

**Users**  
Stores login information of each of the users using the application

**Posts**  
Users will post what they want to sell and indicate what they'd be willing to trade for (can be multiple types of items)

**BarterOptions**  
Associates posts with items, and number of items they'd be willing to trade for

| Users     |                |
|-----------|----------------|
| UserID    | Primary Key    |
| Gamertag  |                |
| FirstName |                |
| LastName  |                |
| Password  | Will be hashed |
| Email     |                |

| Posts    |             |
|----------|-------------|
| PostID   | Primary Key |
| UserID   | Foreign Key |
| ItemID   | Foreign Key |
| PostTime | TimeStamp   |
| GoldCost | Nullable    |
| PostText |             |
| Active   | Boolean     |

| BarterOptions |             |
|---------------|-------------|
| PostID        | Foreign Key |
| ItemID        | Foreign Key |
| askingNum     | Default 1   |
| GroupNum      |             |

| ItemType |             |
|----------|-------------|
| TypeID   | Primary Key |
| TypeName |             |

| Items    |             |
|----------|-------------|
| ItemID   | Primary Key |
| ItemName |             |
| TypeID   | Foreign Key |

**Items**  
-These are items that can be bought and sold in the game.  
-Each row represents a class of item, not an instance.  
-Table prevents raw string data in BarterOptions to represent different items.

**ItemType**  
-Each entry in this table is an allowed type that an item can be. (i.e. sword, shield, potion etc).  
-Items will have an item type.  
-This separates the data to ensure items have proper types