

Análisis y Diseño Ejercicio3 (Arreglos Dinámicos)

Marco Soloj 25701

1. Requisitos funcionales del sistema

- El sistema debe permitir el ingreso de nuevos miembros del gimnasio, ingreso de los entrenadores del gimnasio y el ingreso de las rutinas sea para pecho, abdomen, espalda, bíceps, push, etc....
- El sistema debe dejar asignar un entrenador a un miembro y también debe de poder asignar una rutina a el miembro. Así mismo debe poder identificar cuando no hay entrenadores, miembros o rutinas.
- Debe poder mostrar la rutina con más miembros si es que hay rutinas y con cuantos miembros.
- Debe mostrar el número de rutinas que están activas.
- Debe mostrar el entrenador con la mayor cantidad de miembros asignados.
- Debe dejar que el usuario seleccione la opción que quiera del menú para poder manejarlo y debe permitirle salir del programa en cualquier momento.

2. Clases necesarias y su propósito

Clase	Propósito
Ejer3	Es la clase inicial que inicializa el sistema y se encarga de lanzar la parte de la VistaEjer3
ControladorEjer3	Añade personas en ArrayList, entrenadores y rutinas todos en ArrayLists, así mismo se encarga de retornar o de hacer la lógica de lo que sería las funciones de retornar la rutina con mayor cantidad de miembros, las rutinas activas y los entrenadores con la mayor cantidad de personas asignadas.
VistaEjer3	Se encarga de mostrar el menú al usuario así mismo de solicitar los datos necesarios para realizar las operaciones del Controlador y muestra los resultados de las operaciones realizadas.
PersonaGym	Se encarga de capturar y colocar lo que es el nombre, membresia, rutina y entrenador a un miembro del gimnasio, representa a las personas.

EntrenadorGym	Se encarga de capturar y colocar el nombre del entrenador, captura en una lista la cantidad de personas que se van asignando con el entrenador y asigna personas o las añade a dicha lista.
Rutina	Se encarga de capturar y colocar datos como lo son el nombre de la rutina, la duración, el estado, los miembros asignados por rutina y ver si incrementan los miembros.

3. Atributos de cada clase

Clase: PersonasGym

Atributo	Tipo de dato	Visibilidad	Propósito
Nombre	String	Private	Nombre del miembro
tipoMembresia	String	Private	Tipo de Membresia (Básica o Premium)
rutinaAsignada	Rutina	Private	Rutina que realiza la persona
entrenadorAsignado	EntrenadorGym	Private	Entrenador asignado a la persona

Clase: Rutina

Atributo	Tipo de dato	Visibilidad	Propósito
nombreRutina	String	Private	Nombre de la rutina
duración	Int	Private	Duración en minutos
RutinaActiva	Boolean	Private	Indica si está activa o no
miembrosAsignados	int	Private	Número de miembros que realizan la rutina

Clase: EntrenadorGym

Atributo	Tipo de dato	Visibilidad	Propósito
nombre	String	private	Nombre del entrenador
personasAsignadas	List<PersonaGym>	private	Lista de personas que entrena

Clase: ControladorEjer3

Atributo	Tipo de dato	Visibilidad	Propósito
personas	List<PersonaGym>	private	Lista de miembros registrados
entrenadores	List<EntrenadorGym>	private	Lista de entrenadores registrados
rutinas	List<Rutina>	private	Lista de rutinas disponibles

4. Métodos de cada clase

Clase: PersonasGym

Método	Parámetros	Retorno	Visibilidad	Propósito
getNombre	-	String	public	Obtener el nombre
setNombre	String	void	public	Modificar el nombre
getTipoMembresia	-	String	public	Obtener la membresía
setTipoMembresia	String	void	public	Modificar la membresía
getRutinaAsignada	-	Rutina	public	Consultar rutina
setRutinaAsignada	Rutina	void	public	Asignar rutina
getEntrenadorAsignado	-	EntrenadorGym	public	Consultar entrenador
setEntrenadorAsignado	EntrenadorGym	void	public	Asignar entrenador
toString	-	String	public	Representación textual

Clase: Rutina

Método	Parámetros	Retorno	Visibilidad	Propósito
getNombreRutina	-	String	public	Obtener nombre
setNombreRutina	String	void	public	Modificar nombre
getDuracion	-	int	public	Obtener duración
setDuracion	int	void	public	Modificar duración
isRutinaActiva	-	boolean	public	Verificar si está activa
setRutinaActiva	boolean	void	public	Cambiar estado
getMiembrosAsignados	-	int	public	Consultar miembros
incrementoMiembros	-	void	public	Aumentar contador
toString	-	String	public	Representación textual

Clase: EntrenadorGym

Método	Parámetros	Retorno	Visibilidad	Propósito
getNombre	-	String	public	Obtener el nombre
setNombre	String	void	public	Modificar el nombre
getPersonasAsignadas	-	List<PersonaGym>	public	Consultar lista de personas
asignarPersona	PersonaGym	void	public	Agregar persona a la lista
toString	-	String	public	Representación textual

Clase: ControladorEjer3

Método	Parámetros	Retorno	Visibilidad	Propósito
añadirPersona	PersonaGym	void	public	Agregar persona
añadirEntrenador	EntrenadorGym	void	public	Agregar entrenador
añadirRutina	Rutina	void	public	Agregar rutina
getPersonas	-	List<PersonaGym>	public	Consultar lista de personas
getEntrenadores	-	List<EntrenadorGym>	public	Consultar lista de entrenadores
getRutinas	-	List<Rutina>	public	Consultar lista de rutinas
getRutinaConMasMiembros	-	Rutina	public	Obtener rutina con más miembros
getNumeroRutinasActivas	-	int	public	Contar rutinas activas
getEntrenadorConMasPersonas	-	EntrenadorGym	public	Entrenador más ocupado

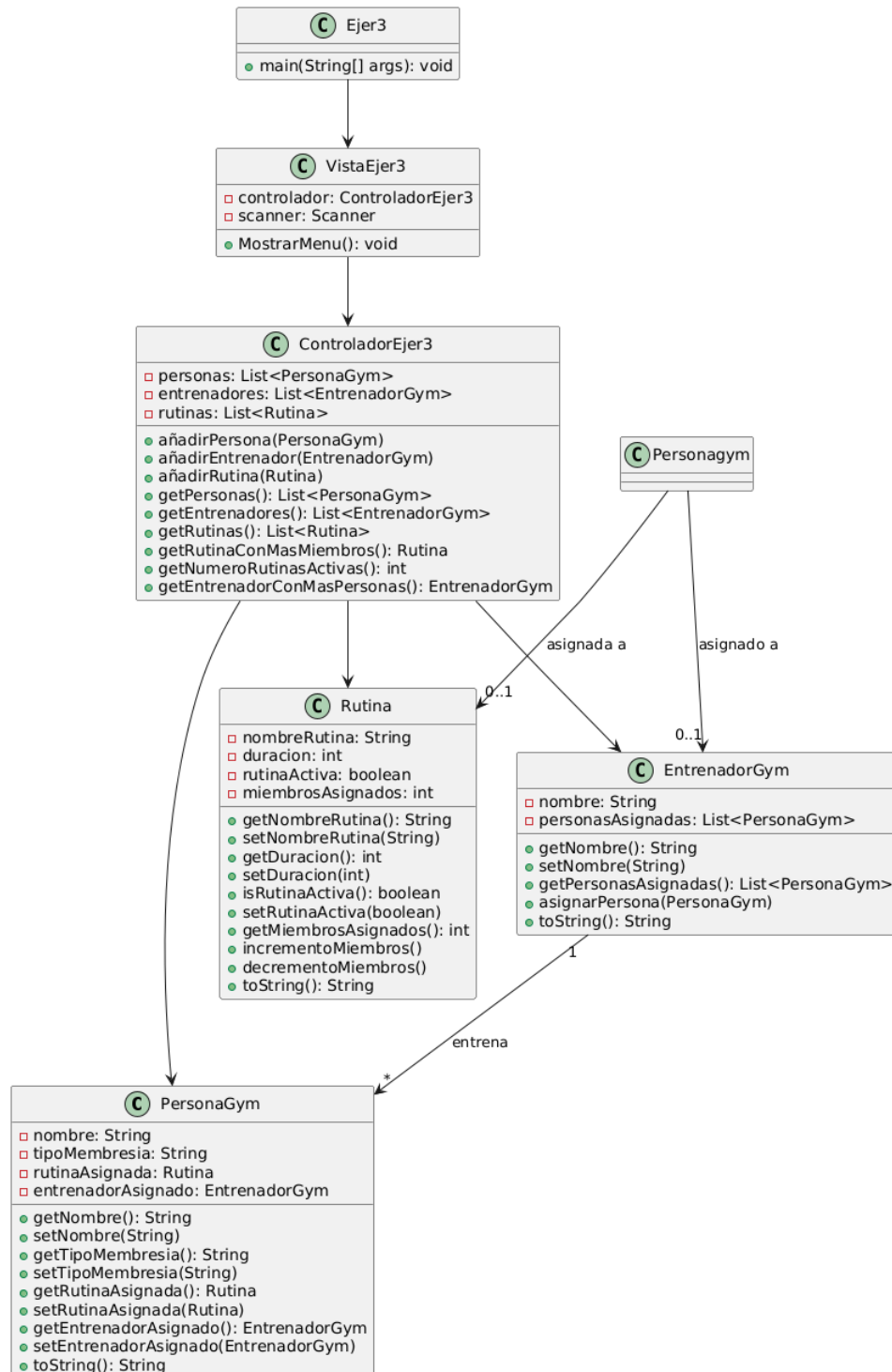
Clase: VistaEjer3

Método	Parámetros	Retorno	Visibilidad	Propósito
MostrarMenu	-	void	public	Mostrar menú y gestionar interacciones

Clase: Ejer3

Método	Parámetros	Retorno	Visibilidad	Propósito
main	String[]	void	public	Ejecutar el programa

5. Diseño de Clases



6. Programa

Enlace a GitHub: <https://github.com/Maquito365/Ejercicio3>

<https://github.com/Maquito365/Ejercicio3>