1) Draw UML/ER Diagram (as PDF file)

- Completed by Tina

2) Assumptions

- USER
  - Each user has a user_id (primary key), username and password for authentication
  - Has a one to many relationship to USER_FAVORITE as one user can have multiple favorite teams/players
- USER_FAVORITE
  - USER_FAVORITE is an entity instead of an attribute of USER because one user can have multiple different favorite teams/players
  - The attribute favorite_type distinguishes if favorite_id is a player_id or a team_id
  - There is a many-to-one relationship between USER_FAVORITE to TEAM/PLAYER because a player/team can be a favorite of many users
- TEAM
  - Each team has a unique team_id with attributes team_name, city and arena
  - Shares a one-to-many relationship with PLAYER and GAME
    - One team has many players and plays many games
  - Shares a one-to-one relationship with TEAM_SENTIMENT
    - Each TEAM can only have one SENTIMENT
- PLAYER
  - Each player has a player_id(primary key) , player_name, position and team_id(foreign key)
    - Team_id is a foreign key because each player belongs to a single team at a time
  - PLAYER is an entity instead of an attribute in TEAM because each player can have a different performance, and the user can select a favorite player and/or a favorite team
  -
- GAME
  - Each game has an unique event with start time, date, venue and final scores for the visitor teams. GAME is an entity because one game has 2 teams and many players. The details, including scores and overtime_status, obtained from GAME cannot be attributes of other entities.

    - Attributes include:
        - game_id: Unique identifier for each game
        - game_date: Indicate the date the game takes place
        - start_time: Indicate when the game starts
        - home_team_id: FK for the home team
        - visitor_team_id: FK for the visitor team
        - arena: Indicate where the game is played
        - home_score: Final score for the home team
        - visitor_score: Final score for the visitor team
        - overtime: Boolean flag to indicate if the game went into overtime
    - Relationships:
        - GAME has a dual one-to-many relationship with TEAM as teams are connected as visitor and home via separate FKs
        - GAME has many-to-many relationships with PLAYER through PERFORMANCE as GAME feature multiple players with individual stats.
        - GAME has one-to-one relationship with PREDICTION as each game possess one unique prediction record

- PERFORMANCE
    - Each performance record tracks a player's statistics in a specific game, with a composite primary key made up of game_id and player_id
    - Attributes include:
        - Points
        - Assists
        - Rebounds
        - minutes played,
        - Three_point_percentage
    - To capture individual game performance
    - Shares a many-to-one relationship with PLAYER because one PLAYER and have multiple PERFORMANCE's
    - Shares a many-to-one relationship with GAME because each performance belongs to one game
    - This entity exists to track detailed game-level performance over time, rather than storing performance as attributes on PLAYER

- PREDICTION
    - Each prediction is a forecast for a specific basketball game, with prediction_id being the primary key
    - Attributes include:
        - predicted _home_score
        - Predicted_visitor_score
        - Win_probability
        - Prediction_date
        - Prediction_score
    - To track both forecasted results and when the prediction was made
    - Shares a one-to-one relationship with GAME because each game can only have one prediction (based on our current design), though we could extend the model in the future to support multiple predictions per game
    - This entity exists separately to preserve a record of predictions independently from the games actual results, enabling comparison and model evaluation based on how accurate predictions were
- TEAM_SENTIMENT
    - Each sentiment record tracks public or analyst sentiment for a specific team on a specific date, with sentiment_id being the primary key
    - Attributes include:
        - Sentiment_score
        - Sentiment_date
    - These attributes representing the sentiment level and the date that it was measured
    - Shares a many-to-one relationship with TEAM because each team can have multiple sentiment records over the course of a season
    - Shares a many-to-one relationship with DATA_SOURCE because each sentiment record comes from one specific source
    - This entity exists to track how sentiment changes over time from different sources, rather than a static attribute on TEAM alone
- DATA_SOURCE
    - Each data source has a unique_score_id as the primary key
    - Attributes include:
        - Source_name
        - Source_url

        - Source_type
    - To describe the source and classify it (i.e. if it came from social media or the news)
    - Shares a one-to-many relationship with TEAM_SENTIMENT because a single source can provide sentiment data for multiple teams
    - This entity exists to maintain a central reference for all data sources, ensuring consistency and traceability across the system

4) Normalize Database (BCNF or 3NF)

- **BCNF**

5) Convert conceptual database design to logical design (relational schema)

USER(user_id: INT[PK], username: VARCHAR(50), email:VARCHAR(100), password: VARCHAR(100))

- This table is already in BCNF

USER_FAVORITE(user_id: INT[PK, FK to USER.user_id], favorite_id: INT[PK], favorite_type: VARCHAR(20))

- This is also in BCNF

TEAM(team_id: INT[PK], team_name: VARCHAR(50), city: VARCHAR(50), arena: VARCHAR(50))

- This table is in BCNF since the city, arena, and team_name depend on the team_id.

PLAYER(player_id: INT[PK], player_name: VARCHAR(100), position: VARCHAR{10), team_id: INT[FK TEAM.team_id],)

- This is in BCNF as all the player stats depend on the player id.

GAME(game_id: INT[PK], game_date: DATE, start_time: TIME. home_team_id: INT [FK to TEAM.team_id], points: INT, assists: int, rebounds: INT, minutes_played: INT)

- Provided that the stats are aggregated into a single number, this table is also in BCNF.

PERFORMANCE(game_id: INT[PK, FK to GAME.game_id], player_id: INT[PK, FK to PLAYER.player_id], points: INT, assists: INT, rebounds: INT, minutes_played: INT, three_point_percentage: DECIMAL)

- Each player will have a unique statline each game, which means this table is also in BCNF.

PREDICTION(prediciton_id: INT[PK], game_id: INT[FK to GAME.game_id], predicted_home_score: DECIMAL, predicted_visitor_score: DECIMAL, win_probability: DECIMAL, prediction_date: DATE)

- If the game id is different for each game, even if it is the same two teams playing again, this table is also in BCNF.

TEAM_SENTIMENT(sentiment_id: INT[PK], team_id: INT[FK to TEAM.team_id], sentiment_score: DECIMAL, sentiment_date: DATE, source_id: INT [FK to DATA_SOURCE.source_id])

- This table is not in BCNF, so it can be altered as follows.

TEAM_SENTIMENT(team_id: INT[PK, FK to TEAM.team_id], sentiment_date: DATE[PK], source_id: INT[FK to DATA_SOURCE.source_id], sentiment_score: DECIMAL)

- Since the team id, date and source id are all used to calculate the sentiment score, using them as composite keys preserves the BCNF balance.

DATA_SOURCE(source_id: INT[PK], source_name: VARCHAR(50), source_url: VARCHAR(100), source_type: VARCHAR(20))

- This table is in BCNF