

**-Parse:** Es la clase Main, contiene una instancia de BD y es por la que entran los datos que se almacenarán en la Base de Datos.

**-BaseDeDatos:** Es la clase que modela las tablas, posee como atributo un ArrayList donde las almacena.

**-BaseDeDatos ():** Es el constructor de la clase.

**-addRegistro (Registro registro):** es el método que se encarga de añadir los registros a las tablas, comprueba sus estructuras y determina si almacenarlos en una u otra tabla o si debe crear una de acuerdo a la estructura.

**-printAll ():** es el método que se encarga de imprimir todos los datos almacenados en todas las

Tablas.

**-Registro:** Es la clase que modela los registros, posee como atributos dos ArrayList, uno para los datos y otro para los tipos de datos. Además de los getters y el constructor de la clase, posee los siguientes métodos:

**-crearRegistro (Object…dato):** recibe una colección de datos y les asigna una estructura, creando así un registro.

**-imprimirLosDatos ():** como su nombre lo indica, imprime los datos en el registro.

**-Tabla:** Es la clase que modela las tablas, posee como atributos tipoDeDatos y registros, dos ArrayList, respectivamente. Además de los constructores y los getters y los métodos para imprimir, posee los siguientes métodos:

**-addRegistro (Registro registro**): es el mismo método usado en la clase BaseDeDatos, al ser e estático puede ser usado sin instancia.

**-tipoDeDatosEsIgual (ArrayList<TipoDeDatos> tipoDeDatos):** determina si un registro tiene una estructura igual a las tablas existentes para determinar si se debe o no crear una nueva con diferente estructura.

**-TipoDeDatos:** es una enumeración que tiene los tipos de datos aceptados por las tablas.

**-retornarTipoDeDato (Object o):** haciendo uso del operador “instanceof”, determina qué tipo de dato es el objeto que se le manda como parámetro y devuelve su respectiva representación en el tipo enumerado.