EAN UNIVERSIDAD  
ESCUELA DE ADMINISTRACIÓN DE NEGOCIOS

SISTEMA DE GESTIÓN PARA UNA UNIVERSIDAD

Profesora: Dilsa Enith Triana Martínez  
  
Mayckoll Andrés Torres Diaz

Bogotá DC  
2025

**INTRODUCCIÓN**  
El sistema permite registrar y gestionar diferentes entidades dentro de la universidad, tales como facultades, profesores, carreras y estudiantes. Está diseñado para facilitar la administración interna de la institución y proporcionar una plataforma eficiente para registrar y consultar información relacionada con sus componentes principales.

Este sistema, implementado en Java, utiliza estructuras de datos como listas y arreglos para almacenar y manipular la información. Además, incorpora un menú interactivo que permite a los usuarios registrar nuevos datos, consultar la información registrada y realizar diversas operaciones de manera sencilla. El objetivo es mejorar la organización administrativa dentro de la universidad, permitiendo una gestión más ágil y centralizada de los distintos recursos educativos.

**OBJETIVOS**

**Objetivo General:**

Desarrollar un sistema de gestión para la UniversidadEan que permita registrar, organizar y consultar información de facultades, profesores, carreras y estudiantes, facilitando la administración y el acceso a los datos relevantes para la institución

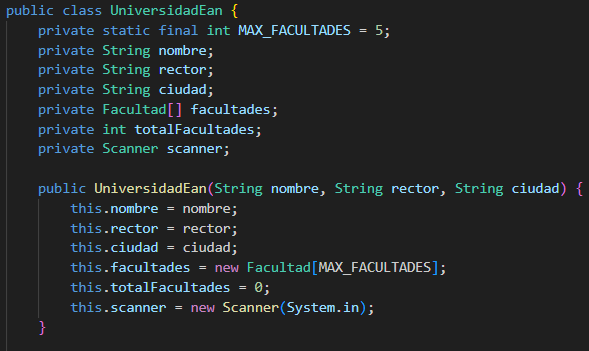
**Objetivos Específicos:**

* Implementar una interfaz interactiva que permita a los usuarios registrar y consultar datos sobre facultades, profesores, carreras y estudiantes de manera sencilla y eficiente.
* Diseñar una estructura de datos adecuada utilizando clases y colecciones que permitan almacenar, modificar y recuperar la información de las entidades de manera organizada y optimizada.
* Garantizar la validación y control de los datos ingresados para asegurar que solo se registren datos válidos, como códigos de facultades existentes y niveles de carrera correctamente especificados.

**RESOLUCIÓN DE LA PROBLEMÁTICA**El sistema fue implementado utilizando el lenguaje de programación Java, y sigue una arquitectura orientada a objetos (OOP).   
  
La herencia también fue utilizada para definir clases específicas para cada tipo de vehículo, mientras que la clase base Vehículo me proporciono los atributos comunes de cada vehículo como la placa cilindraje etc.  
  
 **Tabla de Clases Gestión Universidad**  
  
En esta tabla se distinguen todos aquellos métodos y atributos que se deben crear para cada una de las clases, ya que es necesario para que el sistema funcione adecuadamente.

|  |  |  |
| --- | --- | --- |
| Clase | Atributos | Métodos |
| UniversidadEan | nombre (String), rector (String), ciudad (String), facultades (Facultad[]), totalFacultades (int), scanner (Scanner) | registrarFacultad(), registrarProfesor(), registrarCarrera(), registrarEstudiante(), listarFacultades(), listarProfesores(), menu(), main() |
| Facultad | nombre (String), codigo (int), carreras (List<Carrera>), profesores (List<Profesor>) | agregarCarrera(Carrera), agregarProfesor(Profesor), getCodigo(), mostrarInfo(), listarProfesores() |
| |  | | --- | | **Carrera** |  |  | | --- | |  | | nombre (String), creditos (int), semestres (int), nivel (String), facultad (Facultad) | Constructor (Carrera()), getNombre(), getCreditos(), getSemestres(), getNivel(), getFacultad() |
| Profesor | cedula (String), nombre (String), fechaNacimiento (String), lugarNacimiento (String), profesion (String), nacionalidad (String), sueldo (double), facultad (Facultad) | getNombre(), getProfesion() |
| Estudiante | cedula (String), nombre (String), fechaNacimiento (String), lugarNacimiento (String), colegio (String), fechaIngreso (String | Constructor (Estudiante()), getCedula(), getNombre(), getFechaNacimiento(), getLugarNacimiento(), getColegio(), getFechaIngreso() |

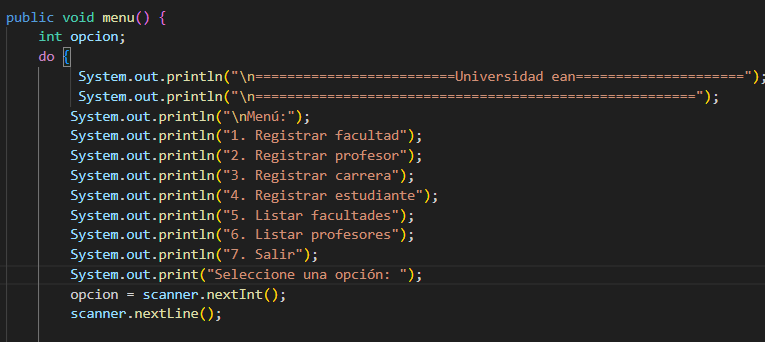
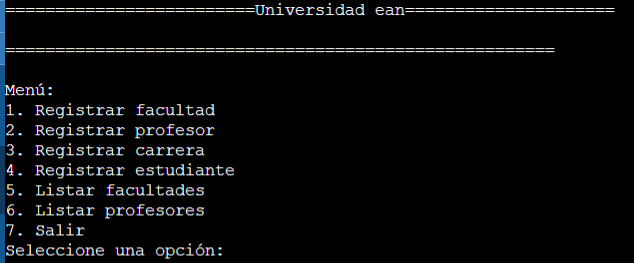
**Clase UniversidadEan**

Encargada de obtener cada uno de los atributos como lo es nombre, rector,ciudad,facultades como array, total de facultades teniendo en cuenta que el máximo actual es 5  


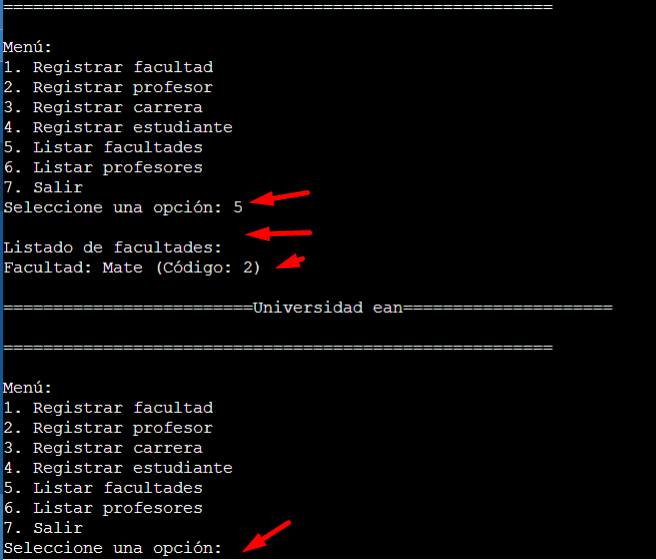
**Metodo RegistrarFacultad**

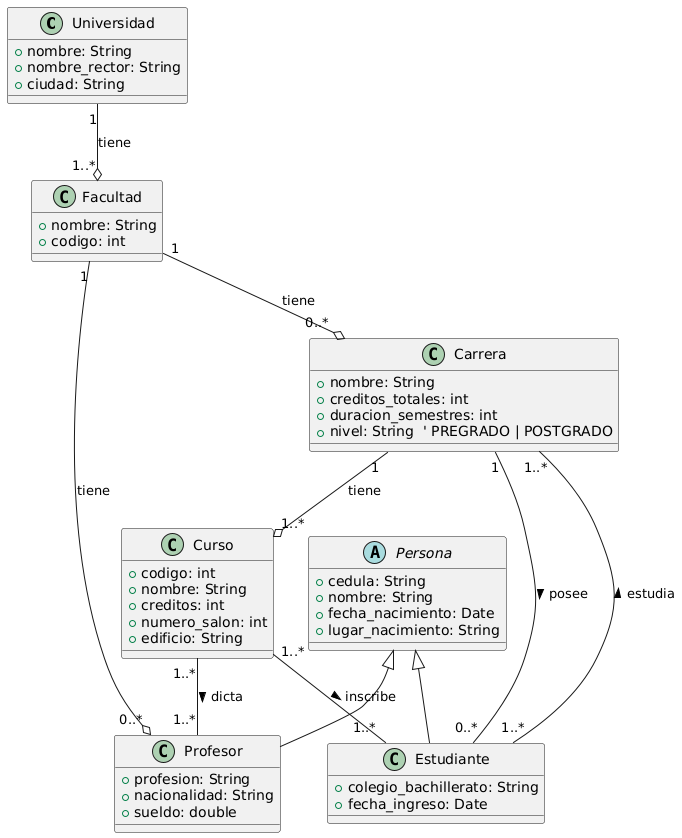
tiene la responsabilidad de registrar una nueva facultad en el sistema, Verifica si ya se ha alcanzado el número máximo de facultades, si el límite se ha alcanzado, imprime un mensaje y termina la ejecución del método mediante return y también se encuentra el método registrar profesor con sus datos y el sueldo referenciado en el ejercicio propuesto en clase.



**SALIDA ESTANDAR DE SCRIPT POR CONSOLA**  
Debido a que tenemos varias opciones se procede a realizar el levantamiento de front-end por lado del cliente en este momento seria vía input cmd.  
  
  
  
Salida por Consola  


Ejecución según orden, cada orden de las 7 opciones esta estandarizada para realizar el envio de datos a las clases y alamacenarlo en un localstorage, como podemos ver se observa que el sistema vuelve a su bucle original para pedir mas data.



**DIAGRAMA DE CLASES UML**

**CONCLUSIONES**

El sistema de gestión de la UniversidadEan está diseñado de manera modular, utilizando clases para representar las entidades fundamentales de la universidad, como facultades, profesores, carreras y estudiantes. Este enfoque facilita la organización y la reutilización del código, permitiendo una gestión más eficiente de los datos.

El sistema UniversidadEan es una herramienta útil y robusta para la gestión académica, con un diseño adecuado para facilitar tanto la administración de recursos como la interacción con los usuarios.