



AGENDA DE ENTRENAMIENTOS PERSONALIZADOS

Dentro de las tendencias actuales relacionadas al cuidado de la salud, se volvió muy frecuente el entrenamiento con sobrecarga progresiva, el cual no es suficiente si no viene acompañado de una alimentación acorde al plan de ejercicios físicos. Para facilitar esta última tarea de los entrenadores presento una agenda de entrenamientos personalizados, la cual está pensada para almacenar alumnos con sus respectivos seguimientos y evoluciones dentro de cada plan individual.

Esta aplicación está desarrollada en Python y, presentada en una interfaz gráfica de usuario, se podrán cargar los datos de cada alumno y consultar los mismos a través de una base de datos para su edición.

- Se podrá acceder a una hoja de evolución personal de cada alumno para su correcto seguimiento.
- Se podrá realizar un cálculo de metabolismo basal mediante la acción de un botón en pantalla, cálculo para cada alumno de acuerdo a los datos ingresados y objetivos de su plan.

RESUMEN

Se utilizó tecnología Python 3.10.0, y Tkinter como módulo GUI.

Se incorporó lo aprendido en el módulo **Python Nivel Intermedio**: **Versión 1.0**

- ✓ Se trabajó con módulos y paquetes.
- ✓ Se aplicaron los conceptos de P.O.O, utilizando métodos constructores y aplicación de herencia entre otros.
- ✓ Se utilizaron variables y métodos de instancia. Se trabajó con GUI (Tkinter) diseñando con este ultimo los requisitos específicos de la App (utilización de botones, marcos).
- ✓ Se aplicaron patrones de diseño (MVC) solo dos módulos, la vista y el controlador, dado la simpleza de la App y la falta de enrutamientos múltiples por parte de un modulo controlador
- ✓ Como fuente de documentación se utilizaron comentarios en el código.
- ✓ En esta etapa inicial 1.0 de desarrollo de la App se cumplieron los requisitos de un C.R.U.D consultando una base de datos Sqlite3. A futuro se piensa actualizar con las respectivas características de esta App incorporando una hoja de evolución para cada ID (alumno) mediante el funcionamiento de un botón. Y la implementación de otro widget para el funcionamiento del cálculo de metabolismo basal.

Versión 1.1

- ✓ Se corrige sintaxis siguiendo la guía de estilo PEP 8.
- ✓ Agrega una validación de campos de entrada (Permitiendo solo caracteres alfabéticos).
- ✓ Agrega un control de excepciones.

Se incorporó lo aprendido en el módulo **Python Nivel Avanzado**: **Versión 1.2**

- ✓ Como diseño de arquitectura ya tenía aplicado el patrón MVC, se le agrega el módulo controlador, simulando distintos enrutamientos (ya que solo posee uno) y afirmando la existencia del patrón.
- ✓ En reemplazo de Sqlite3 como base de datos, se utilizó el ORM Peewee.
- ✓ Se agregó un decorador, su función principal es la de llevar un registro de las distintas operaciones del C.R.U.D en un archivo de texto en la misma ruta de la App.
- ✓ Se agregó un patrón de diseño observador a fines de aplicar lo aprendido en la Unidad 6. Su función es la de registrar en un archivo de texto (en la misma locación) la fecha, nombre y apellido de los alumnos agendados para el entrenador/usuario de la App.

Se incorporó en su última actualización, diseño a ***Pedido del Cliente:*** Versión 2.0

- ✓ El funcionamiento completo del botón evolución, con los botones correspondientes:
 - Botón guardar, para guardar en un archivo serializado (binario) la evolución particular de cada alumno.
 - Botón mostrar, para acceder a la última evolución existente del alumno.
 - Tasa de metabolismo basal (diferentes cálculos de acuerdo al género del alumno y valores ingresados) arroja un resultado en pantalla en ventana evolución.
 - Rutina de ejercicios: el acceso a la búsqueda de una rutina de ejercicios, una plantilla pre armada para mejorar la dinámica del seguimiento de alumnos personalizados en formato .docx/.pdf.