

AuthToken(AuthToken, alias, timestamp)

- Foreign key alias references Users
- Named AuthToken because it stores authentication tokens
- Stores authentication tokens associated with a user
- This table relates to the User table, because each auth token stored is associated with a single user from the user table
- AuthToken - a unique identifier that serves as authentication
- Alias - the primary key from the user table
- Timestamp - Timestamp saves the time when the auth token was given
- Table is normalized because alias and timestamp are dependent on the whole AuthToken, which is the primary key

Feed(feedID, alias, postID)

- Foreign key alias references Users
- Foreign key postID references Posts
- This table relates to the post table and references user with alias as a foreign key
- This table stores an alias associated with a post ID. All of the posts associated with a single alias makeup that user's feed.
- feedID - unique id that serves as a primary key
- Alias - a reference to the user who the feed belongs to
- postID - a reference to a post in the Post table
- This table is normalized because it doesn't store an alias with a string of comma separated postIDs

Followers(followersID, userAlias, followerAlias)

- Foreign key userAlias references Users
- Foreign key followerAlias references Users
- Adds/removes followers
- FollowersID - a primary key that relates to follower
- UserAlias - the primary key from the user table
- FollowerAlias - a reference to another user
- This table is normalized because it doesn't store an alias with a string of comma separated followerAliases. It is also separated out of the users table

Following(followingID, userAlias, followingAlias)

- Foreign key userAlias references Users
- Foreign key followingAlias references Users
- Stores all following actions of users
- FollowingID - unique identifier for action
- userAlias - alias of user that is doing the following
- followingAlias - alias of user being followed
- This table is normalized because it doesn't store an alias with a string of comma separated followingAliases. It is also separated out of the users table

Hashtag(hashtagID, hashtag, postID)

- Foreign key postID references Posts
- Stores references to a specific hashtag
- hashtagID - unique value corresponding to hashtag
- Hashtag - a hashtag
- postID - postID of Post that has hashtag
- This table is normalized because it doesn't store a hashtag with a string of comma separated postIDs.

Posts(postID, alias, content, timestamp, image, video)

- Foreign key alias references Users
- Stores user posts
- Alias - alias corresponding to user
- Content - content stored in post
- Timestamp - timestamp when post was created
- Image - image (if any) in post
- Video - video (if any) in post
- Table is normalized because all of the columns are dependent on the whole postID, which is the primary key

Users(alias, name, photo)

- Stores user profile data
- Name - name of user
- Photo - photo of user
- Table is normalized because all of the columns are dependent on the userAlias and it doesn't store a name and photo with a string of comma separated by userAlias.