**Sesión 2: “Prepararse para atender la problemática a resolver” – Java**

* Java básico
* Definición de la estructura de una clase
* Crear y ejecutar clases, uso de importaciones, paquetes, modificadores de acceso, clases e interfaces abstractas, static.
* Tipos de datos
* Métodos y encapsulación
* Clases del api, arrays
* Flujos de control
* Control de errores

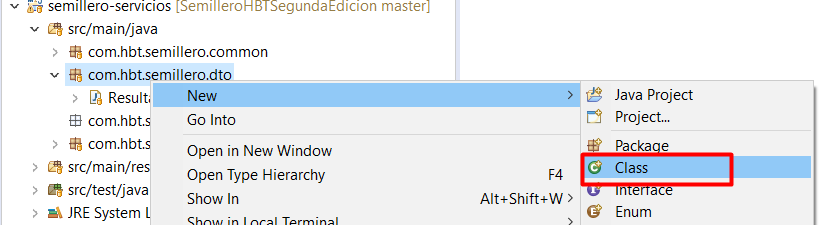
**Objetivos de la sesión:**

1. Entender el uso de datos como se pueden almacenar y manejar

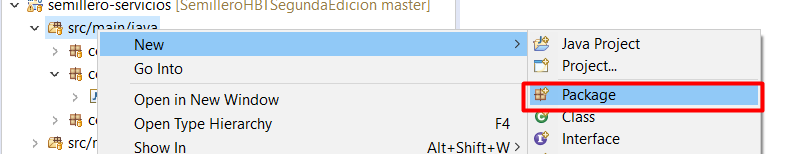
**Desarrollo de la sesión**

1. Entender el uso de datos como se pueden almacenar y manejar

* Como se va a representar los datos requeridos para manejar un comic: Para presentar los datos requeridos se va a usar una clase ComicDTO.java permite definir el contenedor de la información que debe viajar desde la capa de servicios a capa de presentación y una clase Comic.java la que va a permitir representar información que está en la base de datos en la tabla DB\_SEMILLERO.COMIC.
* Donde se va a crear DTO: en el paquete ya existente “com.hbt.semillero.dto” crear una nueva clase que se llame Comic.java



* Donde se va a crear la clase: Se debe crear un nuevo paquete llamado “com.hbt.semillero.entidad”



* Que atributos debe tener la clase Comic.java: Inicialmente se va a crear los siguientes atributos:

**private** String id;

**private** String nombre;

**private** String editorial;

**private** String tematica;

**private** String coleccion;

**private** Integer numeroPaginas;

**private** BigDecimal precio;

**private** String autores;

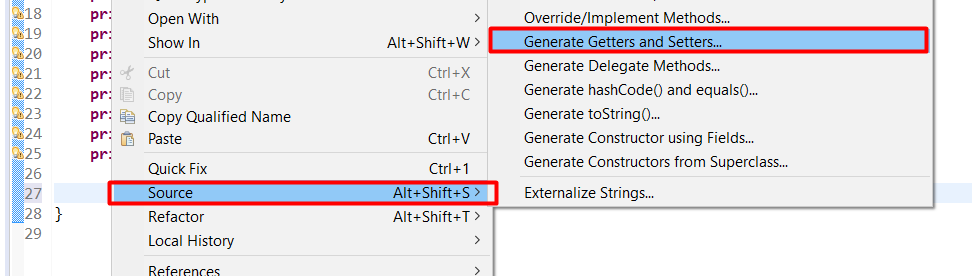
**private** Boolean color;

**private** LocalDate fechaVenta;

**private** String estado;

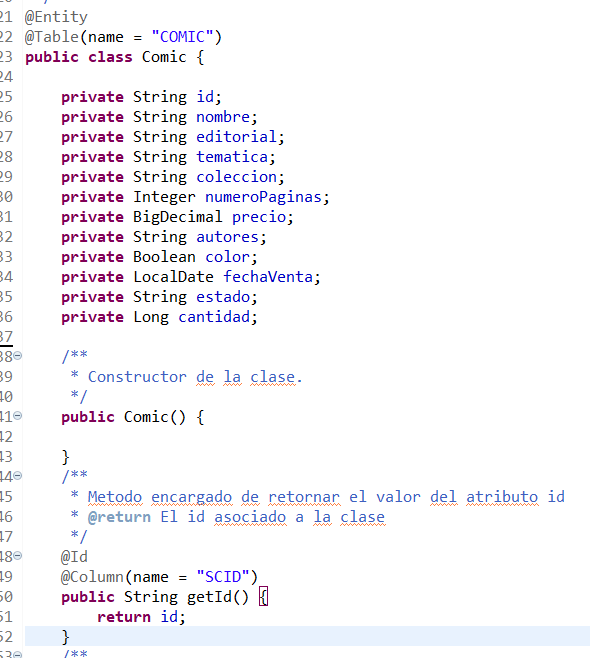
**private** Long cantidad;

* ¿Como se puede acceder a un atributo privado? Se deben crear métodos set (para poner el valor de un dato) y get (para obtener el valor del atributo): Estos métodos se pueden autogenerar, dando clic derecho en la clase y seleccionando lo siguiente:



* Como se crea una entidad: A diferencia de los DTO’s las entidades son clases con mas significado ya que requieren usar la librería javax.persistence por medio de sus métodos permite mapear la tabla, sus campos y más funcionalidades. Se debe tener presente las siguientes reglas para crear una entidad

La clase debe estar anotada con @Entity, se debe tener un constructor (el constructor por defecto es suficiente), si la tabla tiene un nombre diferente a la entidad se debe colocar la anotación @Table(name = "COMIC"), cada campo de la tabla debe estar mapeado o puesto en la entidad con su correspondiente anotación @Column(name = " SCID")



* Como se puede obtener un dato almacenado preveniente para ser manipulado en la capa de negocio en java: Se puede utilizar la librería javax.persistence la cual brinda las clases y métodos indicados para usar la persistencia.
* Completando la entidad, uso de secuencias….
* Como se puede trabajar con los campos que tienen condiciones SCTEMATICA y SCESTADO explicación enumeraciones, cambiar nombre de atributo temática por tematicaEnum, crear el tipo enum TematicaEnum, cambiar las firmas de los métodos get y set agregar anotación

Manejo de fechas

Agregar dependencia

Agregar convertidor

https://thoughts-on-java.org/persist-localdate-localdatetime-jpa/

https://thoughts-on-java.org/hibernate-5-date-and-time/

**Ejercicio sesión 2**

**Conceptos adicionales**

* “Data Transfer Object (DTO) – Patrón de diseño, tiene como finalidad de crear un objeto plano (POJO) con una serie de atributos que puedan ser enviados o recuperados del servidor en una sola invocación, de tal forma que un DTO puede contener información de múltiples fuentes o tablas y concentrarlas en una única clase simple.”
* Entidades son clases que representa al modelo de datos
* Las capas de negocio, presentación permiten determinar una división lógica de las tecnologías y artefactos a usar en la construcción de aplicaciones.

**Enlaces de interés**

* Documento “Material-JPA.docx”