

Examen Base – Desarrollar en Java (Consola).

1) Has una clase llamada **Persona** que siga las siguientes condiciones:

- Sus atributos son: **nombre**, **edad**, **NSS (Numero de Seguro Social)**, **sexo** (H hombre, M mujer), **peso y altura**. No queremos que se accedan directamente a ellos. Piensa que modificador de acceso es el más adecuado, también su tipo. Si quieres añadir algún atributo puedes hacerlo.
- Por defecto, todos los atributos menos el NSS serán valores por defecto según su tipo (0 números, cadena vacía para String, etc.). Sexo será hombre por defecto, usa una constante para ello.
- Se implementará:
 - Un constructor con todos los atributos como parámetro.
- Los métodos que se implementaran son:
 - **calcularIMC()**: calculara si la persona está en su peso ideal ($\text{peso en kg} / (\text{Estatura}^2 \text{ en m})$), devuelve un -1 si está por debajo de su peso ideal, un 0 si está en su peso ideal y un 1 si tiene sobrepeso .Te recomiendo que uses constantes para devolver estos valores.

	Hombre	Mujer
Falta de peso	por debajo de 20	por debajo de 19
Peso normal	20-25	19-24
Sobrepeso	mayor de 25	mayor de 24

- **esMayorDeEdad()**: indica si es mayor de edad, devuelve un booleano.
- **comprobarSexo(char sexo)**: comprueba que el sexo introducido es correcto. Devolver el valor en booleano. No será visible al exterior.
- **toString()**: devuelve toda la información del objeto.
- **generaNSS()**: genera una expresión de 8 caracteres con números y letras al azar. Este método será invocado cuando se construya el objeto. Puedes dividir el método en partes para que te sea más fácil.No será visible al exterior.
- Métodos set de cada parámetro, excepto de NSS.

Ahora, crea una pantalla que haga lo siguiente:

- Pide por formulario el nombre, la edad, sexo, peso y altura.
- Deberá comprobar si está en su peso ideal, tiene sobrepeso o por debajo de su peso ideal con un mensaje.
- Indicar si es mayor de edad.
- Por último, mostrar toda la información ingresada.

Puedes usar métodos, para que te sea más fácil.

Examen Servicios REST – Java (Spring).

Desarrollar API REST con JSON que permita consumir dar de alta, consultar y modificar un cliente.

Las respuesta de la API para cada petición deberá considerar los siguientes campos base dentro de la respuesta JSON:

Cve_Error (donde 0 es exitoso y cual error es con un valor negativo).

Cve_Mensaje (la descripción del código de mensaje).

Los registros no deben permitir ni repetir nombre de usuario ni correo electrónico.

Los clientes deberán tener los siguientes campos:

Name	Type
Cliente_ID	integer
Nombre_Usuario	string
Contraseña	string
Nombre	string
Apellidos	string
Correo_Electronico	string
Edad	integer
Estatura	decimal number
Peso	decimal number
IMC	decimal number
GEB	decimal number
ETA	decimal number
Fecha_Creacion	date
Fecha_Actualizacion	date

1. Realiza el Siguiente POST para agregar un Cliente

[POST /NutriNET/Cliente](#)

Con la siguiente estructura JSON (Diferentes Valores)

```
{
  "Nombre": "Juan",
  "Apellidos": "Perez Hernandez",
  "Nombre_Usuario": "Ju",
  "Correo_Electronico": "jperez@hotmail.com",
  "Contraseña": "juanitoperez"
}
```

2. Realiza un Get (Tendrá la opción de todos o por ID)

[GET /NutriNET/Cliente](#)

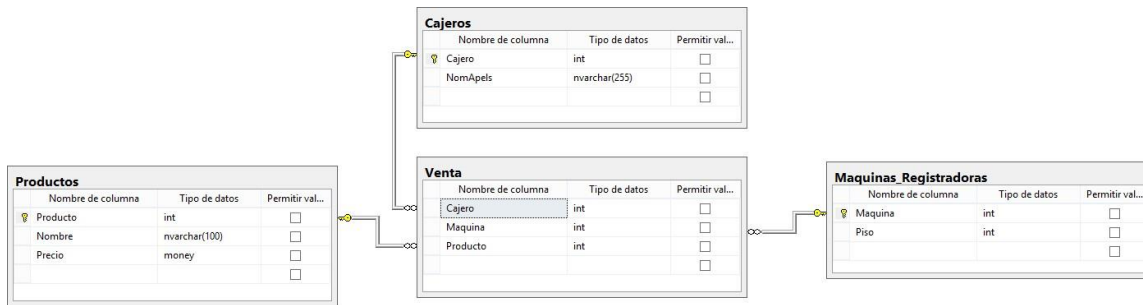
3. Realiza el siguiente PUT, para modificar el peso y la estatura de un cliente

[PUT /NutriNET/Cliente/59](#)

```
{
  "Edad": 39
  "Estatura": 1.80
  "Peso": 60
  "GEB": 1500
}
```

Examen Base de Datos - Utilizar BD MySQL (local).

LOS GRANDES ALMACENES



En base al ER generar los Querys necesarios para solventarlos:

1. Generar script para poblar todas las tablas.
2. Mostrar el número de ventas de cada producto, ordenado de más a menos ventas.
3. Obtener un informe completo de ventas, indicando el nombre del cajero que realizo la venta, nombre y precios de los productos vendidos, y el piso en el que se encuentra la máquina registradora donde se realizó la venta.
4. Obtener las ventas totales realizadas en cada piso.
5. Obtener el código y nombre de cada cajero junto con el importe total de sus ventas.
6. Obtener el código y nombre de aquellos cajeros que hayan realizado ventas en pisos cuyas ventas totales sean inferiores a los 5000 pesos.