Instructions

0. Compile all classes. E.g. javac Server.class (Already compiled in BlackJack.zip)
1. Run the Server class: java Server
    a. input port number: 6789
2. Run the Client class: java Client
    a. input IP address: localhost
    b. input port number: 6789
3. Repeat step 2 for multiple players
4. Instructions for detailed game play (i.e., start or join a game, type game name and user name, make bet, choose hit or stay) are included in the console-based UI.

Design choices

1. Class structure:
    a. Server:
        The Server class contains a list of ongoing Games in this server, and a list of ServerThreads representing all users in this server. It manages the status of each Game, and notifies ServerThreads and Dealers accordingly.
    b. Client:
        The Client class contains all the information of individual clients and displays player-interaction-needed UI. Every time a player makes a decision, it will send a Message to ServerThread, which will notify Server. At certain points, it also receives Message from ServerThread, which is notified by Server.
    c. ServerThread:
        The ServerThread class is a medium for Client and Server to communicate. Each ServerThread is a representative of a Client and communicate with other classes in the system. It continuously receives Message from Client, analyze the Message type and notify the Server to do corresponding actions.
    d. Message:
        The Message class is a communication tool between Client and ServerThread. It has many getters and setters for information needed during the communication.
    e. Dealer:
        The Dealer class contains all the information of a dealer. It is similar to the ServerThread class and communicates with Server directly.
    f. Game:
        The Game class contains information of individual games and manages UI for each player's status
    g. Card:
        The Card class contains information of individual cards in the game
2. Multi-threading: The program uses multi-threading so that players in the same game can have simultaneous notifications during the progression of the game. Clients and Server are related through Socket. ServerThread and Client communicate through ObjectInputStream and ObjectOutputStream.