

Project: Uno but worse...

UML Diagram

Card
<ul style="list-style-type: none"><li>- int _color</li><li>- int _numberOrSpecialty</li><li>- int _secondary</li></ul>
<ul style="list-style-type: none"><li>+ Card(int color, int numberOrSpecialty)</li><li>+ boolean isWild()</li><li>+ String getColor()</li><li>+ String getNumberOrSpecialty()</li><li>+ void setSecondary()</li><li>+ String getSecondary()</li><li>+ String toString()</li></ul>

Deck
<ul style="list-style-type: none"><li>- ArrayList&lt;Card&gt; collection</li><li>- Stack&lt;Card&gt; shuffled</li><li>- Card top</li></ul>
<ul style="list-style-type: none"><li>+ Deck()</li><li>+ void replenish (Stack&lt;Card&gt; asdf)</li><li>+ void swap(int card1, int card2)</li><li>+ void shuffle()</li><li>+ Card draw()</li><li>+ void putCardsDown()</li><li>+ int getSize()</li><li>+ Card removeFromDeck()</li><li>+ String toString()</li></ul>

Table
<ul style="list-style-type: none"><li>- Stack&lt;Card&gt; placed</li><li>- Deck aDeck</li><li>- int direction</li><li>- Player current</li><li>- boolean isStacking</li><li>- int stack_size</li></ul>

<ul style="list-style-type: none"> <li>- final String RESET</li> <li>- final String RED</li> <li>- final String GREEN</li> <li>- final String YELLOW</li> <li>- final String BLUE</li> <li>- final String PURPLE</li> <li>- final String CYAN</li> </ul>
<ul style="list-style-type: none"> <li>+ Table(Player, Player, Player, Player)</li> <li>+ void go()</li> <li>+ void placeCard(int chosen)</li> <li>+ void processCard()</li> <li>+ void addCard(Card used)</li> <li>+ Card placeFirst()</li> <li>+ void drawCurrent(int num)</li> <li>+ void distribute()</li> <li>+ Card getTop()</li> <li>+ Stack&lt;Card&gt; getPlaced()</li> </ul>

Player
<ul style="list-style-type: none"> <li>- Player nextPlayer</li> <li>- Player prevPlayer</li> <li>- boolean isOut</li> <li>- String name</li> <li>- ArrayList&lt;Card&gt; hand</li> </ul>
<ul style="list-style-type: none"> <li>+ Player getNext()</li> <li>+ Player getPrev()</li> <li>+ void setNext(Player a)</li> <li>+ void setPrev(Player a)</li> <li>+ Player NextInLine(int direction)</li> <li>+ boolean validateChoice()</li> <li>+ void placeCard()</li> <li>+ void uno()</li> <li>+ void unoOut()</li> <li>+ void setName()</li> <li>+ void drawCard()</li> <li>+ void wonOrNot()</li> </ul>

Human (extends Player)
<ul style="list-style-type: none"> <li>+ Human(String name)</li> </ul>

<ul style="list-style-type: none"><li>+ boolean isHuman()</li><li>+ void setSecondary(Card wild)</li><li>+ int go(Card top)</li><li>+ int respondToAdding(Card top)</li><li>+ void uno(Deck aDeck)</li></ul>
--

Bot (extends Player)
----------------------

<ul style="list-style-type: none"><li>+ Bot(String name)</li><li>+ boolean isHuman()</li><li>+ void setSecondary(Card wild)</li><li>+ int go(Card top)</li><li>+ int respondToAdding(Card top)</li><li>+ void uno(Deck aDeck)</li></ul>
---

<ul style="list-style-type: none"><li>+ void placeCard() (overridden)</li></ul>
---

Woo
-----

<ul style="list-style-type: none"><li>+ static void startMenu()</li><li>+ static void play()</li></ul>
--