```
           1..1          1..1
┌─────────────────────┐ ┌─────────────────────────┐        product              warranty
│ class Product {     │ │ class Warranty {        │      ┌──────────────────┐  ┌────────────────────────────┐
│   int productNo;    │ │   int warrantyNo;       │      │ product_no (pk)  │  │ warranty_no (pk)           │
│   String productName;│ │   LocalDate issuedDate; │      │ product_nm       │  │ issued_dt                  │
│   String manufacturer;│ │   int days;            │      │ manufacturer     │  │ days                       │
│   String category;  │ │   String termsAndConditions;│  │ category         │  │ terms_and_conditions       │
│   double price;     │ │   String customerName;  │      │ price            │  │ customer_nm                │
│   Warranty warranty;│ │   Product product;      │      └──────────────────┘  │ product_no (fk) (unique)   │
│                     │ │ }                       │                            └────────────────────────────┘
│ }                   │ └─────────────────────────┘
└─────────────────────┘
```

Mapping Association Relationship:
1. for all the primitive attributes of an entity class, we need to directly store/retrieve them from corresponding columns of its table.
2. for associated objects inside an entity class
   - we map them by writing foreign key in the tables

To map association object we need to tell 2 things
1. what is the foreign key column
2. where is the foreign key column (is the foreign key column existing in entity class that has the associated object or other side?)

Warranty (W1). = in relational model we need to write primary key of one object as foreign key in another object to map the association

|-Product (P1)

```
<class name="Warranty" table="warranty">
  <id name="warrantyNo" column="warranty_no">
    <generator class="increment"/>
  </id>
  <property …/>
  <many-to-one name="product">
    <column name="product_no" class="Product" unique="true"/>
  </many-to-one>

</class>
```