

Relational Schema:

Approach with Collapse Down → for ISA (Rest Merge Rule):

Admin(admin_login: varchar(15) primary key, email_address: varchar(50), password: varchar(16), CustomerRep_login: varchar(15), foreign key(CustomerRep_login) references CustomerRep NOT NULL)

CustomerRep(CustomerRep_login: varchar(15) primary key, email_address: varchar(50), password: varchar(16))

EndUser(endUser_login: varchar(15) primary key, email_address: varchar(50), password: varchar(16), isBuyer: boolean, isSeller: boolean, Item.ID: int, BidAlert: text, foreign key(Item.ID) references Items)

Question(question.ID: int primary key, answer_text: text, question_text: text, CustomerRep_login: varchar(15), endUser_login: varchar(15), foreign key(CustomerRep_login) references CustomerRep NOT NULL, foreign key(endUser_login) references EndUser NOT NULL)

Bid(Bid.ID: int primary key, Bid_Date: date, Bid_time: time, Bid_Amount: double, Auction.ID: int, endUser_login: varchar(15), foreign key(Auction.ID) references Auction NOT NULL, foreign key(endUser_login) references EndUser NOT NULL)

Auction(Auction.ID: int primary key, Current_Price: double, Auction_Closing_Date: date, Auction_Closing_time: time, Bid_Increment: double, Initial_Price: double, Minimum: double, Upper_Limit: double, Winner: varchar(50), endUser_login: varchar(15), Item.ID: int, foreign key (Item.ID) references Items NOT NULL, foreign key (endUser_login) references EndUser NOT NULL)

Items(Item.ID: int primary key, brand: varchar(25), name: varchar(50), color_variants: varchar(50))

ItemsSubattributes(Item.ID: int, subattribute: varchar(25), primary key(Item.ID, subattribute), foreign key(Item.ID) references Items NOT NULL)

Posts(Item.ID: int, endUser_login: varchar(15), primary key(Item.ID, user_login), foreign key(Item.ID) references Items, foreign key(endUser_login) references EndUser)

Category(Category.ID: int primary key, Category_Name: varchar(50), Sub_Category.ID: int, foreign key(Sub_Category.ID) references Category)

Approach with Merge Rule → for ISA:

User(login: varchar(15) primary key, email_address: varchar(50), password: varchar(16), user_type: ENUM(Admin, CustomerRep, End-User), isBuyer: boolean, isSeller: boolean, BidAlert: text, foreign key(Item.ID) references Items)

Question(question.ID: int primary key, answer_text: text, question_text: text, user_login: varchar(15), foreign key(user_login) references User NOT NULL)

Items(Item.ID: int primary key, brand: varchar(25), name: varchar(50), Color_Variants: varchar(50))

Posts(Item.ID: int, user_login: varchar(15), primary key(Item.ID, user_login), foreign key(Item.ID) references Items, foreign key(user_login) references Users)

ItemsSubattributes(Item.ID: int, subattribute: varchar(25), primary key(Item.ID, subattribute), foreign key(Item.ID) references Items NOT NULL)

Bid(Bid.ID: int primary key, Bid_Date: date, Bid_time: time, Bid_Amount: double, Auction.ID: int, user_login: varchar(15), foreign key(Auction.ID) references Auction NOT NULL, foreign key(user_login) references User NOT NULL)

Auction(Auction.ID: int primary key, Current_Price: double, Auction_Closing_Date: date, Auction_Closing_time: time, Bid_Increment: double, Initial_Price: double, Minimum: double, Upper_Limit: double, Winner: varchar(50), login: varchar(15), Item.ID: int, foreign key (Item.ID) references Items NOT NULL, foreign key (login) references User NOT NULL)

Category(Category.ID: int primary key, Category_Name: varchar(50), Sub_Category.ID: int, foreign key(Sub_Category.ID) references Category)

*Note:

For subattribute - Attribute, we decided to write it as a weak entity since there might possibly be more than one sub attribute, of which at this current time, we don't know how many. Partial Key: 'subattribute'.

In terms of .SQL schema:

Approach with Collapse Down → for ISA (Rest Merge Rule):

```
CREATE TABLE CustomerRep (  
    CustomerRep_login VARCHAR(15) PRIMARY KEY,  
    email_address VARCHAR(50),  
    password VARCHAR(16)  
);
```

```
CREATE TABLE Admin (  
    admin_login VARCHAR(15) PRIMARY KEY,  
    email_address VARCHAR(50),  
    password VARCHAR(16),  
    CustomerRep_login VARCHAR(15) NOT NULL,  
    FOREIGN KEY (CustomerRep_login) REFERENCES CustomerRep(CustomerRep_login)  
);
```

```
CREATE TABLE Items (  
    Item_ID INT PRIMARY KEY,  
    brand VARCHAR(25),  
    name VARCHAR(50),  
    color_variants VARCHAR(50)  
);
```

```
CREATE TABLE EndUser (  
    endUser_login VARCHAR(15) PRIMARY KEY,  
    email_address VARCHAR(50),  
    password VARCHAR(16),  
    isBuyer BOOLEAN,  
    isSeller BOOLEAN,  
    Item_ID INT,  
    BidAlert TEXT,  
    FOREIGN KEY (Item_ID) REFERENCES Items(Item_ID)  
);
```

```
CREATE TABLE Question (  
    question_ID INT PRIMARY KEY,  
    answer_text TEXT,  
    question_text TEXT,  
    CustomerRep_login VARCHAR(15),
```

```
endUser_login VARCHAR(15),  
FOREIGN KEY (CustomerRep_login) REFERENCES CustomerRep(CustomerRep_login) NOT NULL,  
FOREIGN KEY (endUser_login) REFERENCES EndUser(endUser_login) NOT NULL  
);
```

```
CREATE TABLE Auction (  
    Auction_ID INT PRIMARY KEY,  
    Current_Price DOUBLE,  
    Auction_Closing_Date DATE,  
    Auction_Closing_time TIME,  
    Bid_Increment DOUBLE,  
    Initial_Price DOUBLE,  
    Minimum DOUBLE,  
    Upper_Limit DOUBLE,  
    Winner VARCHAR(50),  
    endUser_login VARCHAR(15),  
    Item_ID INT,  
    FOREIGN KEY (Item_ID) REFERENCES Items(Item_ID) NOT NULL,  
    FOREIGN KEY (endUser_login) REFERENCES EndUser(endUser_login) NOT NULL  
);
```

```
CREATE TABLE Bid (  
    Bid_ID INT PRIMARY KEY,  
    Bid_Date DATE,  
    Bid_time TIME,  
    Bid_Amount DOUBLE,  
    Auction_ID INT,  
    endUser_login VARCHAR(15),  
    FOREIGN KEY (Auction_ID) REFERENCES Auction(Auction_ID) NOT NULL,  
    FOREIGN KEY (endUser_login) REFERENCES EndUser(endUser_login) NOT NULL  
);
```

```
CREATE TABLE ItemsSubattributes (  
    Item_ID INT,  
    subattribute VARCHAR(25),  
    PRIMARY KEY (Item_ID, subattribute),  
    FOREIGN KEY (Item_ID) REFERENCES Items(Item_ID) NOT NULL  
);
```

```
CREATE TABLE Posts (  
    Item_ID INT,  
    endUser_login VARCHAR(15),
```

```

PRIMARY KEY (Item_ID, endUser_login),
FOREIGN KEY (Item_ID) REFERENCES Items(Item_ID),
FOREIGN KEY (endUser_login) REFERENCES EndUser(endUser_login)
);

```

```

CREATE TABLE Category (
  Category_ID INT PRIMARY KEY,
  Category_Name VARCHAR(50),
  Sub_Category_ID INT,
  FOREIGN KEY (Sub_Category_ID) REFERENCES Category(Category_ID)
);

```

Approach with Merge Rule → for ISA:

```

CREATE TABLE User (
  login VARCHAR(15) PRIMARY KEY,
  email_address VARCHAR(50),
  password VARCHAR(16),
  user_type ENUM('Admin', 'CustomerRep', 'End-User'),
  isBuyer BOOLEAN,
  isSeller BOOLEAN,
  BidAlert TEXT,
  -- Including a reference to Items as per your original design, though it's unconventional
  Item_ID INT,
  FOREIGN KEY (Item_ID) REFERENCES Items(Item_ID)
);

```

```

CREATE TABLE Items (
  Item_ID INT PRIMARY KEY,
  brand VARCHAR(25),
  name VARCHAR(50),
  Color_Variants VARCHAR(50)
);

```

```

CREATE TABLE Question (
  question_ID INT PRIMARY KEY,
  answer_text TEXT,
  question_text TEXT,
  user_login VARCHAR(15),

```

```
FOREIGN KEY (user_login) REFERENCES User(login) NOT NULL  
);
```

```
CREATE TABLE Posts (  
    Item_ID INT,  
    user_login VARCHAR(15),  
    PRIMARY KEY (Item_ID, user_login),  
    FOREIGN KEY (Item_ID) REFERENCES Items(Item_ID),  
    FOREIGN KEY (user_login) REFERENCES User(login)  
);
```

```
CREATE TABLE ItemsSubattributes (  
    Item_ID INT,  
    subattribute VARCHAR(25),  
    PRIMARY KEY (Item_ID, subattribute),  
    FOREIGN KEY (Item_ID) REFERENCES Items(Item_ID) NOT NULL  
);
```

```
CREATE TABLE Bid (  
    Bid_ID INT PRIMARY KEY,  
    Bid_Date DATE,  
    Bid_time TIME,  
    Bid_Amount DOUBLE,  
    Auction_ID INT,  
    user_login VARCHAR(15),  
    FOREIGN KEY (Auction_ID) REFERENCES Auction(Auction_ID) NOT NULL,  
    FOREIGN KEY (user_login) REFERENCES User(login) NOT NULL  
);
```

```
CREATE TABLE Auction (  
    Auction_ID INT PRIMARY KEY,  
    Current_Price DOUBLE,  
    Auction_Closing_Date DATE,  
    Auction_Closing_time TIME,  
    Bid_Increment DOUBLE,  
    Initial_Price DOUBLE,  
    Minimum DOUBLE,  
    Upper_Limit DOUBLE,  
    Winner VARCHAR(50),  
    login VARCHAR(15),  
    Item_ID INT,  
    FOREIGN KEY (Item_ID) REFERENCES Items(Item_ID) NOT NULL,
```

```
FOREIGN KEY (login) REFERENCES User(login) NOT NULL  
);
```

```
CREATE TABLE Category (  
    Category_ID INT PRIMARY KEY,  
    Category_Name VARCHAR(50),  
    Sub_Category_ID INT,  
    FOREIGN KEY (Sub_Category_ID) REFERENCES Category(Category_ID)  
);
```