# Assignment 5 - Part 1: Pseudo-Code (C# Style)

## Main Method

do  
{  
 Display menu:  
 1. Setup Game  
 2. Deal Hands  
 3. Display Gameboard  
 4. Quit  
  
 Prompt user for choice  
  
 switch (userChoice)  
 {  
 case 1:  
 Prompt: include jokers? (yes/no)  
 Prompt: suit priority (enter 4 suits in order)  
 Create new GameState(jokersIncluded, suitPriority)  
 break  
  
 case 2:  
 Prompt: number of players  
 Prompt: cards per player  
 gameState.DealHands(numPlayers, cardsPerPlayer)  
 break  
  
 case 3:  
 Console.WriteLine(gameState.ToString())  
 break  
  
 case 4:  
 Console.WriteLine("Exiting game.")  
 break  
 }  
  
} while (userChoice != 4)  
  
Console.ReadLine()

## Card Class

class Card  
{  
 string rank  
 string suit  
 string colour  
  
 Card(string rank, string suit)  
 Set rank, suit, and colour  
  
 Card(string colour)  
 Joker constructor  
  
 Card(int number)  
 Map to proper rank/suit or Joker  
  
 string ToString()  
 Return formatted rank/suit  
 bool Equals(Card other)  
 Compare all fields  
  
 int GetHashCode()  
 Hash of fields  
}

## Deck Class

class Deck  
{  
 List<Card> cards  
  
 int CardsLeft  
 get → return cards.Count  
  
 Deck()  
 Empty deck  
  
 Deck(bool hasJokers)  
 52 or 54 cards  
  
 Deck(List<Card> customCards)  
 Assign custom deck  
  
 Card Draw()  
 Remove and return top card  
  
 void Shuffle()  
 Shuffle cards  
  
 Card Peek()  
 Return top card  
  
 void PlaceOnTop(Card c)  
 Add c at index 0  
  
 string ToString()  
 Return count  
}

## Hand Class

class Hand  
{  
 List<Card> cards  
 string[] suitPriority  
  
 int Size  
 get → return cards.Count  
  
 Hand(string[] priority)  
 Initialize hand  
  
 void AddCard(Card c)  
 Add and reorder  
  
 Card RemoveCard()  
 Remove one card  
  
 bool Contains(Card target)  
 Return true/false  
  
 void OrderBySuit()  
 Sort by suitPriority  
  
 string ToString()  
 Return formatted hand  
}

## GameState Class

class GameState  
{  
 Deck drawDeck  
 Deck discardPile  
 List<Hand> hands  
 string[] suitPriority  
  
 GameState(bool jokersIncluded, string[] suitPriority)  
 Setup decks and hands  
  
 void DealHands(int players, int cardsPerPlayer)  
 Deal cards to each hand  
 Recycle discard if needed  
  
 string ToString()  
 Return all game info  
}