Instrucciones de Entrega – Examen Final

<https://uao-my.sharepoint.com/:w:/g/personal/njvelandia_uao_edu_co/EX8lqzyvkoxInNxRW1ynN6EBHkAx6yrbvRS8ebQk0YY66A?e=sCUdsy>

• Sube los archivos de código correspondientes al proyecto en el apartado de “Examen Final” en la plataforma.  
• Incluye también el enlace al repositorio de GitHub donde se encuentra el proyecto completo.  
• Agrega como colaborador del repositorio a: natali.velandia@gmail.com

Importante:  
El desarrollo del proyecto debe realizarse únicamente con los conceptos y estructuras vistos en clase o con código proporcionado directamente por la profesora.  
SE PUEDEN INVESTIGAR FUNCIONES ESPECIFICAS PERO NO FRAGMENTOS COMPLETOS DE CÓDIGOS EXTERNOS Y  debes referenciar correctamente la fuente utilizada.  
Se calificará la trazabilidad del trabajo en GitHub, por lo tanto, recuerda hacer push con cada cambio que realices.

Enunciado del Proyecto

Sistema de Gestión de Rendimiento en Pruebas Deportivas

Un centro deportivo está organizando una competencia donde se evalúa el rendimiento físico de sus participantes en tres pruebas diferentes. Cada prueba tiene un nivel de dificultad aleatorio, lo que influye en el cálculo del puntaje final ponderado de cada participante.  
El sistema debe implementarse como una interfaz gráfica utilizando tkinter y debe ofrecer el siguiente menú:

1. Registrar participante  
2. Mostrar reporte general  
3. Mostrar reporte individual por participante  
4. Salir

1. Registrar participante

Los datos pueden almacenarse en listas o en un diccionario.  
  
- El usuario ingresa el nombre del participante.  
- Ingresa los resultados (puntajes entre 0 y 100) en tres pruebas físicas: resistencia, fuerza y velocidad.  
- Para cada prueba, el programa genera automáticamente un nivel de dificultad decimal aleatorio entre 1.0 y 1.3 usando random.  
- Se calcula el puntaje final ponderado con la fórmula:  
  Puntaje Final = (∑ (puntaje × dificultad)) / (∑ dificultades)  
- El puntaje se redondea a un entero usando round() o math.ceil().  
- Si el puntaje es mayor o igual a 70, el participante clasifica. De lo contrario, no clasifica.

2. Mostrar reporte general

Debe incluir:  
- Nombre de cada participante  
- Puntaje final  
- Estado: Clasificó / No clasificó  
  
Además, debe mostrar:  
-estadísticas importantes (describe en pandas)  
- Puntaje promedio del grupo  
- Gráfico de torta con el total de clasificados y no clasificados  
- Matriz de correlación de los puntajes (si se usó pandas)

3. Mostrar reporte individual por participante

- El usuario ingresa el nombre del participante.  
- Se debe mostrar:  
  - Puntaje final  
  - Estado  
  - Resultados por prueba en un histograma con matplotlib  
  - Nivel de dificultad aplicado en cada prueba en un histograma con matplotlib  
  - Puntaje ponderado por prueba en un histograma con matplotlib