

## Entities & Attributes

### Customer

- customer\_id (PK)
- cust\_name
- cust\_email
- cust\_phone
- cust\_address

### Manufacturer

- manufacturer\_id (PK)
- manufacturer\_name
- manufacturer\_website

### Product

- product\_id (PK)
- manufacturer\_id (FK)
- model\_type
- model\_price
- min\_order\_qty
- scale
- product\_name

### Inventory

- product\_id (PK, FK)
- qoh
- reorder\_level

### Invoice

- invoice\_number (PK)
- customer\_id (FK)

- invoice\_date
- total\_amount
- shipping\_charge
- invoice\_status

#### Line

- invoice\_number (FK)
- line\_number (PK)
- product\_id (FK)
- line\_units
- unit\_price
- line\_status
- ship\_date

#### Shipment

- shipment\_id (PK)
- invoice\_number (FK)
- line\_number (FK)
- qty\_shipped
- ship\_date
- tracking\_number
- shipment\_status

#### Constraints

- Each customer\_id, manufacturer\_id, product\_id, invoice\_number, shipment\_id, and line\_number must be unique, not null, and numeric.
- cust\_phone must have 10 digits.
- model\_price, unit\_price, and total\_amount must be greater than or equal to 0.
- line\_units, min\_order\_qty, and qty\_shipped must be positive integers.
- invoice\_status ∈ {Open, Partially Shipped, Closed}.

- `line_status` ∈ {Backordered, Partial, Fulfilled}.
- `shipment_status` ∈ {Created, Shipped, Delivered}.
- `model_type` ∈ {Car, Aircraft, Ship, Decal}.
- `scale` must match one of the standard values (1/144, 1/100, 1/72, 1/48, 1/35, 1/32, 1/24).
- `manufacturer_id` in `Product` must exist in `Manufacturer`.
- `product_id` in `Inventory`, `Line`, and `Shipment` must exist in `Product`.
- `customer_id` in `Invoice` must exist in `Customer`.
- `invoice_number` in `Line` and `Shipment` must exist in `Invoice`.
- `qty_shipped` for a line may not exceed `line_units`.
- A shipment's `ship_date` must be on or after `invoice_date`.
- If  $qoh \leq reorder_level$ , system triggers reorder of at least `min_order_qty`.

#### Business Rules

1. A customer may generate many invoices.
2. Each invoice belongs to one customer.
3. Each invoice can have many lines (products ordered).
4. Each line belongs to one invoice and references one product.
5. A product can appear on many lines.
6. A manufacturer can produce many products.
7. Each product has one inventory record.
8. Each invoice can have multiple shipments.
9. Each line can have multiple shipments (for partial delivery).
10. Backorders are handled by comparing `line_units` vs. `qty_shipped`.
11. Customers are billed only when items ship.
12. Line and invoice statuses update automatically based on shipments.

#### Assumptions

- The invoice acts as the “order header.” No separate order table is needed.

- Shipment table captures partial deliveries.
- Inventory tracks current quantity (qoh) and triggers reorder using reorder\_level and min\_order\_qty.
- Decals and models share the same Product table, distinguished by model\_type.
- Prices are snapshotted in the Line table and do not change with later price updates.
- CC Bank (credit card processor) is external and not part of this model.
- One-to-one: Product ↔ Inventory
- One-to-many: Customer ↔ Invoice, Manufacturer ↔ Product, Invoice ↔ Line, Line ↔ Shipment