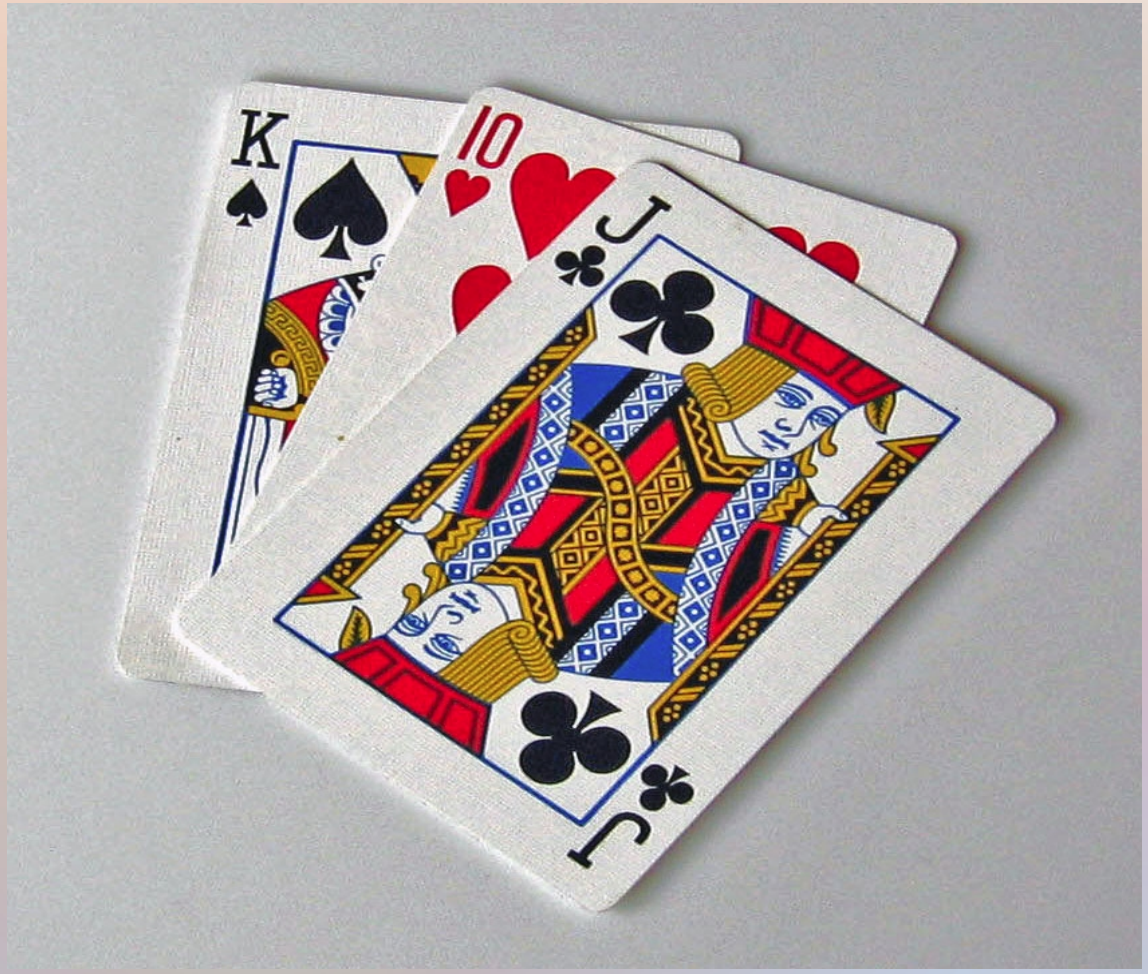
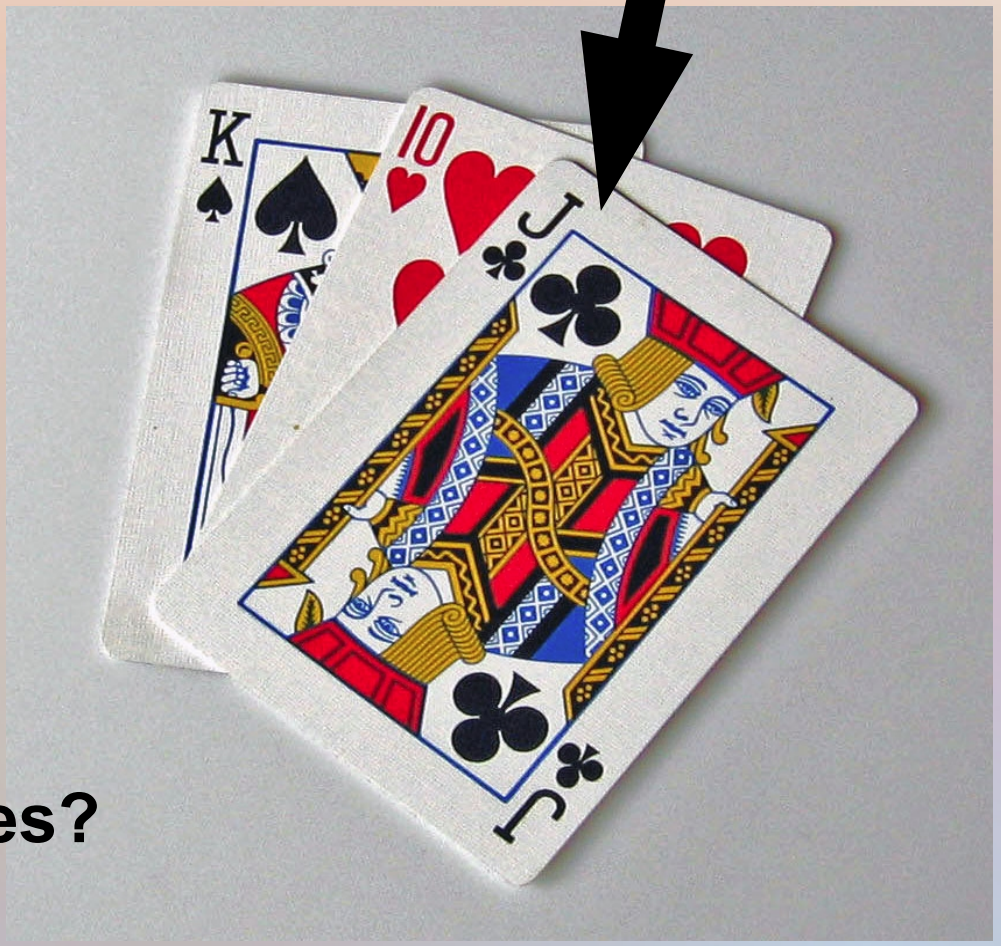
Documenting a Program for presentation

First explain the problem to be solved and the model used

Problem solving:
What is a deck of cards?
How can cards be modeled?



This card can be card 18 in the deck and is a Jack of Clubs



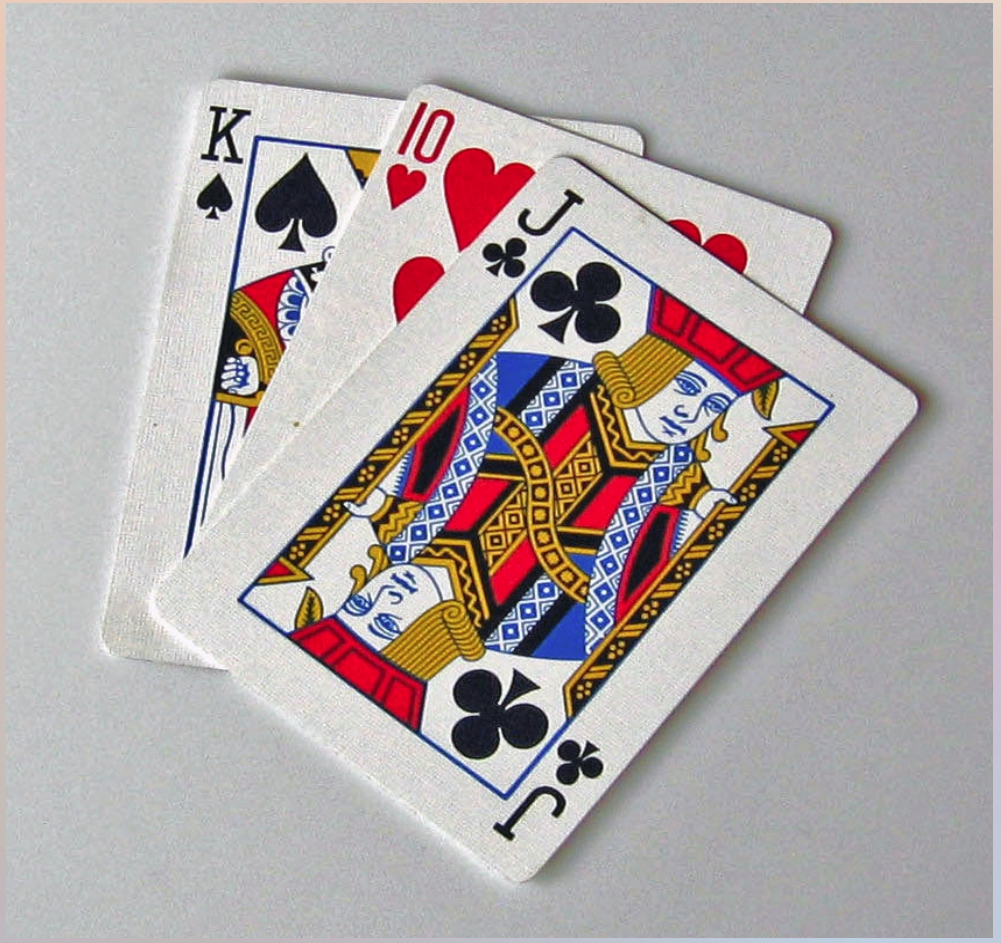
The card number is 18
Clubs is the suit
Jack is the face value

Problem solving:
How many cards?
How many suits?
How many face values?

Problem Solving:

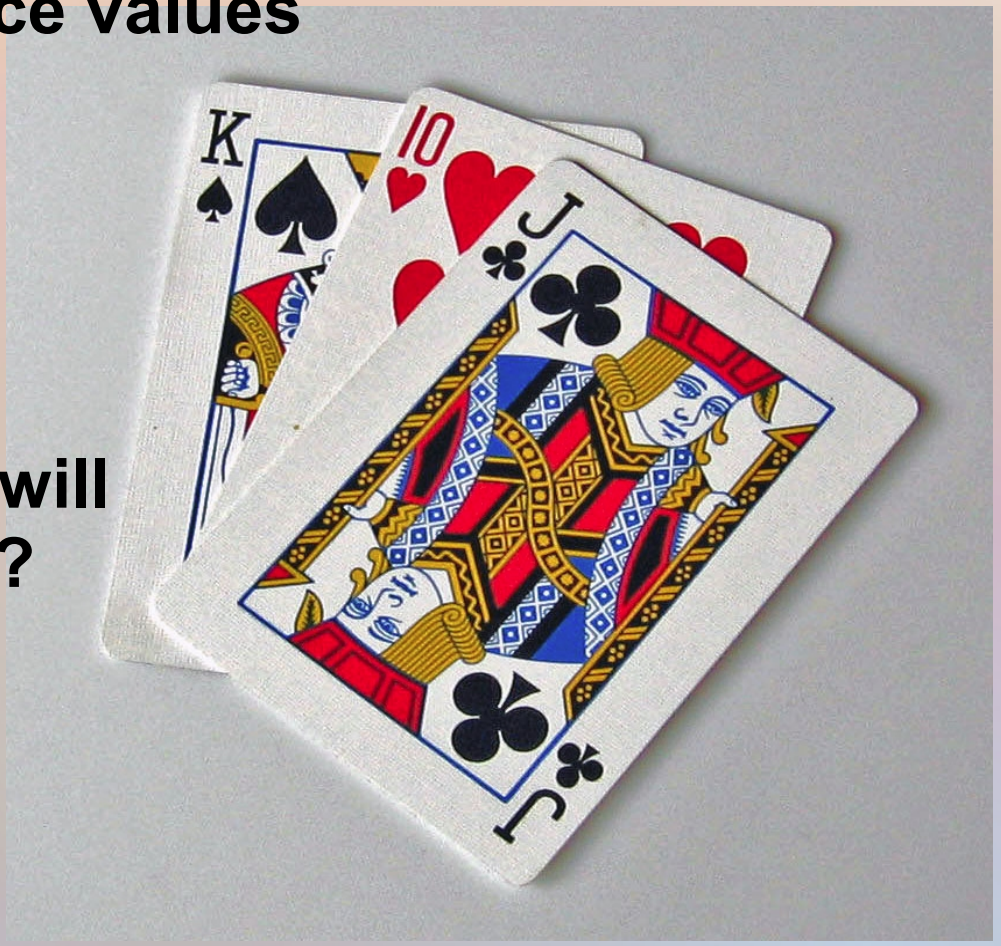
Suit values are
0- Hearts, 1-Diamonds
2- Clubs, 3-Spades

Face Values are
0-Ace
1-Two
2-Three
3-Four
4-Five
5-Six
6-Seven
7-Eight
8-Nine
9-Ten
10-Jack
11-Queen
12-King



So the card deck array will need
a row for each card

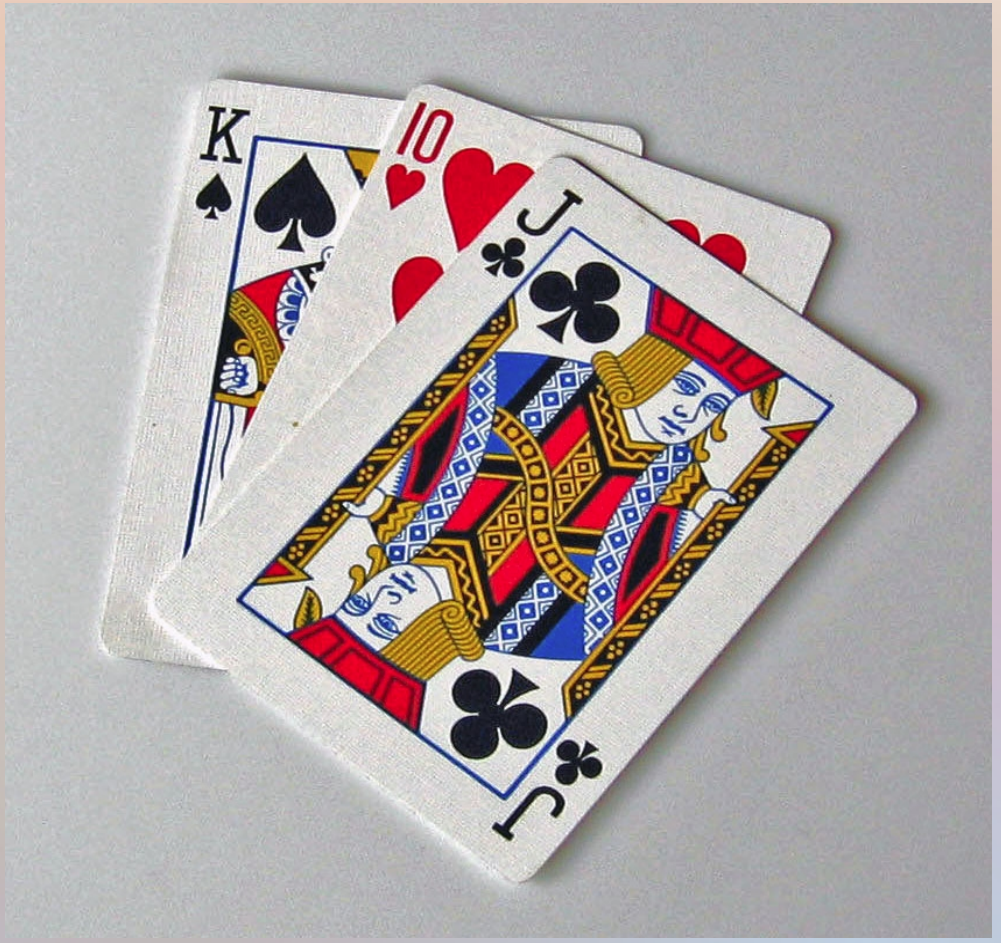
and each card row will need
elements for suit and face values



Problem Solving:
How many dimensions will
CardDeck array require?

Use an array to represent cards
First dimension:
The element represents the card row number
Each array row is a card
How large is this array dimension?

CardDeck[??]



How much information needed for each card?
How many dimensions does the array need?

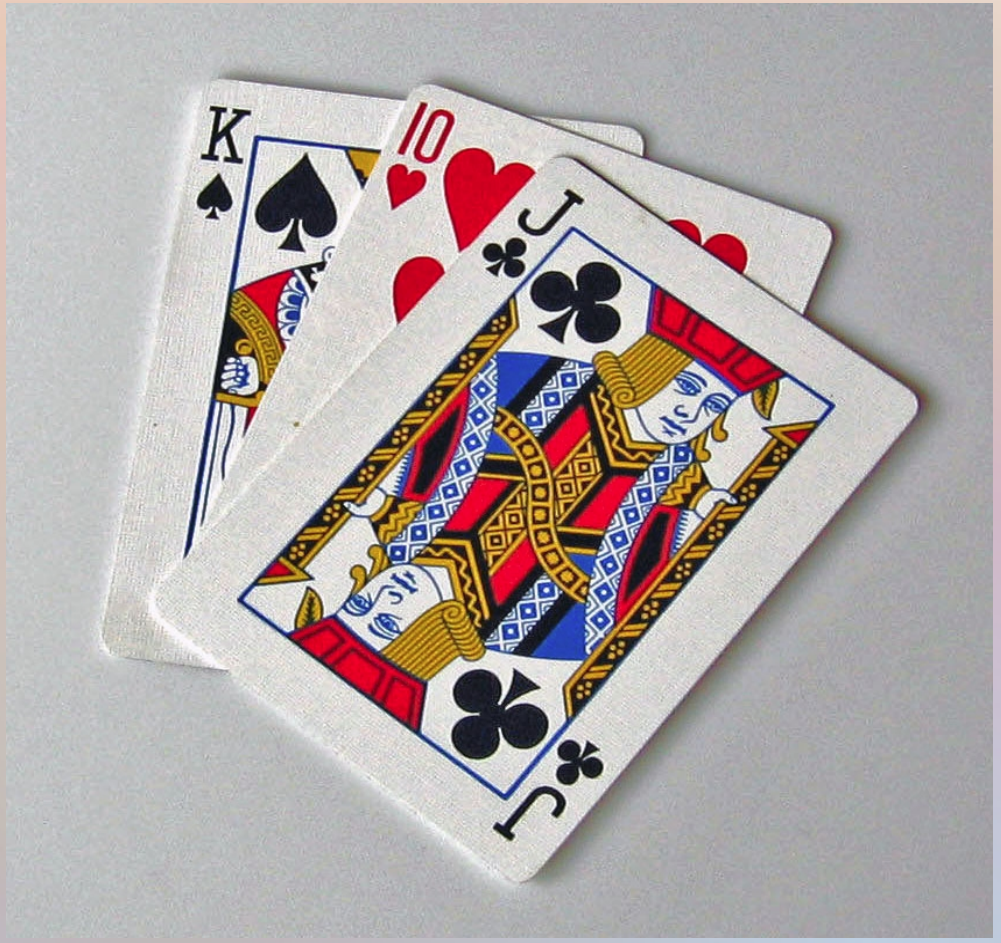
CardDeck[??][??]

Problem solving:

Each card has a number for the card row

and elements for the card suit and face values

How many elements needed for each card?



Problem Solving:

The *first dimension*: each element represents
card row in deck

The *second dimension* represents
card characteristic elements

first element holds suit number of card

second element holds face number of card

According to prior definition table

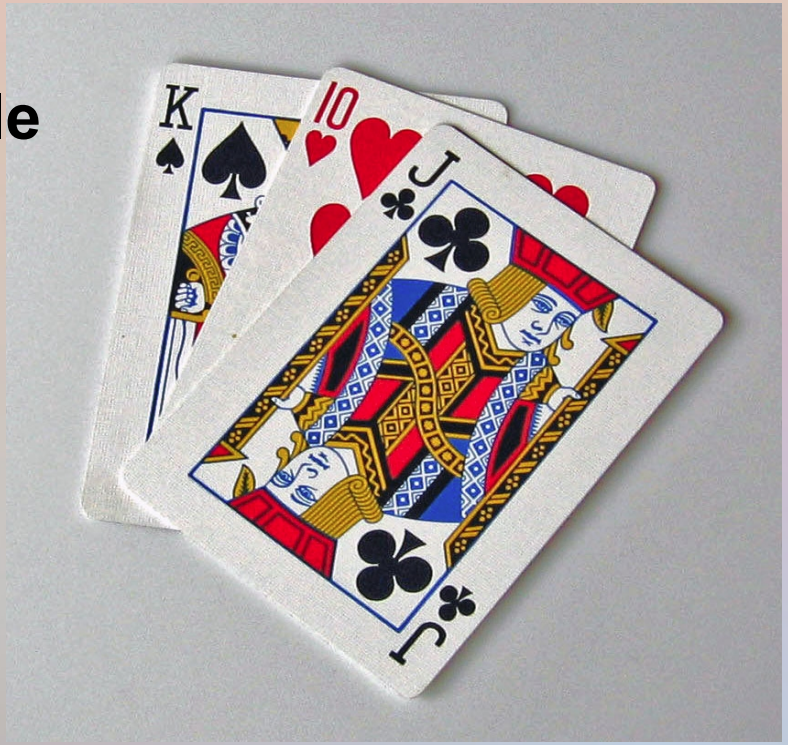
suit

Card[18][0] = 2 (Clubs)

Card[18][1] = 10(Jack)

face

Card Row Number



Card#	Suit #	Face Value
Row#	Column [0]	Column [1]
0	0 - Hearts	0 - Ace
1	0 - Hearts	1 - Two
2	0 - Hearts	2 - Three
3	0 - Hearts	3- Four
4	0 - Hearts	4 - Five
5	0 - Hearts	5 - Six
6	0 - Hearts	6 - Seven
7	0 - Hearts	7 - Eight
8	0 - Hearts	8 - Nine
9	0 - Hearts	9 - Ten
10	0 - Hearts	10 - Jack
11	0 - Hearts	11 - Queen
12	0 - Hearts	12 - King
13	1 - Diamonds	0 - Ace
14	1 - Diamonds	1 - Two
15	1 - Diamonds	2 - Three
16	1 - Diamonds	3- Four

A multi-dimension array can be visualized as a table with Rows being the first dimension and Columns being the second

Overall Program

- Statement: What does it do?
- Process Diagram
- Sequence steps
- Objects
- Functions

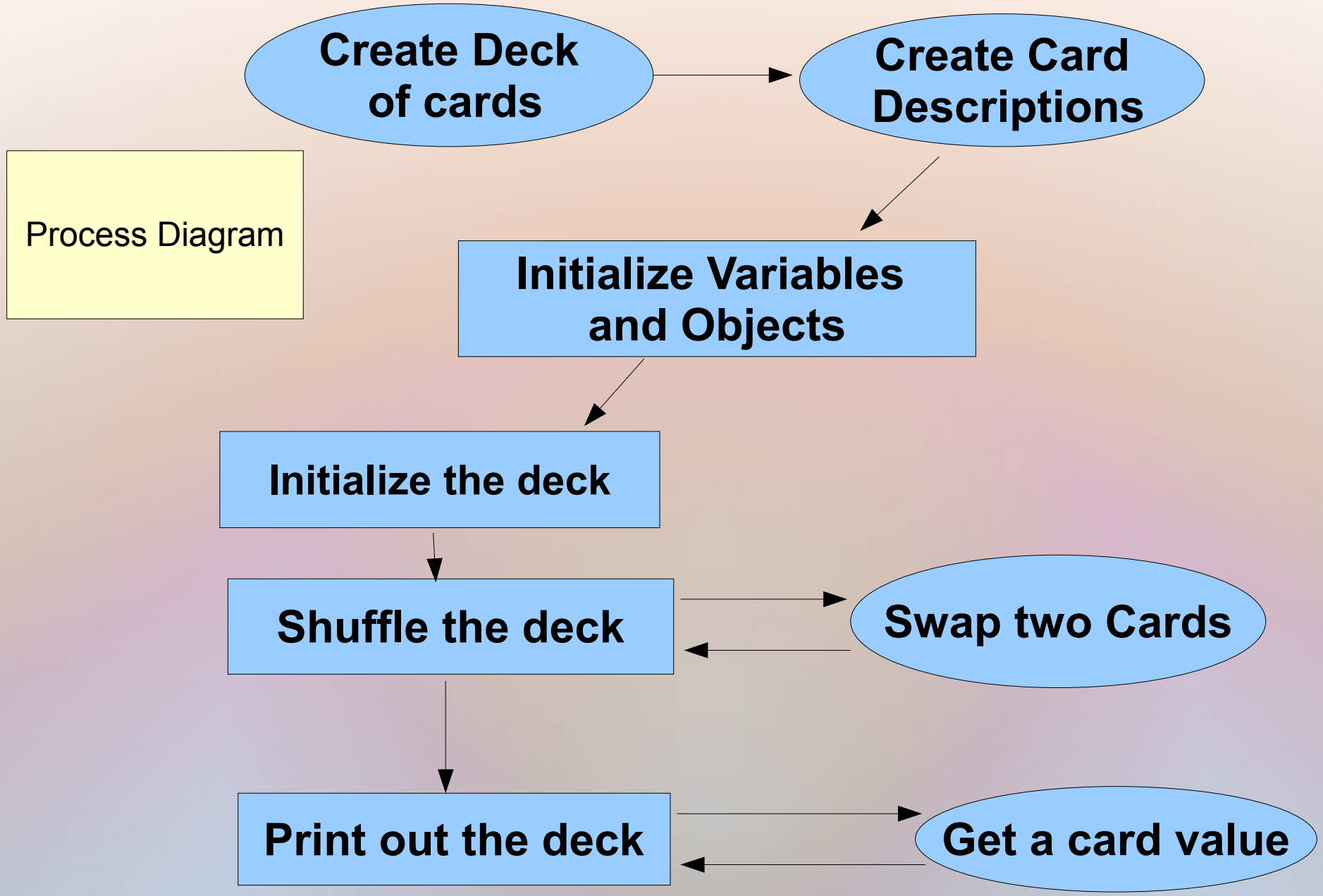
Functions

- Statement: What does it do?
- Calling and return arguments
- Sequence steps
- Objects
- Functions

What does the program do?:

Creates and use a model for a deck of cards.

The deck of cards is created and initialized
Then deck of cards is shuffled and printed.


Create Deck of cards
Create Card Descriptions
Process Diagram
Initialize Variables and Objects
Initialize the deck
Shuffle the deck
Swap two Cards
Print out the deck
Get a card value

Sequence: main()

- **Declare:**
  - **card deck**
  - **card text descriptions**

- **Initialize:**
  - **variables**
  - **card deck contents (function call)**
  - **random number generator (function call)**

- **Print out the card deck (function call)**
- **Shuffle the deck (function call)**
- **Print out the card deck (function call)**

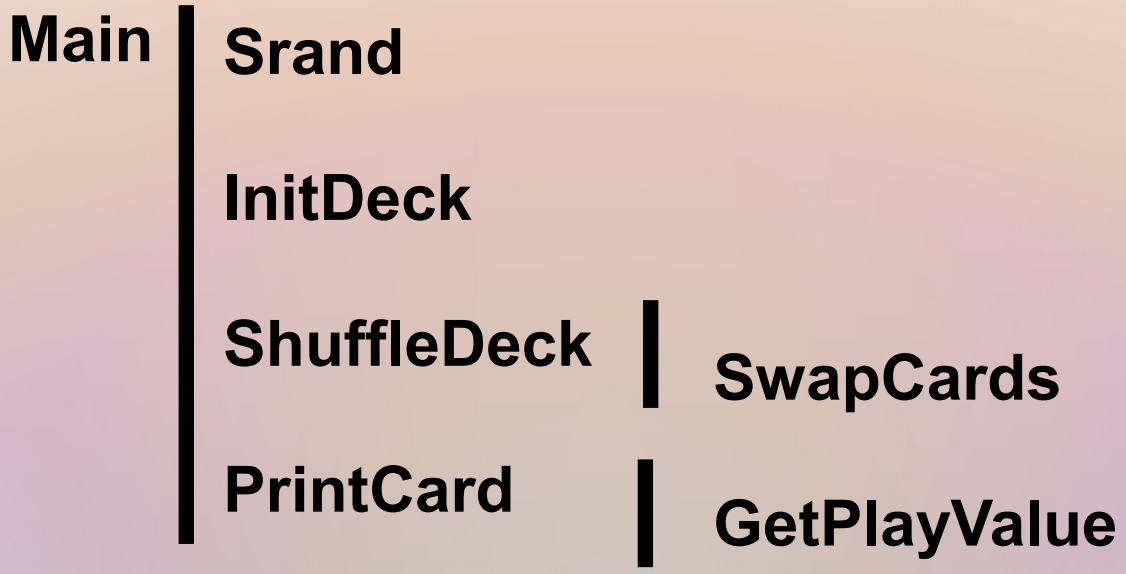
Objects:

- Card deck array
- Card text descriptions
- variables

Functions:

- Main function
- Initialize deck
- Print a card
  - Get card value
- Shuffle the deck
  - Rand & SRand
  - Swap two cards

Program Hierarchy



Function Name: InitDeck

Initialize the deck with card values
Calling Arguments: card deck
Return Argument: none
Sequence:
    Create variables
    Loop through deck
    Each card row:
        Set the suit value
        Set the face value
Objects:
    loop increment variable
Functions:
    none

Function Name: Shuffle

Shuffle the card deck
Calling Arguments: card deck
Return Argument: none
Sequence:
    Create local variables
    Loop through each card row
    Swap that card and another
        random card row in deck
Objects:
    loop increment variable
    Source and destination card variable
Functions needed:
    Swap two cards

Function Name: SwapCards

Swap two cards in deck
Calling Arguments:
    deck of cards, card1, card2
Return Argument:
    none
Sequence:
    Create local variables
    Loop to do card face & suit columns
        set temporary value to card1
        set card1 value to card2
        set card2 to temporary value
Objects needed:
    loop increment variable
    temporary variable
Functions needed:
    none

Function Name: PrintCard

Print a card suit and face value
Calling Arguments:
    card deck, card to print
Return Argument:
    none
Sequence:
    Create local variables
    retrieve suit & face value from row
    get the play value of card
    print card information
Objects needed:
    local face, suit, play variables
Functions needed:
    GetPlayValue

Function Name: GetPlayValue

Determine the play value of a card
Calling Arguments:
    deck of cards, card number
Return Argument:
    play value of card
Process List:
    Create local variables
    retrieve suit & face value of card row
    determine play value of card
    return card value
Objects needed:
    local face, suit, play variables
Functions needed:
    none