

#DESAFIO GRUPAL



CALENTANDO MOTORES

¡Bienvenido al Desafío Grupal! En un mundo en constante cambio, el aprendizaje es una de las habilidades más importantes que debemos cultivar.

Dentro de esta actividad, te vas a encontrar con un solo ejercicio que tienen como objetivo brindarte la oportunidad de trabajar en equipo, y nivelarte junto con quienes comparten tu misma guía de Java POO, además de que puedas ponerte a prueba. Para este desafío, vas a tener que investigar e introducir nuevos conceptos en los cuales profundizaremos más adelante, con las guías de Colecciones.

Te recomendamos leer atentamente el enunciado, plantear una posible solución junto con tu equipo, y comenzar a programar. ¡Muchos Éxitos!

#DESAFIO GRUPAL



ACLARACIÓN IMPORTANTE

Como te mencionamos anteriormente, para realizar este desafío van a tener que investigar acerca de los siguientes temas, a manera de introducción.

1. ¿Que es CRUD?
2. Declaración y creación de una ArrayList:
 - ¿Cómo declarar un ArrayList?
3. Operaciones CRUD:
 - Agregar elementos a una ArrayList.
 - Leer elementos de una ArrayList.
 - Actualizar elementos de una ArrayList.
 - Eliminar elementos de una ArrayList.

#DESAFIO GRUPAL

- EJERCICIO 1 (ACTIVIDAD PARA EL DÍA N° 18)

Realizar una aplicación de Gimnasio, que permita tener agregar clientes, y rutinas. Para ello, vamos a utilizar diferentes clases y metodos;

CLASES:

- Crea una clase de entidad llamada "**Cliente**" con los siguientes atributos: id, nombre, edad, altura, peso, objetivo (cadena de caracteres)
- Crea una clase de entidad llamada "**Rutina**" con los siguientes atributos: id, nombre, duracion, nivelDificultad, descripcion
- Crea una clase **ServicioCliente** y una clase **ServicioRutina**.

#DESAFIO GRUPAL

La clase "ServicioCliente" debe tener los siguientes métodos de CRUD:

- **registrarCliente:** lo registra en el sistema.
- **obtenerClientes():** devuelve una lista con todos los clientes registrados en el sistema.
- **actualizarCliente(int id, String nombre, int edad, double altura, double peso, String objetivo):** recibe el identificador de un cliente existente y los nuevos datos del cliente, y actualiza la información correspondiente en el sistema.
- **eliminarCliente(int id):** recibe el identificador de un cliente existente y lo elimina del sistema.

La clase "ServicioRutina" debe tener los siguientes métodos de CRUD:

- **crearRutina(Rutina rutina):** recibe un objeto de tipo Rutina y lo agrega al sistema.
- **obtenerRutinas():** devuelve una lista con todas las rutinas registradas en el sistema.
- **actualizarRutina(int id, String nombre, int duracion, String nivelDificultad, String descripcion):** recibe el identificador de una rutina existente y los nuevos datos de la rutina, y actualiza la información correspondiente en el sistema.
- **eliminarRutina(int id):** recibe el identificador de una rutina existente y la elimina del sistema.
- En el método principal (main):

#DESAFIO GRUPAL

Una vez terminado de codear, realizar lo siguiente:

- Crea al menos cinco clientes con datos ficticios.
- Crea al menos cinco rutinas con datos ficticios.
- Registra los clientes creados en el sistema utilizando el método correspondiente del servicio de clientes.
- Agrega las rutinas creadas al sistema utilizando el método correspondiente del servicio de rutinas.
- Obtén la lista de todos los clientes registrados y muéstralos por pantalla.
- Obtén la lista de todas las rutinas registradas y muéstralas por pantalla.
- Realiza una actualización en uno de los clientes existentes y muestra la lista de clientes actualizada.
- Realiza una actualización en una de las rutinas existentes y muestra la lista de rutinas actualizada.
- Elimina uno de los clientes existentes y muestra la lista de clientes final.
- Elimina una de las rutinas existentes y muestra la lista de rutinas final.

Nota adicional: Puedes agregar más funcionalidades y métodos a las clases de entidad y servicio según consideres necesario para incrementar aún más la complejidad del ejercicio.

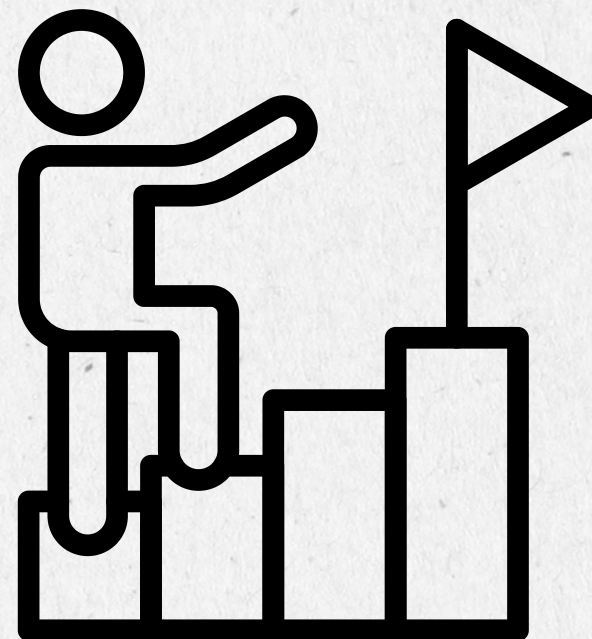
#DESAFIO GRUPAL

- **NOTA:**

Para realizar este desafío, es importante realizar una investigación de los temas mencionados (ArrayList, y CRUD). Es por ello que para este desafío tendrán 2 días de trabajo para poder realizarlo.

Lo importante es no apresurarse en querer completarlo, sino poner en practica y despejar dudas.

No te preocupes si no entiendes el manejo de las Listas aún, eso lo profundizaremos más adelante en la guía de Colecciones.



#DESAFIO GRUPAL

- **DESAFÍO EXTRA**

En el caso de haber completado junto con tu grupo el desafío, te proponemos lo siguiente:

- **REFACTORIZAR** el código de un compañero, intentando reducir un 10% la cantidad de líneas de código existentes. (Si hay **10** líneas, hay que buscar la forma de "optimizar" el código al menos 1 línea.) Aquí es importante **NO** buscar arreglar o modificar la lógica de un compañero, lo interesante de esta actividad es obtener resultados comprendiendo el proceso lógico de otra persona.

