**PROGRAMACIÓN ORIENTADA A OBJETOS**

**Guía de Laboratorio N11**

**Objetivo:**

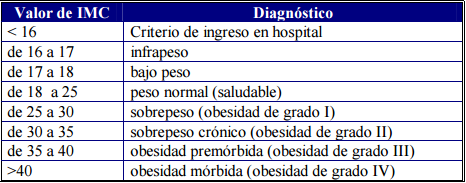
Interfaces graficas

**CASO DE NEGOCIO 01: CALCULADORA**

Crear una interfaz gráfica que simule el funcionamiento de una calculadora de números enteros y muestre el resultado de las principales operaciones: suma, resta, multiplicación y división. Adicione algunas validaciones como el caso de la división entre 0

**CASO DE NEGOCIO 02: CALCULO IMC**

Diseñar una interfaz gráfica que simule el cálculo del IMC en base al peso y altura y muestre un mensaje de recomendación. La fórmula para encontrar el IMC es PESO/ALTURA2



**CASO DE NEGOCIO 03: ADMINISTRACIÓN DE INSTITUCIONES.**

La corporación educativa “COSMOVISION S.R.L” le ha solicitado el desarrollo de un sistema que permita la administración de sus instituciones. Luego de las primeras reuniones se obtuvo la siguiente información:

1. Se tienen 3 categorías: Colegio, Instituto Superior, Universidad con información de código, nombre, ubicación, cantidad de alumnos.
2. La cantidad de alumnos para los colegios se determina en base a la suma de los primeros 2 números primos entre 500 y 800.
3. La cantidad de alumnos de un instituto superior es la suma de los dos primeros números divisibles por 7 entre 500 y 700.
4. La cantidad de alumnos de las Universidades es un aleatorio entre 400 y 700 que termine en 6.

Adicionalmente diseñe la siguiente funcionalidad:

* Método que permita ingresar una institución, como máximo 10 instituciones
* Método que permita eliminar una institución por un código enviado como parámetro, recuerdo desplazar todas las otras instituciones.
* Método que permita crear un arreglo de instituciones con cantidad de alumnos mayores a 300 y otro con menores partiendo del arreglo principal.
* Método que ingresar manualmente una nueva institución en una posición especifica enviada como parámetro, todas las instituciones posteriores deben ser traslados a las siguientes posiciones. Verifique que haya espacio disponible.

Diseñe una clase gráfica con la información solicitada tanto para el ingreso como salida de datos, cada funcionalidad debe estar asociada a un botón.

**CASO DE NEGOCIO 04: PROYECTO VENTA DE PASAJES.**

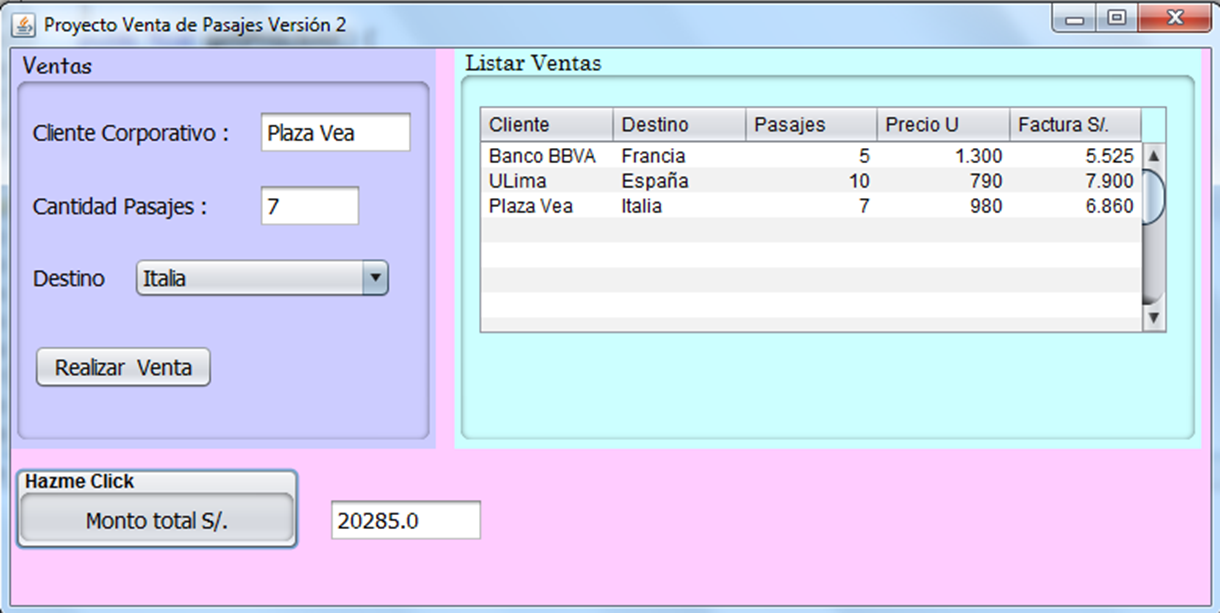
La solución del siguiente caso de negocio deberá estar Orientada a Objetos (abstracción, ocultamiento de la información, encapsulación).

Viajes Paradisiacos Perú SAC ofrece tres destinos a sus clientes corporativos, con la siguiente política de descuentos: ***SUGERENCIA: establecer una estructura selectiva por cada situación***

|  |  |  |
| --- | --- | --- |
| **Destino** | **Cantidad de pasajes** | **Descuento** |
| Francia | <=30 | 15% |
| [30 – 70] | Por cada $1000.00 , $40 de descuento |
| >70 | 7x3 (llevas 7 y sólo pagas 3) |
| Italia | <=50 | Sin descuento |
| >50 | 20% |
| España |  | Si la cantidad de pasajes es mayor a 20, se descuenta $500.00 |

1. Diseñe la **clase Venta** con variables de instancia: nombre del cliente corporativo, destino, cantidad de pasajes, precio unitario, monto facturado, con un constructor con argumentos. También considerar los métodos que permitan hallar el monto facturado, el método concatenarUnaVenta y otros métodos que Ud. crea conveniente.
2. Diseñe la **clase Prueba** y utilizando la clase Venta realice lo siguiente:

* Cree tres objetos VentaPasajes.
* Mostrar todas las ventas realizadas (nombre del cliente corporativo, destino, cantidad de pasajes, precio unitario y monto facturado).
* Mostrar también el número de ventas realizadas (cantidad de objetos VentaPasajes creados). Utilizar una variable de clase.



**CASO DE NEGOCIO 04: PROYECTO GESTION NOTAS.**

Se desea gestionar el promedio de prácticas (en total 6 notas) de un alumno considerando para ello sus 5 mejores notas.

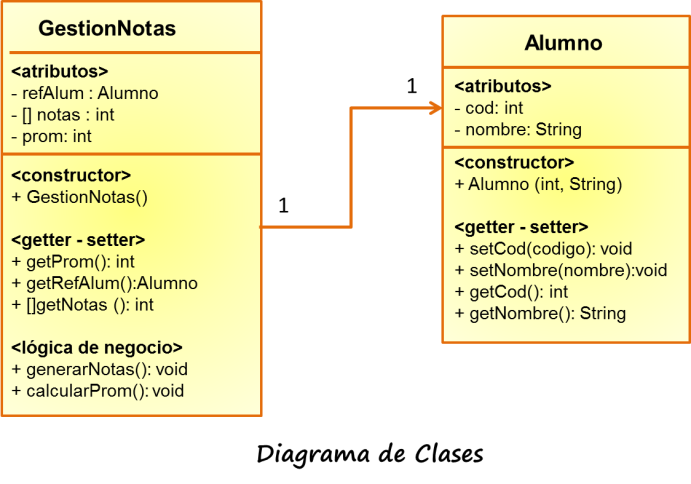
Se pide:

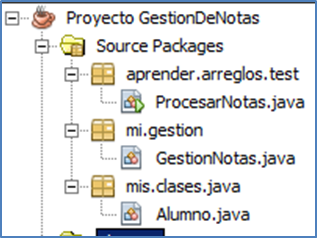
* Diseñar la clase **Alumno**, con variables de instancia: código y nombre.
* Un constructor sin parámetros que inicialice por defecto por atributos de la clase.
* Los métodos getter y setter.
* Diseñar la clase **GestionNotas** que tiene como variables de instancias: un objeto referencia a la clase Alumno, un arreglo para almacenar 6 notas y el promedio de notas.
* Un constructor con un parámetro, el objeto referencia a alumno, y tiene como código la creación del arreglo de las 6 notas y el promedio de notas inicializado en cero.
* Un método que genera aleatoriamente las 6 notas del alumno y los almacene en el arreglo.
* Un método que calcule el promedio de notas considerando las 5 mejores.
* Un método que retorna la concatenación de toda la información, como son los datos de alumno, su promedio; así como todas sus notas.

Y otros métodos que usted considere.

Diseñar la clase de prueba llamada **ProcesarNotas** que permite realizar las siguientes tareas:

* Crear un objeto referencia **objAlumno** de tipo Alumno, en el cual el código, y nombre es ingresado por pantalla.
* Crear un objeto referencia de tipo **GestionNotas**.
* Se generan las 6 notas, se calcula su promedio y se muestra toda la información.





**CASO DE NEGOCIO 05: FUTBOL INFANTIL.**

Una escuela infantil de futbol desea tener siempre actualizada la información de sus 11 jugadores de futbol, para cual se le pide:

* Diseñar una clase llamada **EquipoDeFutbol,** con variables de instancia nombre del equipo de futbol y dos arreglos de tamaño 11, que contendrán, el nombre de los jugadores y el peso de cada jugador.
* Un constructor con un parámetro, nombre del equipo y que además permita crear los dos arreglos de tamaño 11. ***SUGERENCIA: recuerde el dimensionamiento es importante para establecer recorridos de aeestablecer una estructura selectiva por cada situación***
* Un método que almacene los datos de los 11 jugadores, se ingresa los nombres y el peso de cada jugador considerando que estos pueden ser valores mayores a 29 y menores e iguales a 50. ***Sugerencia: genérelos aleatoriamente, considerando que son números enteros***.
* Un método que permita calcular y retornar en forma de cadena, la cantidad de jugadores con peso:

1. entre 30 y 35 kg inclusive
2. entre 36 y 40 kg inclusive
3. mayor a 40 kg.

* Un método que calcule y muestre los nombres de los jugadores con mayor y menor peso.
* Y otros métodos que usted considere.
* Diseñar la clase prueba llamada **UnDiaJugando,** la cual permite conocer el estado físico de su equipo de futbol. Se pide:
* Crear un objeto referencia (equipoLuchador) de **EquipoDeFutbol** cuyo nombre de equipo es ingresado por pantalla.
* Procesar el ingreso de datos de todos los jugadores del equipoLuchador.
* Mostrar un reporte de todos los cálculos pedidos. ***Sugerencia: recorra el arreglo concadenado la información, recuerde siempre de presentarlo de manera que sea fácil para el usuario visualizarlo.***