Java Go Fish
Diagrams

10, 000 Foot View:

Project Structure:

JavaGoFish
- Source Packages
  - ca.sheridancollege.javagofish.Cards
  - ca.sheridancollege.javagofish.InputRules
  - ca.sheridancollege.javagofish.Players
  - ca.sheridancollege.javagofish.Start
  - ca.sheridancollege.javagofish.Turns
  - ca.sheridancollege.javagofish.UnitTests
  - ca.sheridancollege.javagofish.Utility
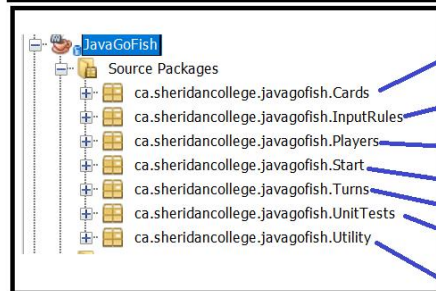
Classes for cards, dealer, and deck.
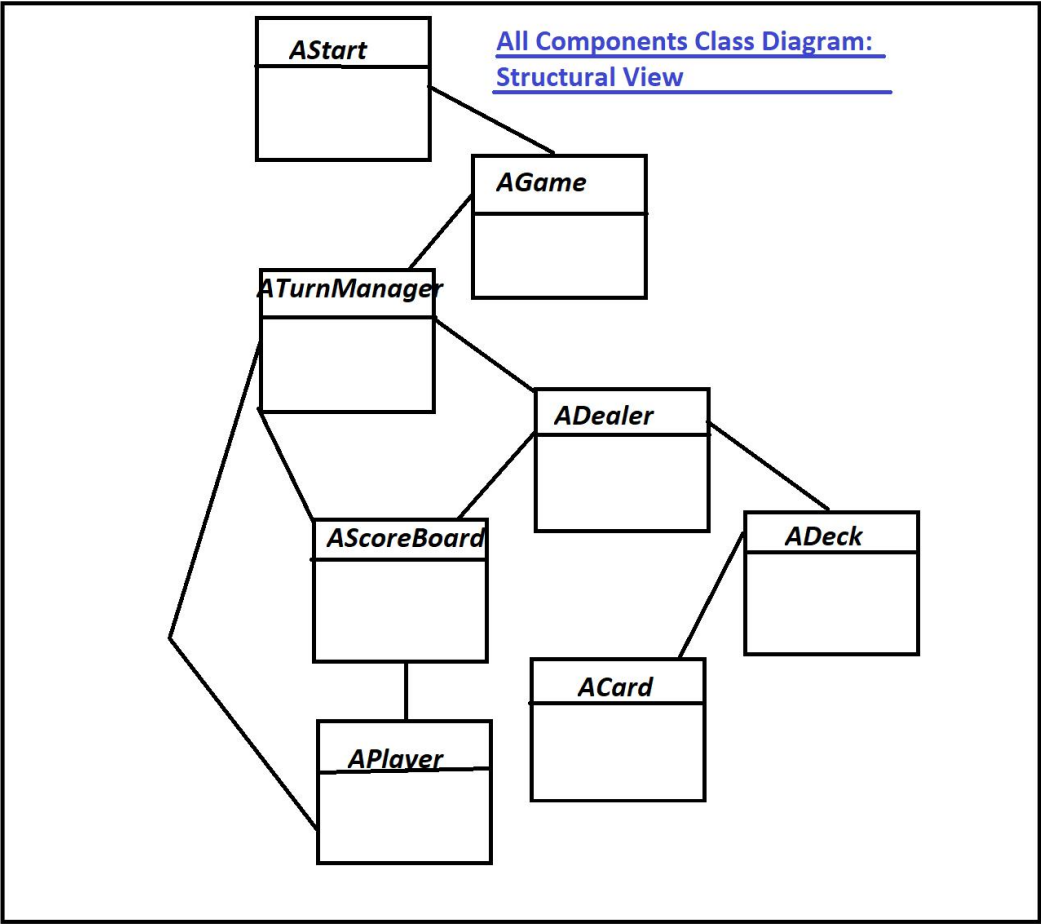
A class for input rules.

Classes for player types.

Classes for games, starting, and rounds,

Classes for keeping score, and managing turns.

Classes modelling coponents in isolation.

Printing & Input classes.

All Components Class Diagram: Structural View

AStart

AGame

ATurnManager

ADealer

AScoreBoard

ADeck

ACard

APlayer

User Interface Description

```
Creating Hands:
```
Feedback for human player when action occurs.

```
Creating Duplicate Lists:
```
Feedback for action occuring.

```
Setting who goes first
```
Feedback for action occuring.

```
Round One

[Aleks] -In Play Hand:
```
Human players hand as console output.
```
ACE of SPADES
ACE of DIAMONDS
EIGHT of SPADES
FOUR of SPADES
FIVE of HEARTS
QUEEN of HEARTS
TWO of DIAMONDS

[Aleks] -In Play Desirable List:
```
Human players duplicate list.
```
ACE of SPADES
ACE of DIAMONDS

[Computer] -Not In Play Hand:
```
Computer hand so prof can test and see changes.
```
EIGHT of CLUBS
KING of HEARTS
QUEEN of CLUBS
QUEEN of SPADES
SIX of HEARTS
SIX of SPADES
THREE of CLUBS
```

```
[Computer] -Not In Play Desirable List:
```
Computer duplicate list. For professor to test my code and see changes.
```
QUEEN of SPADES
QUEEN of CLUBS
SIX of HEARTS
SIX of SPADES

Which card value do you want?
```
Human asks for a card of the computer here.
```
· Enter a String:
```
Can type now on keyboard

```
Which card value do you want?
Enter a String:
eight
```
I typed eight because I know computer has it.

```
Computer does have EIGHT
```
Feedback for human user to know if ask was a success.

```
Calculating dupes for Aleks
```
Duplicates are recalculated for Human because they get the card.

```
Aleks dupes:
ACE of SPADES
ACE of DIAMONDS
EIGHT of SPADES
EIGHT of CLUBS
```
Human duplicates are printed out for Human to see.

```
[Aleks] In Play Hand:
ACE of SPADES
ACE of DIAMONDS
EIGHT of SPADES
EIGHT of CLUBS
FIVE of HEARTS
FOUR of SPADES
QUEEN of HEARTS
TWO of DIAMONDS
```
Humans updated hand is also printed out for clarity.

```
[Computer] Not In Play Hand:
KING of HEARTS
QUEEN of CLUBS
QUEEN of SPADES
SIX of HEARTS
SIX of SPADES
THREE of CLUBS
```
Computer updated hand is printed out for prof to see my code working.

```
Which card value do you want?
Enter a String:
```
Game waits for Human to ask for a card again because there ask was a success.

```
Which card value do you want?
Enter a String:
ace
Computer does not have ACE


Calculating dupes for Aleks

Aleks dupes: [ACE of SPADES, ACE of DIAMONDS, EIGHT of SPADES, EIGHT of CLUBS]

Aleks In Play Hand:
ACE of SPADES
ACE of DIAMONDS
EIGHT of SPADES
EIGHT of CLUBS
FIVE of HEARTS
FOUR of SPADES
JACK of HEARTS
QUEEN of HEARTS
TWO of DIAMONDS


Computer Not In Play Hand:
KING of HEARTS
QUEEN of CLUBS
QUEEN of SPADES
SIX of HEARTS
SIX of SPADES
THREE of CLUBS
```

I asked for a card I know computer doesn't have to show how turns are switched. (annotation, pointing to `ace`)

Feedback to let Human know there ask was unsuccessful. (annotation for `Computer does not have ACE`)

Recalculate duplicates for Human and notify action is occuring. (annotation for `Calculating dupes for Aleks`)

Print Human duplicates. (annotation for `Aleks dupes: [...]`)

Print Humans hand as latest current. (annotation for `Aleks In Play Hand:`)

Print computers hand as latest current. (annotation for `Computer Not In Play Hand:`)

```
Switching who's in play

[Computer] In Play

[Aleks] Not In Play

Calculating books for Aleks

[Computer] -In Play Hand:
KING of HEARTS
QUEEN of CLUBS
QUEEN of SPADES
SIX of HEARTS
SIX of SPADES
THREE of CLUBS


[Computer] -In Play Desirable List:
QUEEN of CLUBS
SIX of HEARTS
SIX of SPADES


[Aleks] -Not In Play Hand:
ACE of SPADES
ACE of DIAMONDS
EIGHT of SPADES
EIGHT of CLUBS
FIVE of HEARTS
FOUR of SPADES
JACK of HEARTS
QUEEN of HEARTS
TWO of DIAMONDS
```

Swtiching turn and notifying action is occuring. (annotation for `Switching who's in play`)

Notifying who's currently the player who's turn it is. (annotation for `[Computer] In Play`)

Notifying the new player who's not in play currently. (annotation for `[Aleks] Not In Play`)

Notifying that we're checking fo four of a kinds for player turn that ended. (annotation for `Calculating books for Aleks`)

Printing computers current hand so user doesn't have to scroll. (annotation)

Showing computers duplicates fo prof to verify correct code. (annotation)

Printing humans current hand so they don't have to scroll up. (annotation)

```
[Aleks] -Not In Play Desirable List:
ACE of SPADES
ACE of DIAMONDS
EIGHT of SPADES
EIGHT of CLUBS


Computer is asking for: QUEEN
Do you have any?
Enter a String:
```

Show humans hand to prevent scrolling.

Freeze program with a user prompt, so Human can follow what's happening without having to scroll back up.

```
Computer is asking for: QUEEN
Do you have any?
Enter a String:
yes
Ok thanks for confirming

Aleks does have QUEEN

Calculating dupes for Computer

Computer dupes:
QUEEN of CLUBS
QUEEN of SPADES
QUEEN of HEARTS
SIX of SPADES
SIX of HEARTS

[Computer] In Play Hand:
KING of HEARTS
QUEEN of CLUBS
QUEEN of SPADES
QUEEN of HEARTS
SIX of SPADES
SIX of HEARTS
THREE of CLUBS
```

Your response doesn't change aanything, it's just to freeze section of program so Human doesn't have to scroll up.

Feedback to give user some interaction.

Let Human know if the ask was successful.

Printing updated duplicates for computer because he got a card.

Printing the duplicates so prof. can verify correct code.

Print computers current hand as latest for prof. verification.

```
Computer is asking for: QUEEN
Do you have any?
Enter a String:
no
Ok, computer will go fish then
Aleks does not have QUEEN

Calculating dupes for Computer

Computer dupes: [QUEEN of CLUBS, QUEEN of SPADES, QUEEN of HEARTS, SIX of SPADES, SIX of HEARTS, THREE of CLUBS, THREE of SPADES]
Computer In Play Hand:
KING of HEARTS
QUEEN of CLUBS
QUEEN of SPADES
QUEEN of HEARTS
SIX of SPADES
SIX of HEARTS
THREE of CLUBS
THREE of SPADES

Aleks Not In Play Hand:
ACE of SPADES
ACE of DIAMONDS
EIGHT of SPADES
EIGHT of CLUBS
FIVE of HEARTS
FOUR of SPADES
JACK of HEARTS
```

Human doesn't have so I said no. response has no weight it's just for scroll prevention.

Human get's notified that the ask was not successful. Which they already know.

Print current duplicates for computer is action occuring.

Show current duplicates of computer for prof. code verification.

Print current computer hand for prof. code verification.

Print humans current hand.

```
Switching who's in play
```
Let human know players whos in turn is being switched.

```
[Aleks] In Play
```
Notify who's the current player in play.

```
[Computer] Not In Play
```
Notify players whos not in play.

```
Calculating books for Computer
```
Notify 4 of a kind is calculated for previous in play player.

```
[Aleks] -In Play Hand:
ACE of SPADES
ACE of DIAMONDS
EIGHT of SPADES
EIGHT of CLUBS
FIVE of HEARTS
FOUR of SPADES
JACK of HEARTS
TWO of DIAMONDS
```
Print humans hand to prevent scrolling annoyance.

```
[Aleks] -In Play Desirable List:
ACE of DIAMONDS
EIGHT of SPADES
EIGHT of CLUBS
```
Print current human hand for same reason.

```
[Computer] -Not In Play Hand:
KING of HEARTS
QUEEN of CLUBS
QUEEN of SPADES
QUEEN of HEARTS
SIX of SPADES
SIX of HEARTS
THREE of CLUBS
```
Print computer's hand so prof. can see updates happening.

```
[Computer] -Not In Play Desirable List:
QUEEN of CLUBS
QUEEN of SPADES
QUEEN of HEARTS
SIX of SPADES
SIX of HEARTS
THREE of CLUBS
THREE of SPADES
```
Print computers hand.

```
Which card value do you want?
Enter a String:
```
Human back to being the asker.

This cycle keeps repeating per cards in the deck.