```sql
/* This table describes the properties of a supplier with supplier id,
supplier name and zip code for potential distance calculations. */
CREATE TABLE Supplier(
    sid varchar2(35),
    sname varchar2(35),
    zipcode varchar2(6),
    PRIMARY KEY (sid)
);

/* This table describes properties of items throughout the logistic
chain, including product id, product name, dimensions, weight,
handling instructions, customer rating and complaints filed against
it. The supplier's storage quota for its products are also labelled.
*/
CREATE TABLE Item(
    pid varchar2(35),
    pname varchar2(35),
    length INTEGER,
    width INTEGER,
    height INTEGER,
    weight FLOAT,
    handling varchar2(35),
    rating FLOAT,
    complaint VARCHAR2(35),
    mfgspaceavai INTEGER,
    mfgspaceused INTEGER,
    PRIMARY KEY (pid),
    CHECK (mfgspaceavai>=0 and mfgspaceused>=0)
);

/* This table records contracts signed between a supplier and the
retailer for a certain product, including contract id, supplier id,
product id, quantity, price and estimated date of delivery. */
CREATE TABLE Contract(
    cid varchar2(35),
    sid varchar2(35) NOT NULL,
    pid varchar2(35) NOT NULL,
    quantity INTEGER,
    price FLOAT,
    deliverdate DATE,
    PRIMARY KEY (cid),
```

```
    FOREIGN KEY (sid) REFERENCES Supplier,
    FOREIGN KEY (pid) REFERENCES Item,
    CHECK (quantity>0)
);

/* This table depicts the management of transportation resources,
including the ids of vehicle stations, with its location, name of
manager, operating hours and phone number. */
CREATE TABLE TranspMgmt(
    stnid varchar2(35),
    location varchar2(35),
    stnmgr varchar2(35),
    ophour varchar2(35),
    phone varchar2(35),
    PRIMARY KEY (stnid)
);

/* This table tracks status of vehicle fleets during transportation,
including fleet id, station id that the fleet belongs to, the type of
vehicle possessed by this fleet, the total number of vehicles and the
number of vehicles that are sent out. */
CREATE TABLE Fleet(
    fid varchar2(35),
    stnid varchar2(35) NOT NULL,
    vehicletype varchar2(35),
    vehiclenum INTEGER,
    vehiclesent INTEGER,
    PRIMARY KEY (fid),
    FOREIGN KEY (stnid) REFERENCES TranspMgmt ON DELETE CASCADE
);

/* This table tracks status of labor in terms of squads during
transportation, including squad id, station id that the squad belongs
to, the type of labor that makes the squad, the total number of labors
and the number of labors sent out. */
CREATE TABLE Squad(
    sqid varchar2(35),
    stnid varchar2(35) NOT NULL,
    labortype varchar2(35),
    labornum INTEGER,
    laborsent INTEGER,
```

```
    PRIMARY KEY (sqid),
    FOREIGN KEY (stnid) REFERENCES TranspMgmt ON DELETE CASCADE
);


/* This table stores information about a storage site which could be
either a warehouse or a retail store, including storage id, its
location, the number of employees, name of the manager and phone
number. */
CREATE TABLE Storage(
    stoid varchar2(35),
    location varchar2(35),
    zipcode varchar2(6),
    numemployee INTEGER,
    manager varchar2(35),
    phone varchar2(35),
    PRIMARY KEY (stoid)
);


/* This table records status of a warehouse-type storage site,
including storage id, shelf id and description which distinguishes
locations for different types of goods stored (e.g. clothes,
electronics, referagerated foods etc), the space available for each
shelf and the space already used. */
CREATE TABLE Warehouse(
    stoid varchar2(35),
    shelfid varchar2(35),
    shelfdesc varchar2(35),
    spaceavai INTEGER,
    spaceused INTEGER,
    PRIMARY KEY (stoid, shelfid),
    FOREIGN KEY (stoid) REFERENCES Storage ON DELETE CASCADE,
    CHECK (spaceavai>=0 and spaceused>=0)
);


/* This table records individual retailer information besides those
properties listed in the storage table, such as storage id, available
and used space in store, and average volume of customers it serves.
*/
CREATE TABLE Retailer(
    stoid varchar2(35),
    spaceavai INTEGER,
```

```
    spaceused INTEGER,
    customervol INTEGER,
    PRIMARY KEY (stoid),
    FOREIGN KEY (stoid) REFERENCES Storage ON DELETE CASCADE,
    CHECK (spaceavai>=0 and spaceused>=0)
);
```

/* This table keeps record of every transportation from suppliers to warehouses, containing delivery id, supplier id, warehouse storage id, date of delivery and arrival, number of vehicles used and from which fleet, number of labor and the squad they are from, and overall transportation costs. */

```
CREATE TABLE SupplierDelivery(
    did varchar2(35),
    sid varchar2(35),
    stoid varchar2(35),
    deliverdate DATE,
    arrivedate DATE,
    vehiclenum INTEGER,
    fid varchar2(35),
    labornum INTEGER,
    sqid varchar2(35),
    cost FLOAT,
    PRIMARY KEY (did),
    FOREIGN KEY (sid) REFERENCES Supplier,
    FOREIGN KEY (stoid) REFERENCES Storage,
    FOREIGN KEY (fid) REFERENCES Fleet,
    FOREIGN KEY (sqid) REFERENCES Squad
);
```

/* This table records every transportation that happens between warehouses, containing delivery id, from and to warehouse storage ids, with other fields similar to the previous table. */

```
CREATE TABLE WarehouseTransit(
    did varchar2(35),
    fromstoid varchar2(35),
    tostoid varchar2(35),
    deliverdate DATE,
    arrivedate DATE,
    vehiclenum INTEGER,
    fid varchar2(35),
```

```
    labornum INTEGER,
    sqid varchar2(35),
    cost FLOAT,
    PRIMARY KEY (did),
    FOREIGN KEY (fromstoid) REFERENCES Storage(stoid),
    FOREIGN KEY (tostoid) REFERENCES Storage(stoid),
    FOREIGN KEY (fid) REFERENCES Fleet,
    FOREIGN KEY (sqid) REFERENCES Squad
);

/* This table keeps record of transportation from warehouses to
retailers, containing delivery id, warehouse storage id, retailer
storage id, etc. */
CREATE TABLE WtoR(
    did varchar2(35),
    fromstoid varchar2(35),
    tostoid varchar2(35),
    deliverdate DATE,
    arrivedate DATE,
    vehiclenum INTEGER,
    fid varchar2(35),
    labornum INTEGER,
    sqid varchar2(35),
    cost FLOAT,
    PRIMARY KEY (did),
    FOREIGN KEY (fromstoid) REFERENCES Storage(stoid),
    FOREIGN KEY (tostoid) REFERENCES Storage(stoid),
    FOREIGN KEY (fid) REFERENCES Fleet,
    FOREIGN KEY (sqid) REFERENCES Squad
);

/* This table records every transportation that happens between
retailer stores. */
CREATE TABLE RetailerTransit(
    did varchar2(35),
    fromstoid varchar2(35),
    tostoid varchar2(35),
    deliverdate DATE,
    arrivedate DATE,
    vehiclenum INTEGER,
    fid varchar2(35),
```

```
    labornum INTEGER,
    sqid varchar2(35),
    cost FLOAT,
    PRIMARY KEY (did),
    FOREIGN KEY (fromstoid) REFERENCES Storage(stoid),
    FOREIGN KEY (tostoid) REFERENCES Storage(stoid),
    FOREIGN KEY (fid) REFERENCES Fleet,
    FOREIGN KEY (sqid) REFERENCES Squad
);

/* This table depicts the smallest unit of goods that is involved
during transportation, having unique bid for each batch of each type
of item, with the contract id that is related with delivery of this
batch of item, the date of expiration (if applicable), quantity of
items in this batch, and current location plus all transportation
references. */
CREATE TABLE Batch(
    bid varchar2(35),
    pid varchar2(35) NOT NULL,
    cid varchar2(35) NOT NULL,
    expiration DATE,
    quantity INTEGER,
    stoid varchar2(35),
    intransit INTEGER,
    sold INTEGER,
    did1 VARCHAR(20) NOT NULL,
    did2 VARCHAR(20),
    did3 VARCHAR(20),
    did4 VARCHAR(20),
    UNIQUE(cid),
    PRIMARY KEY (bid),
    FOREIGN KEY (pid) REFERENCES Item,
    FOREIGN KEY (cid) REFERENCES Contract,
    FOREIGN KEY (did1) REFERENCES SupplierDelivery(did),
    FOREIGN KEY (did2) REFERENCES WarehouseTransit(did),
    FOREIGN KEY (did3) REFERENCES WtoR(did),
    FOREIGN KEY (did4) REFERENCES RetailerTransit(did),
    CHECK (intransit=0 or intransit = 1),
    CHECK (sold=0 or sold = 1),
    CHECK (quantity>0)
);
```

```
/* The table stores data about customers who purchase products from
retail stores, including name, sex, age, and zip code. */
CREATE TABLE Customer(
    cuid varchar2(20),
    name varchar2(35),
    sex varchar2(6),
    age INTEGER,
    zipcode varchar2(6),
    PRIMARY KEY (cuid),
    CHECK (sex='M' or sex='F' or sex='U'),
    CHECK (age>10 and age<100)
);

/* This table records each purchase history, including customer id,
batch id of the product, date of purchase and quantity. */
CREATE TABLE Purchase(
    cuid varchar2(20),
    bid varchar2(35),
    datepur DATE,
    quantity INTEGER,
    PRIMARY KEY (cuid,bid,datepur),
    FOREIGN KEY (cuid) REFERENCES Customer,
    FOREIGN KEY (bid) REFERENCES Batch,
    CHECK (quantity>0)
);

/* This table records a product being returned to a retail store due
to some complaints, together with customer id, batch id, date and
quantity. */
CREATE TABLE Return(
    cuid varchar2(20),
    bid varchar2(35),
    daterec DATE,
    quantity INTEGER,
    complaint varchar2(35),
    stoid varchar2(35),
    PRIMARY KEY (cuid,bid,daterec),
    FOREIGN KEY (cuid) REFERENCES Customer,
    FOREIGN KEY (bid) REFERENCES Batch,
    FOREIGN KEY (stoid) REFERENCES Retailer,
```

```sql
        CHECK (quantity>0)
);


/* This table records products that are sent to retail store and then
to supplier for repairing, including customer id, batch id, date the
store received, quantity, problem with the item, date sent to the
supplier and the date sent back to the store. */
CREATE TABLE Repair(
    cuid varchar2(20),
    bid varchar2(35),
    daterec DATE,
    quantity INTEGER,
    problem varchar2(35) NOT NULL,
    datetosupplier DATE,
    dateretrieve DATE,
    PRIMARY KEY (cuid,bid,daterec),
    FOREIGN KEY (cuid) REFERENCES Customer,
    FOREIGN KEY (bid) REFERENCES Batch,
    CHECK (quantity>0)
);


/* This table shows when a product is being recycled from a customer,
how the condition of the article leads to different processing
decisions and the date of execution. */
CREATE TABLE Recycler(
    cuid varchar2(20),
    bid varchar2(35),
    daterec DATE,
    quantity INTEGER,
    condition varchar2(35),
    decision varchar2(35),
    execdate DATE,
    PRIMARY KEY (cuid,bid,daterec),
    FOREIGN KEY (cuid) REFERENCES Customer,
    FOREIGN KEY (bid) REFERENCES Batch,
    CHECK    (decision='disassemble'    or    decision='refurbish'    or
decision='discard'),
    CHECK (quantity>0)
);


/* THis table shows a list of raw materials and parts provided to a
```

supplier with price, unit dimensions and weight. The table also contains the information on how much space is used and available in the supplier's inventory. */

```sql
CREATE TABLE MatlSto(
    mid varchar2(35),
    matltype varchar2(35),
    price FLOAT,
    length FLOAT,
    width FLOAT,
    height FLOAT,
    weight FLOAT,
    spaceavai INTEGER,
    spaceused INTEGER,
    PRIMARY KEY (mid),
    CHECK (spaceavai>=0 and spaceused>=0)
);
```

/* This table describes how a product is disassembled into usable raw materials and parts with respective quantities. */

```sql
CREATE TABLE Disassemble(
    pid varchar2(35),
    mid varchar2(35),
    quantity FLOAT,
    PRIMARY KEY (pid,mid),
    FOREIGN KEY (pid) REFERENCES Item,
    FOREIGN KEY (mid) REFERENCES MatlSto,
    CHECK (quantity>0)
);
```

/* This table shows how a supplier manages raw material and parts demands based on product and corresponding contract to determine the amount needed. */

```sql
CREATE TABLE MfgPlan(
    pid varchar2(35),
    cid varchar2(35),
    mid varchar2(35),
    quantity INTEGER,
    PRIMARY KEY (pid,cid,mid),
    FOREIGN KEY (pid) REFERENCES Item ON DELETE CASCADE,
    FOREIGN KEY (cid) REFERENCES Contract ON DELETE CASCADE,
    FOREIGN KEY (mid) REFERENCES MatlSto,
```

```
    CHECK (quantity>0)
);

/* This table summarizes the material demand for a supplier, namely
the date and quantity for each type of material or part. */
CREATE TABLE StockPlan(
    mid varchar2(35),
    dateneed DATE,
    quantity INTEGER,
    PRIMARY KEY (mid,dateneed),
    FOREIGN KEY (mid) REFERENCES MatlSto ON DELETE CASCADE,
    CHECK (quantity>0)
);

/* This table stores the history of raw material/part delivery to the
supplier, including delivery id, material id, quantity, deliver date,
arrival date, number of vehicles used and the fleet they belong to,
number of labors involved and their squad id, finally the cost of
delivery. */
CREATE TABLE StockDelivery (
    did varchar2(35),
    mid varchar2(35),
    quantity INTEGER,
    deliverdate DATE,
    arrivedate DATE,
    vehiclenum INTEGER,
    fid varchar2(35),
    labornum INTEGER,
    sqid varchar2(35),
    cost FLOAT,
    PRIMARY KEY (did),
    FOREIGN KEY (mid) REFERENCES MatlSto,
    FOREIGN KEY (fid) REFERENCES Fleet,
    FOREIGN KEY (sqid) REFERENCES Squad,
    CHECK (quantity>0)
);
```