipulación de Datos.

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_Tarea2.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 2 - Desarrollo de código para el tratamiento y manejo de información, 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_Tarea2.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: Intermedio

La máxima puntuación posible es de 115 puntos	
<p>Primer criterio de evaluación:</p> <p>Del procedimiento: Desarrollo del Ejercicio 1 de Datos en línea y tipos de datos.</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 1 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>Segundo criterio de evaluación:</p> <p>Del procedimiento: Desarrollo del Ejercicio 2 de Limpieza y procesamiento de datos.</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 2 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>Tercer criterio de evaluación:</p> <p>Del procedimiento: Desarrollo del Ejercicio 3 de Combinación, unión y concatenación de DataFrames.</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>Este criterio representa 25 puntos del total de 115 puntos de la actividad</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 condiciones para realización y entrega del documento final.</p> <p>Este criterio representa 20 puntos del total</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> <p>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</p> <p>Nivel bajo: Entrega el trabajo con menos del 70% de las condiciones exigidas para entrega del documento.</p>

**de 115 puntos de
la actividad**

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