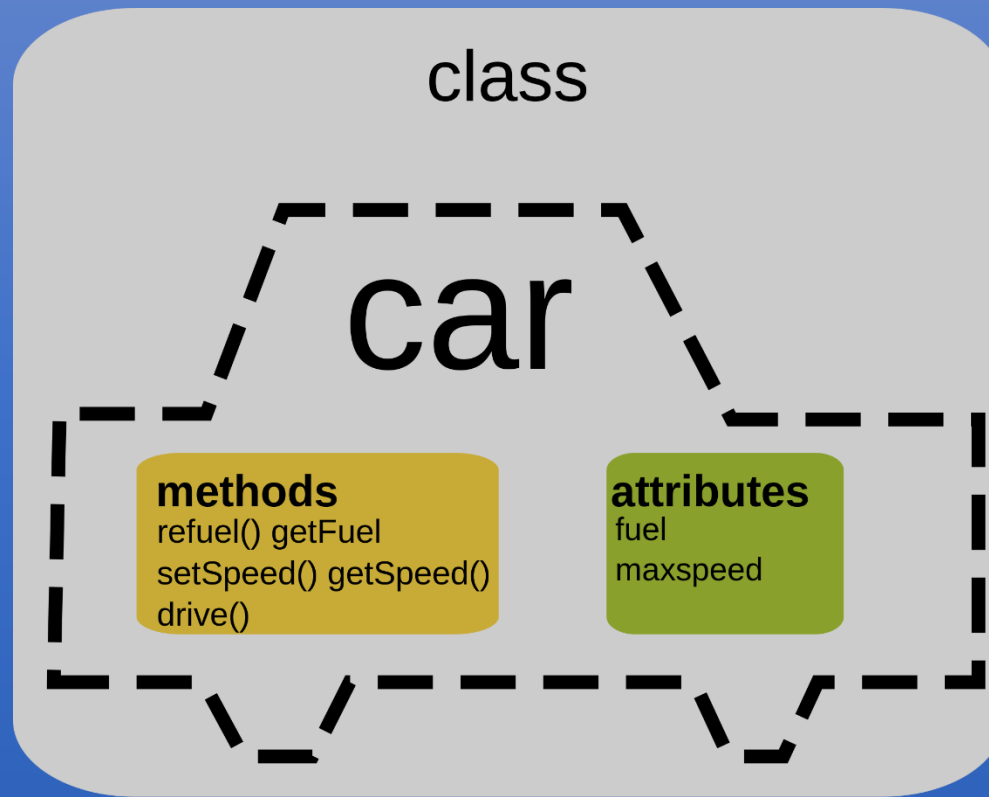


Lab 4:

Class Design



Lab 4: Class Design

Content



- Learn how to design a class (composite class)
- Practice

PRACTICE



Exercise 1



- Objective:
 - Design simple program with composite class
- Problem:
 - Simulate the deck of card with card **shuffling** and **dealing** by develop class called "DeckOfCards".
 - (This class can then be used to implement applications that play specific card games).



Exercise 1 (cont)



- **Sample output**

List of cards in the deck

9 of Hearts	5 of Diamonds	4 of Clubs	A of Spades
10 of Clubs	5 of Hearts	A of Clubs	9 of Clubs
6 of Spades	10 of Hearts	K of Diamonds	J of Hearts
7 of Clubs	4 of Diamonds	J of Spades	A of Hearts
8 of Spades	5 of Clubs	2 of Spades	A of Diamonds
K of Clubs	Q of Hearts	3 of Spades	4 of Hearts
2 of Hearts	J of Diamonds	6 of Hearts	10 of Diamonds
3 of Diamonds	8 of Hearts	9 of Diamonds	2 of Diamonds
4 of Spades	6 of Clubs	7 of Spades	9 of Spades
5 of Spades	K of Hearts	7 of Hearts	Q of Diamonds
3 of Clubs	J of Clubs	3 of Hearts	Q of Clubs
8 of Clubs	6 of Diamonds	K of Spades	10 of Spades
7 of Diamonds	Q of Spades	2 of Clubs	8 of Diamonds

Requirement



- Simulate the deck of card with card shuffling and dealing.



This class can then be used to implement applications that play specific card games

Classes and Methods (1/3)



Simulate the **deck of card** with card **shuffling** and **dealing**

- Class identification:
 - Get from **noun** of the requirement.
- Card
- Deck (contains many Cards)
- Method identification:
 - Get from the **verb** of the requirement
- Shuffler
- Deal

new 를 이용하면 모두 reference
array object = reference
나머지 = value

Classes and Methods (2/3)



- Card:

- String rank

- A, 2, 3, ..., Q, K,

- String suit





















































- Spade, Heart, Club or Diamond



- Deck

- Card[] deck
 - void shuffle()
 - Card deal()

Example set of 52 playing cards; 13 of each suit clubs, diamonds, hearts, and spades

	Ace	2	3	4	5	6	7	8	9	10	Jack	Queen	King
Clubs:													
Diamonds:													
Hearts:													
Spades:													

Classes and Methods (3/3)



- Class Card
 - Members
 - **DIAMOND** : **int**
 - **CLUB** : **int**
 - **HEART** : **int**
 - **SPADE** : **int**
 - **SUIT_STRING**: **String**[]
 - **RANK_STRING**: **String**[]
 - **rank** : **int**
 - **suit** : **int**
 - Methods
 - **Card(int, int)**
 - **getFace(): int**
 - **getSuit(): int**
 - **toString(): String**
- Class DeckOfCards (composite class)
 - Members:
 - **TOTAL_CARDS = 52**: **int**
 - **deck** : **Card**[]
 - **randNum** : **Random**
 - **curCard** : **int**
 - Methods:
 - **DeckOfCards()**
 - **shuffle(): void**
 - **deal(): Card**

Exercise 2



- Objective:
 - Design simple address book which store the information of people
 - User can have multiple phone number (max = 3)
 - User can add, delete and search information about people

Name: Barack Obama
Birthday: August 4, 1961
Mobile: 01012345678
Fax: 1111-1111-11
Email: BO@whitehouse.org
Company: White House

