**Ship Movement**

Step 1: mapping of regular entities: SHIP, SHIP\_TYPE, STATE/COUNTRY AND SEA/OCEAN/LAKE.

* + Regular entity SHIP mapped as SHIP relation. All simple attributes are included. Attribute “Sname” is chosen as a primary key.
  + Regular entity SHIP\_TYPE mapped as SHIP\_TYPE relation. All simple attributes are included. Attribute “Type” is chosen as a primary key.
  + Regular entity STATE/COUNTRY mapped as STATE/COUNTRY relation. All simple attributes are included. Attribute “Name” is chosen as a primary key.
  + Regular entity SEA/OCEAN/LAKE mapped as SEA/OCEAN/LAKE relation. Attribute “Name” is chosen as a primary key.

Step 2: mapping of weak entities: SHIP\_MOVEMENT, PORT AND PORT\_VISIT.

* + Weak entity SHIP\_MOVEMENT mapped as SHIP\_MOVEMENT relation. All simple attributes are included. Primary key “Sname” from SHIP relation is added as Foreign Key. The primary key the combination of “Sname” and “Time\_Stamp”.
  + Weak entity PORT mapped as PORT relation. Attribute “Name” is added as Foreign key from “STATE/COUNTRY” and “SEA/OCEAN/LAKE” relations. The primary key is the combination of “Name” and “Pname”.
  + Weak entity PORT\_VISIT mapped as PORT\_VISIT relation. All simple attributes are included. Attribute “Sname” is added as foreign key from “SHIP” relation. The primary key is the combination of attribute “Sname” and “Start\_date”.

Step 3: mapping of binary 1:1 relationship: PORT\_VISIT

* + 1:1 relationship TYPE is mapped as a foreign key attribute “Start\_Date” in SHIP relation that corresponds to “Start\_Date” partial key attribute in PORT\_VISIT relation.

Step 4: mapping of binary 1:N relationship: TYPE, ON, IN, and HOME\_PORT.

* + 1:N relationship TYPE is mapped as a foreign key attribute “Type” in SHIP relation (“N” side) that corresponds to “Type” primary key attribute in SHIP\_TYPE relation (“1” side)
  + 1:N relationship IN is mapped as a foreign key attribute “Name” in PORT relation (“N” side) that corresponds to “Name” primary key attribute in STATE/COUNTRY relation (“1” side)
  + 1:N relationship ON is mapped as a foreign key attribute “Name” in PORT relation (“N” side) that corresponds to “Name” primary key attribute in SEA/OCEAN/LAKE relation (“1” side)
  + 1:N relationship HOME\_PORT is mapped as a foreign key attribute “Pname” in SHIP relation (“N” side) that corresponds to “Pname” primary key attribute in PORT relation (“1” side)

Step 5: mapping of binary M:N relationship: no relationship Step 6: mapping of multivalued attributes: no multivalued attribute. Step 7: mapping of n-ary relationship: no n-ary relationship

1. [9:45 PM]

Can someone have a look at the writeup for question 1

**Car dealership**

Step 1: Map regular entity types: VEHICLE, SALESPERSON, and CUSTOMER.

* + Regular entity VEHICLE mapped as VEHICLE relation. All simple attributes are included. Attribute “VIN” is chosen as a primary key.
  + Regular entity SALESPERSON mapped as SALESPERSON relation. All simple attributes are included. Attribute “SID” is chosen as a primary key.
  + Regular entity CUSTOMER mapped as CUSTOMER relation. All simple attributes are included. Attribute “Ssn” is chosen as a primary key.

Step 2: no weak entity types.

Step 3: no 1:1 relationships.

Step 4: Establish relationship between VEHICLE (S) and CUSTOMER (T).

* + Added SSN as a foreign key attribute in VEHICLE, relating to Ssn in CUSTOMER.
  + Added Sid as an attribute in VEHICLE, relating to the primary key Sid in SALESPERSON.

Step 5: there are no many-to-many relationships.

Step 6: there are no multi-valued attributes.

Step 7: Create the entity SALE with attributes Date, VIN, SSN, and SID.

* + VIN is a foreign key relating to VEHICLE.
  + SSN is a foreign key relating to CUSTOMER.
  + SID is a foreign key relating to SALESPERSON.
  + The combination of these foreign keys forms the primary key for SALE.

Step 8-a: Create subclasses CAR, TRUCK, and SUV inheriting from VEHICLE.

* + CAR: Engine\_Size, VIN (foreign key to VEHICLE).
  + TRUCK: Tonnage, VIN (foreign key to VEHICLE).
  + SUV: No\_Seats, VIN (foreign key to VEHICLE).
  + The combination of Engine\_Size and VIN (in CAR), Tonnage and VIN (in TRUCK), or No\_Seats and VIN (in SUV) serves as the primary key for each subclass.

In summary, the entity relationship model includes regular entities such as VEHICLE, SALESPERSON, and CUSTOMER. The relationships are established between VEHICLE and CUSTOMER, with foreign keys SSN and Sid. Additionally, the SALE entity is created to capture sales information with foreign keys relating to VEHICLE, CUSTOMER, and SALESPERSON. CAR, TRUCK, and SUV are subclasses of VEHICLE, each having their own specific attributes and primary key combinations.