Port 6666

Class

There are totally 13 classes in the project which includes Card, CardHand, CardPack, Client, Deck, Face, GamePanel, gameTable, Player, PlayerCardHand, Server, Suit and Util.

Suit is a class to represent the suit of the playing card.

Face is a class to represent the value of the playing card. Particularly, Ace contains two values. In addition, the value of JQK is 10.

Card is a class to represent a playing card which combines class Suit with class Face. All playing cards need to be initialized by class Card.

Deck is a class which extends Stack<Card>. Deck represents one/n shuffled deck(s) of playing cards. Particularly, we can Initialize the deck by passing a variable through the constructor function. 1 means one pack of cards, while n means n packs of cards. Afterwards, we add the CardPack object to the end of the stack in the constructor. CardPack implements the import of card. Meanwhile, CardPack imports 0-52 cards by traversing suits and faces.

CardHand is a class extends Vector.

CardHand.getTotal(): Get the value of the playing card.

CardHand.isBust(): Judge if the player has a burst point.

CardHand.clear(): Remove all cards.

CardHhand.hasBlackjack(): Judge if the player has a blackjack.

Specifically, In CardHand.getTotal(), the last card is hidden because other clients are considered because of the masking effect. Therefore, when we are processing on the client side, we will set the Name of the suit of the last card of other players to hidden so that when we calculate the total points, we can automatically remove the last card points.

PlayerCardHand extends CardHand. PlayerCardHand is used to determine

whether playing card is burst or blackjack when the player adds another card.

Playercardhand.add (Card) adds a card to the hand, requiring a Card object to be passed.

Protocol

Due to the problem that the client could not receive when transferring the serialized data of the object, the problem could not be solved after a period of trying, so the data was transmitted in a specified data format:

Notification: System:notice# notification message

Current round ends: system:result# game results

Notification of new player launch: system:newplayer# notification message

Display all players: allplayer#1,2,3,4 (or 1,2/1,2,3)

The device is bound to the player: system:bindplayer#1(2/3/4)

Bet: system:bet#bet

Gets and sets gold: system:money# amount

Gets the gold String: system:changeMoney#amount

Turn the current player to choose: system:showchoose

Get a player's card:

system:getPlayerCard#1#Ten,Clubs,36@Five,Diamonds,18#15(split card with “@”, then split card with”,” by split face, suit, card number)

Server

server handles the rules and process of game.

Firstly, it is necessary to determine the number of players. Then, server starts to receive client connections until the client connection satisfies conditions. Afterwards, it calls Deck to shuffle cards, and Server will traverse all clients and make the bet. Furthermore, it will traverse all clients to deal cards to clients. Thirdly, it will make operations according to acquiring choose by traversing clients. At last, Server will determine whether if the end of game rule is met and it will settle gold. Then, it will wait the input for new number of players.

Client

Client is responsible for server-side message parsing and interface building.

After the top rendering, call socket long connection to achieve the interface operation and message parsing.

Open the window through AppWindow and wrap the GamePanel.

The GamePanel implements the main interface and controls.

GameTable draws the layout of the game cards to achieve a top to bottom level operation.

GameTable inherits jPanel and implements the repaint method by overriding the jPanel and paintComponent(Graphics g) method to complete the redrawing requirement after each player's card acquisition.

Execution

In addition, shortcuts are set:

Hit - ctrl + C

Double - ctrl + D

Stand - ctrl + F

$1 – ctrl + 1

$5 - ctrl + 2

$10 - ctrl + 3

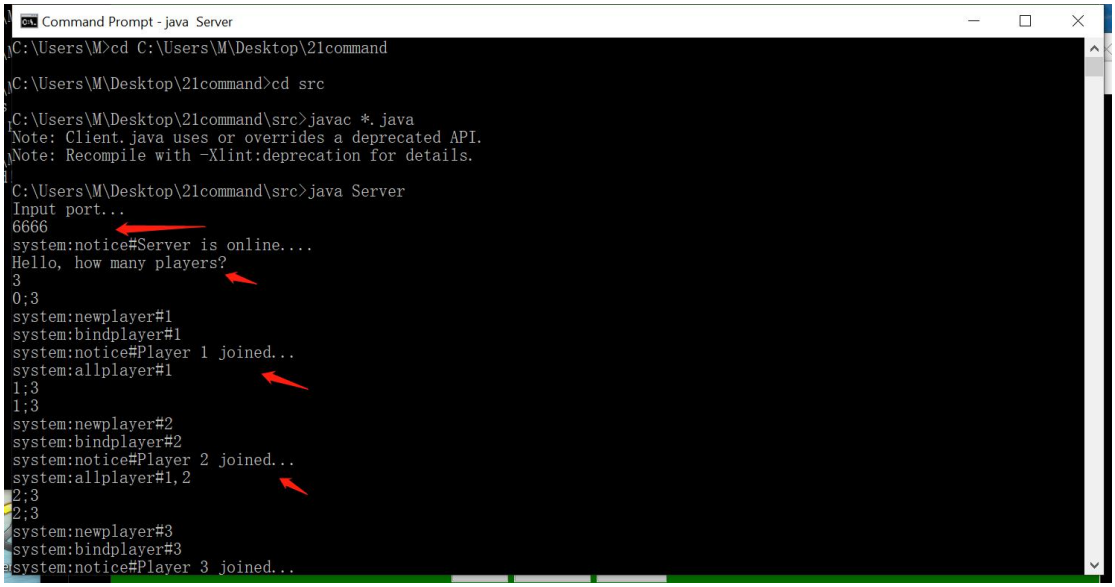
$25 - ctrl + 4

$100 - ctrl + 5

Preparation:

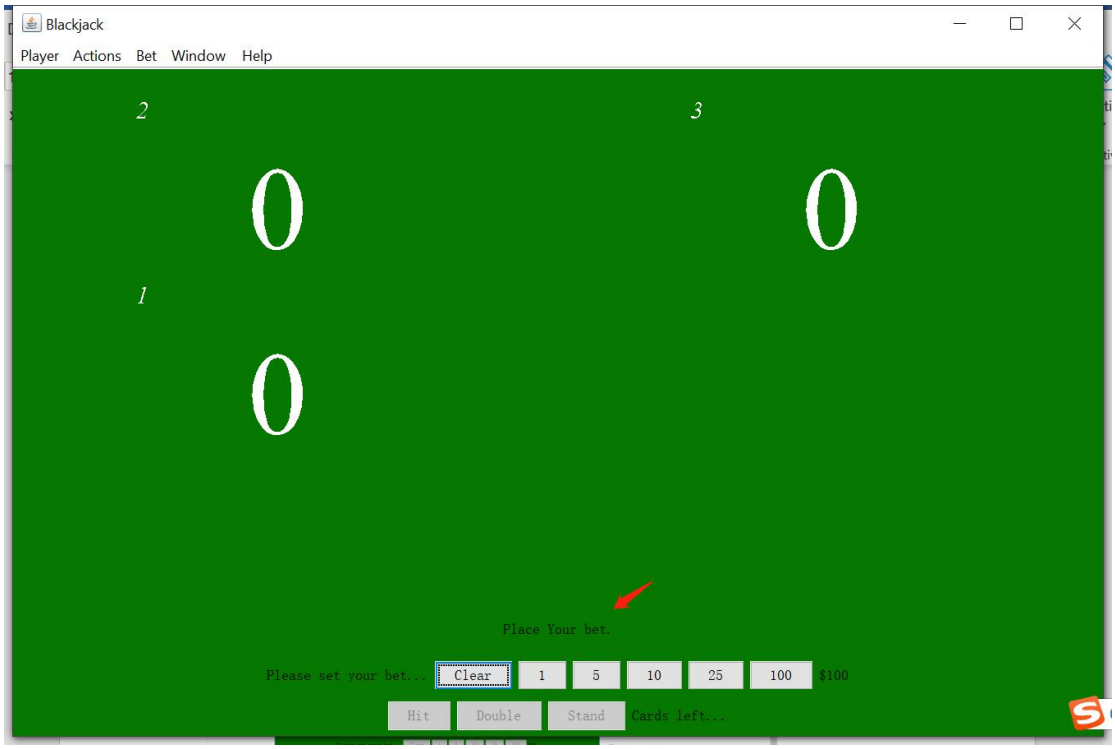
Port – 6666

It is allowed 2 – 4 players to join the game



Preparation (Not bet):

After adding players into the game, we need to make a bet to start a new round.

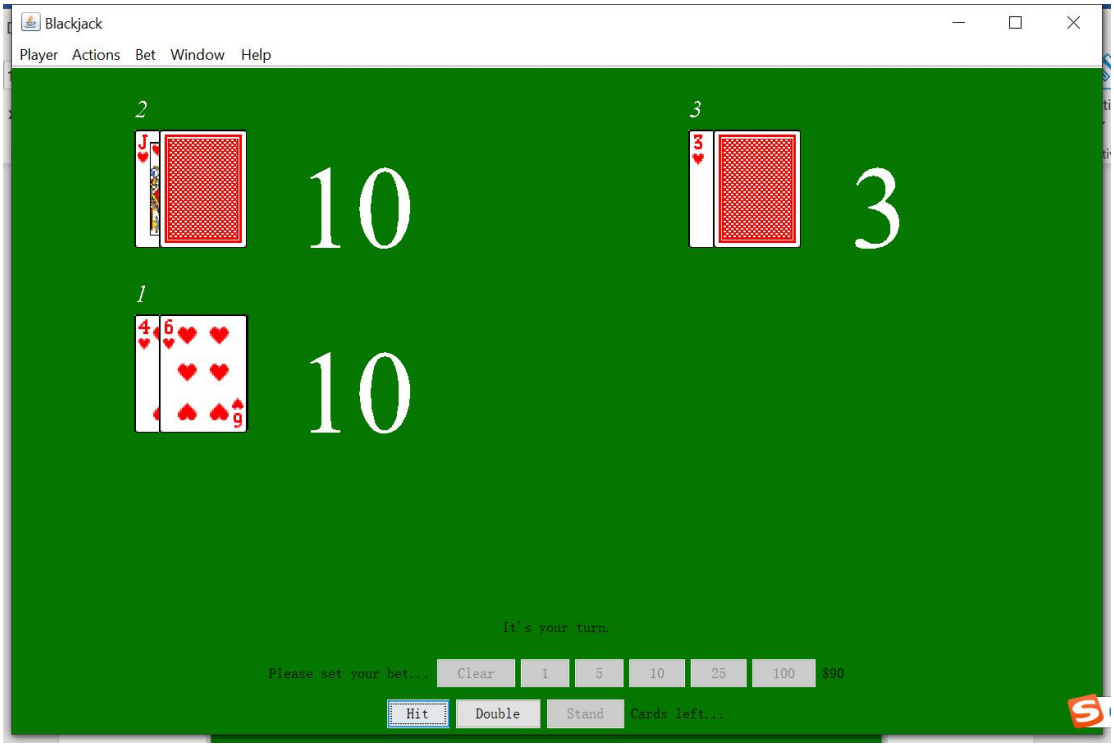


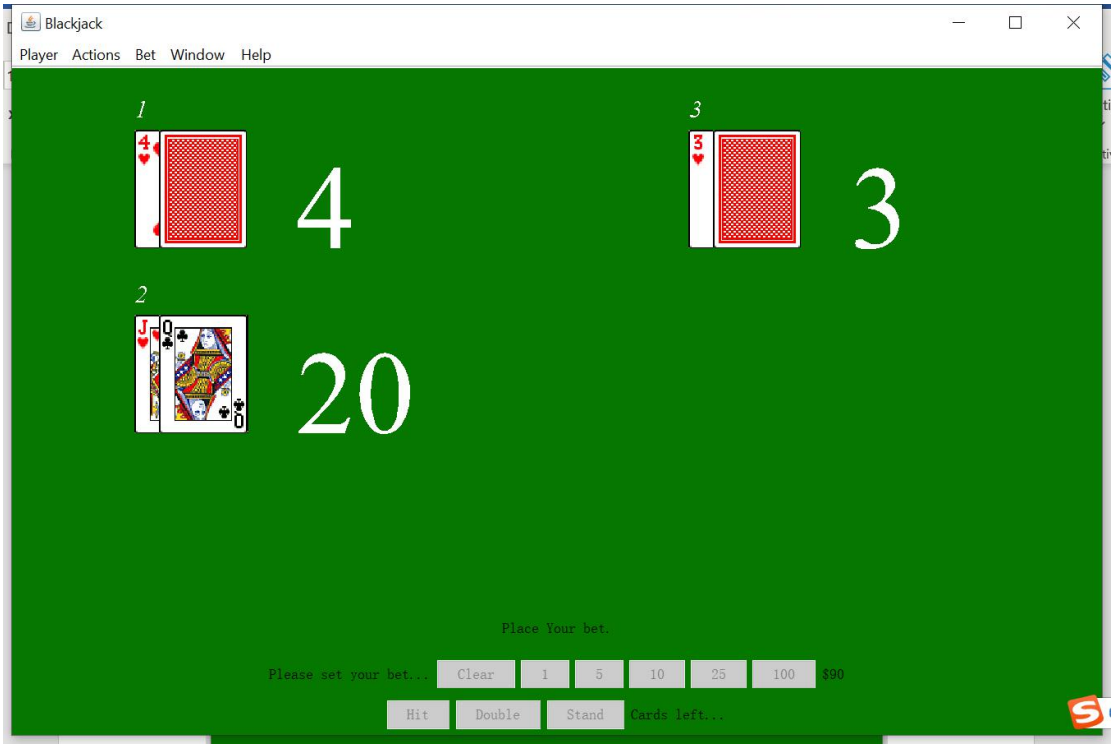
Game start (bet):

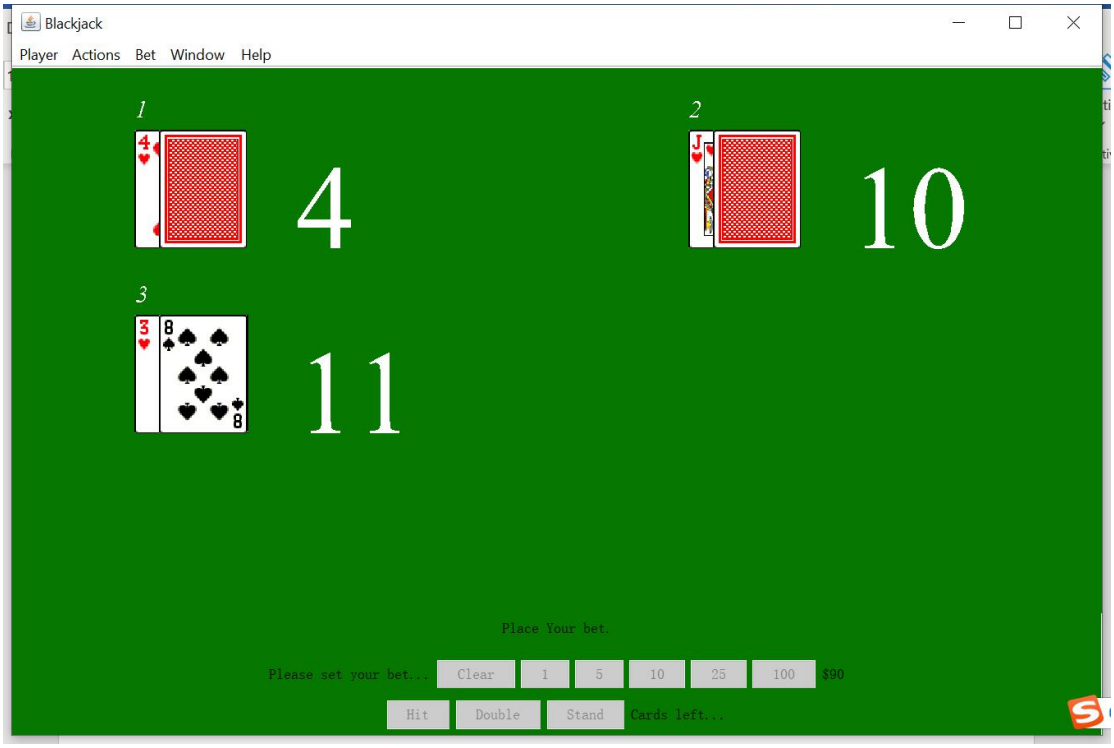
The next three pictures show the hand card of three players

Each operation should be performed sequentially though it is a Stand

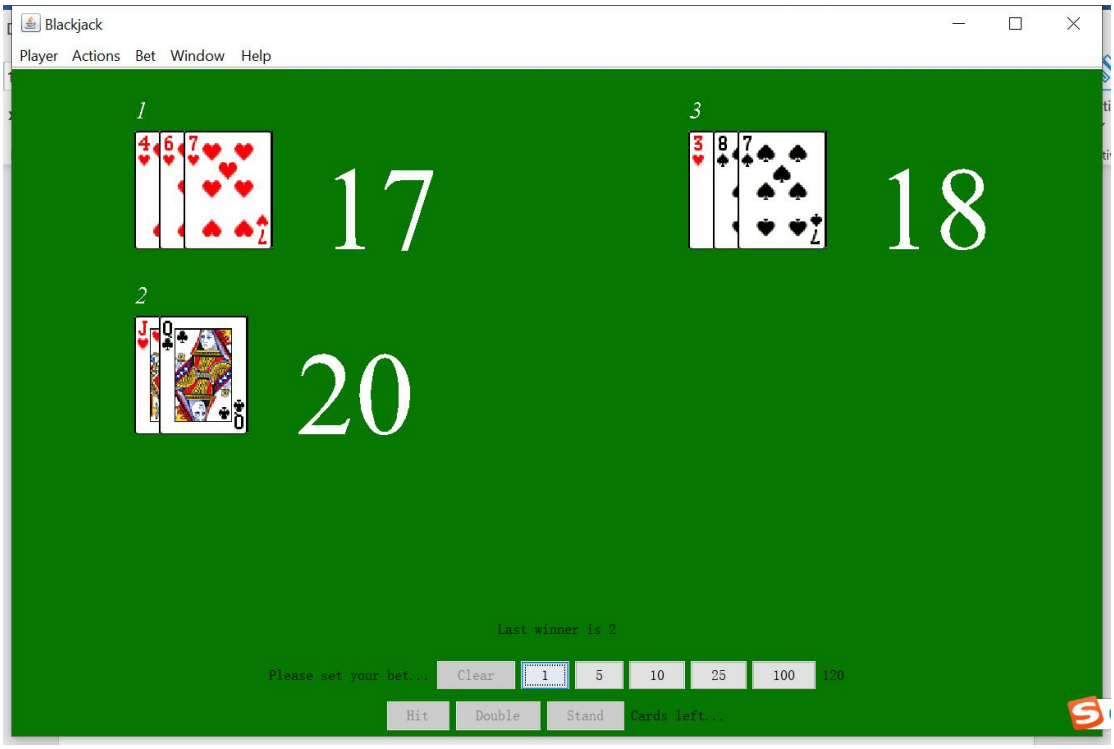
If the initial two hand card is below 15, you have to hit first







After selecting hit or stand, the final winner is player 2.



In addition, you can remove or add player as you want