```sql
-- Create Users Table
CREATE TABLE Users (
    UserID NUMBER PRIMARY KEY,
    Name VARCHAR2(100) NOT NULL,
    Email VARCHAR2(100) NOT NULL,
    Password VARCHAR2(100) NOT NULL,
    RegistrationDate DATE DEFAULT SYSDATE
);

-- Create Auctions Table
CREATE TABLE Auctions (
    AuctionID NUMBER PRIMARY KEY,
    Title VARCHAR2(200) NOT NULL,
    Description VARCHAR2(1000),
    StartDate DATE DEFAULT SYSDATE,
    EndDate DATE NOT NULL,
    Status VARCHAR2(20) DEFAULT 'ACTIVE',
    CreatorID NUMBER REFERENCES Users(UserID)
);

-- Create Bids Table
CREATE TABLE Bids (
    BidID NUMBER PRIMARY KEY,
    AuctionID NUMBER REFERENCES Auctions(AuctionID),
    UserID NUMBER REFERENCES Users(UserID),
    BidAmount NUMBER(10,2) NOT NULL,
    BidTime TIMESTAMP DEFAULT SYSTIMESTAMP
);

-- Create Watchlist Table
CREATE TABLE Watchlist (
    WatchlistID NUMBER PRIMARY KEY,
    UserID NUMBER REFERENCES Users(UserID),
    AuctionID NUMBER REFERENCES Auctions(AuctionID),
    AddedDate DATE DEFAULT SYSDATE
);

-- Create Winners Table
CREATE TABLE Winners (
    WinnerID NUMBER PRIMARY KEY,
    AuctionID NUMBER REFERENCES Auctions(AuctionID),
    UserID NUMBER REFERENCES Users(UserID),
    WinningBid NUMBER(10,2) NOT NULL,
    DeclaredTime TIMESTAMP DEFAULT SYSTIMESTAMP
);
```

```
-- Insert Dumb Data into Users Table
INSERT INTO Users VALUES (1, 'John Doe', 'john.doe@example.com', 'password123',
SYSDATE);
INSERT INTO Users VALUES (2, 'Jane Smith', 'jane.smith@example.com', 'password456',
SYSDATE);
INSERT INTO Users VALUES (3, 'Alice Johnson', 'alice.johnson@example.com',
'password789', SYSDATE);
INSERT INTO Users VALUES (4, 'Bob Brown', 'bob.brown@example.com', 'password321',
SYSDATE);
INSERT INTO Users VALUES (5, 'Charlie Davis', 'charlie.davis@example.com',
'password654', SYSDATE);

-- Insert Dumb Data into Auctions Table
INSERT INTO Auctions VALUES (1, 'Old Painting', 'A vintage oil painting from 18th
century', SYSDATE-10, SYSDATE+5, 'ACTIVE', 1);
INSERT INTO Auctions VALUES (2, 'Vintage Car', 'Classic car from the 1960s',
SYSDATE-8, SYSDATE+6, 'ACTIVE', 2);
INSERT INTO Auctions VALUES (3, 'Rare Book', 'First edition of a famous novel',
SYSDATE-12, SYSDATE+2, 'CLOSED', 3);
INSERT INTO Auctions VALUES (4, 'Diamond Necklace', 'A beautiful diamond necklace',
SYSDATE-5, SYSDATE+3, 'ACTIVE', 4);
INSERT INTO Auctions VALUES (5, 'Antique Vase', 'A porcelain vase from the Ming
Dynasty', SYSDATE-3, SYSDATE+4, 'ACTIVE', 5);

-- Insert Dumb Data into Bids Table
INSERT INTO Bids VALUES (1, 1, 2, 1000, SYSTIMESTAMP-2);
INSERT INTO Bids VALUES (2, 1, 3, 1500, SYSTIMESTAMP-1);
INSERT INTO Bids VALUES (3, 2, 1, 5000, SYSTIMESTAMP-3);
INSERT INTO Bids VALUES (4, 2, 4, 5500, SYSTIMESTAMP-4);
INSERT INTO Bids VALUES (5, 3, 5, 300, SYSTIMESTAMP-1);
INSERT INTO Bids VALUES (6, 3, 2, 700, SYSTIMESTAMP-2);
INSERT INTO Bids VALUES (7, 4, 3, 12000, SYSTIMESTAMP-5);
INSERT INTO Bids VALUES (8, 4, 1, 13000, SYSTIMESTAMP-6);
INSERT INTO Bids VALUES (9, 5, 4, 2000, SYSTIMESTAMP-7);
INSERT INTO Bids VALUES (10, 5, 2, 2500, SYSTIMESTAMP-8);

-- Insert Dumb Data into Watchlist Table
INSERT INTO Watchlist VALUES (1, 1, 1, SYSDATE);
INSERT INTO Watchlist VALUES (2, 2, 2, SYSDATE);
INSERT INTO Watchlist VALUES (3, 3, 3, SYSDATE);
INSERT INTO Watchlist VALUES (4, 4, 4, SYSDATE);
INSERT INTO Watchlist VALUES (5, 5, 5, SYSDATE);

-- Insert Dumb Data into Winners Table
```

```
INSERT INTO Winners VALUES (1, 1, 2, 1500, SYSTIMESTAMP-1);
INSERT INTO Winners VALUES (2, 2, 4, 5500, SYSTIMESTAMP-2);
INSERT INTO Winners VALUES (3, 3, 5, 700, SYSTIMESTAMP-3);
INSERT INTO Winners VALUES (4, 4, 1, 13000, SYSTIMESTAMP-4);
INSERT INTO Winners VALUES (5, 5, 3, 2500, SYSTIMESTAMP-5);

-- Create Sequences
CREATE SEQUENCE SEQ_BID_ID START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
CREATE SEQUENCE WATCHLIST_SEQ START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
CREATE SEQUENCE bid_seq START WITH 100 INCREMENT BY 1;

-- Create Triggers
CREATE OR REPLACE TRIGGER TRG_BI_BIDS
BEFORE INSERT ON Bids
FOR EACH ROW
BEGIN
    IF :NEW.BIDID IS NULL THEN
        :NEW.BIDID := SEQ_BID_ID.NEXTVAL;
    END IF;
END;

CREATE OR REPLACE TRIGGER trg_watchlist_id
BEFORE INSERT ON Watchlist
FOR EACH ROW
BEGIN
    SELECT NVL(MAX(WatchlistID), 0) + 1 INTO :NEW.WatchlistID FROM Watchlist;
END;

COMMIT;
```