

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN CCPG1043 / CCPG1801 - FUNDAMENTOS DE PROGRAMACIÓN TERCERA EVALUACIÓN

I TÉRMINO 2025-2026/ Septiembre 12, 2025

Compromiso de Honor

Yo,, matrícula del paralelo de Fundamentos de Programación, declaro que he sido informado y conozco las normas disciplinarias que rigen a la ESPOL, en particular el Código de Ética y el Reglamento de Disciplina. Al aceptar este compromiso de honor, asumo la responsabilidad de realizar este examen de manera honesta, sin recurrir a prácticas de plagio, fraude o deshonestidad académica por medios físicos o electrónicos. Me comprometo a responder las preguntas con mis propias habilidades y conocimientos, sin recibir ayuda no autorizada. Además, garantizo que respetaré los derechos de propiedad intelectual y no divulgaré ni copiaré el contenido del examen.

Acepto el presente compromiso, como constancia de haber leído y aceptado la declaración anterior y me comprometo a seguir fielmente las directrices que se indican para la realización de la presente evaluación. Estoy consciente que el incumplimiento del presente compromiso anulará automáticamente mi evaluación y podría ser objeto del inicio de un proceso disciplinario.

Firma:

Tema 1	Tema 2	Tema 3	Tema 4	Total	
					Firma Revisé mi calificación

Funciones y propiedades de referencia en Python.

para listas :	para cadenas :	random as rd :
lista.append(...) lista.extend(...) lista.count(...) lista.index(...) lista.pop() lista.insert(pos, elem) elemento in lista sum(lista) len(lista)	cadena.islower() cadena.isdigit() cadena.isalpha() cadena.lower() cadena.upper() cadena.split(x) cadena.capitalize() cadena.count(x) cadena.replace(a,b)	rd.randint(inicio,fin) rd.randrange(inicio,fin, salto) rd.choice(lista) rd.choices(lista,k=cant) rd.sample(lista,cant) rd.shuffle(lista)
para diccionarios :	pandas as pd :	
clave,valor in dic.items() clave in dic dic.values() dic[clave] dic.get(clave, valor_defecto) dic.setdefault(clave, valor_defecto)	df = pd.DataFrame(...) df.head() df.tail() df.info() df.describe() df["col"] df[lista] df.loc[fila] df.iloc[idx_fila] df["col"].sum()	df["col"].mean() df["col"].unique() df["col"].idxmax() df["col"].reset_index() df["col"].max() df["col"].min() df["col"].count() df["col"].value_counts() df.sort_values("col") df.groupby("col1")["col2"].sum() df["col"].mask(condicion, valor) pd.read_csv("file.csv")

TEMA 1

[25 pts] Dada una cadena de texto como la que se encuentra en la variable `mensaje`.

```
mensaje = "Las redes sociales son el epicentro de la vida digital para los jóvenes,
conectándolos con amigos, tendencias y causas globales. Plataformas como TikTok e
Instagram no solo ofrecen entretenimiento, sino que también moldean identidades y
opiniones, desde la moda hasta el activismo."
```

A partir de esta cadena realice lo siguiente:

1. Elimine todos los signos de puntuación (. , ;)
2. Convierta la cadena a minúsculas.
3. Seleccione 20 palabras distintas al azar. **Con este grupo de palabras:**
 - Ordénelas alfabéticamente.
 - Asegúrese de que la primera palabra tenga al menos 8 caracteres. Si no alcanza esa longitud, agregue el carácter '-' al final de la palabra hasta completar.
 - Convierta a mayúsculas las palabras en los índices impares (1,3,5,...).
 - Forme una nueva cadena uniendo todas estas palabras. (Ver ejemplo)
 - Muestre la nueva cadena resultante.

Ejemplo de salida:

```
causas-- COMO conectándolos EL epicentro INSTAGRAM la LAS los MOLDEAN no OFRECEN
para QUE redes TAMBIÉN tiktok VIDA y Y
```

TEMA 2

[20 pts] Implemente la función `productos_disponibles(d_prod)` que recibe un diccionario de productos. El diccionario tiene como claves los códigos de producto y como valores una lista con tres elementos:

1. Nombre del producto (cadena).
2. Precio por unidad (número).
3. Disponibilidad en stock (booleano).

La función debe devolver un nuevo diccionario donde las claves sean los **nombres** de los productos que tienen **disponibilidad True** y cuyo valor es el **precio** correspondiente.

Ejemplo del diccionario:	Salida esperada:
<pre>{ "P001": ["Manzanas", 2.5, True], "P002": ["Peras", 3.0, False], "P003": ["Bananas", 1.8, True] }</pre>	<pre>{ "Manzanas": 2.5, "Bananas": 1.8 }</pre>

TEMA 3

3.1 [20 pts] Implemente la función `es_perfecto` que reciba un número `n` entero mayor a 1. La función debe devolver `True` si `n` es un número perfecto, o `False` en caso contrario.

Un número es perfecto cuando la suma de sus divisores propios (todos los divisores positivos distintos del propio número) es igual a él mismo.

Ejemplos:

Número	Divisores propios	Suma	Perfecto
2	1	1	No
3	1	1	No
4	1, 2	3	No
5	1	1	No
6	1, 2, 3	6	Si

3.2 [10 pts] Escriba un programa que:

- Solicite el ingreso de un número por teclado.
- Valide hasta que el valor ingresado sea un número entero mayor a 1. Considere que el usuario podría ingresar cualquier tipo de caracter por teclado.
- Muestre **cuántos** números perfectos existen entre 2 y el número ingresado inclusive.

TEMA 4

Se tiene un DataFrame `df` con información de productos vendidos en una tienda en línea. Sus columnas son:

- **fecha (str)**: Fecha de publicación del producto.
- **id_producto (str)**: Código único del producto.
- **marca (str)**: Marca del producto.
- **descripción (str)**: Descripción corta del producto.
- **precio (float)**: Precio en dólares.
- **categoría (str)**: Categoría a la que pertenece el producto.
- **valoración (float)**: Promedio de valoraciones de los clientes entre 1 y 5.

Nota: En la siguiente tabla se presenta información de ejemplo.

fecha	id_producto	marca	descripción	precio	categoría	valoración
01/07/2024	B0946W4VJ9	Ted Baker	Bolsos y accesorios Nikon	109.00	Bolsos de mano	3.8
18/05/2024	B0BNL81YV7	Adidas	Gorra	34.00	Nueva temporada	4.8
23/02/2024	B083LTB6CP	Besttravel	Mochila para portátil	61.99	Equipaje	4.2
15/15/2024	B0C339DBNZ	Skechers	Zapato de punta de acero	208.66	Zapatos	4.6
12/10/2024	B0DK42VCQ4	Casio	Reloj informal F-91WB-7ACF blanco	115.44	Relojes	4.7

Responda los siguientes literales usando pandas:

- 4.1 [6 pts] Del producto más **costoso**, muestre únicamente (`id_producto`, `descripción`, `valoración`).
- 4.2 [6 pts] De los productos marca **Adidas**, muestre el precio promedio y la valoración más baja.
- 4.3 [6 pts] Muestre toda la información de los 5 productos más baratos y ordénelos de mayor a menor precio.
- 4.4 [7 pts] Para cada **categoría**, calcule el número total de productos, la valoración promedio y genere un nuevo DataFrame con esta información.