

2024년 상반기 K-디지털 트레이닝

# JAVA

[KB] IT's Your Life

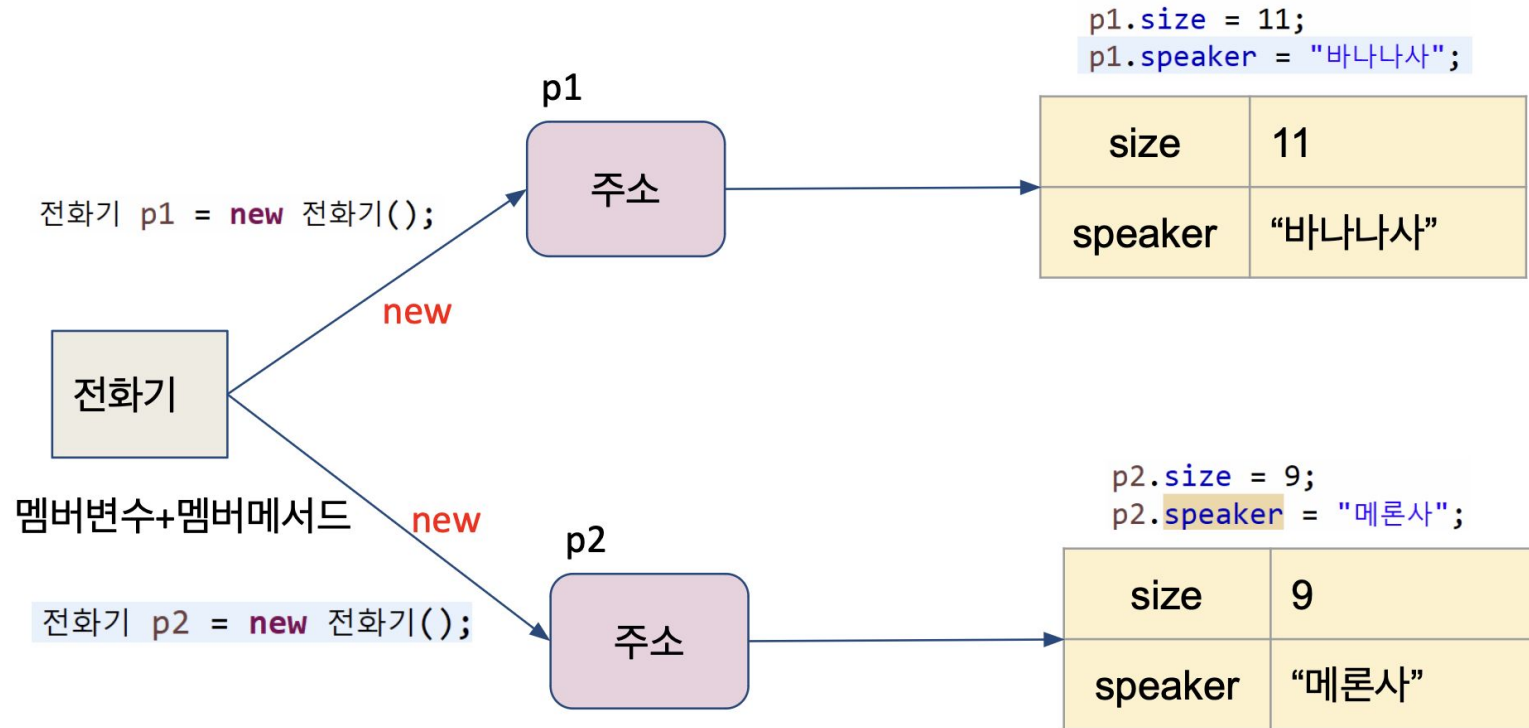


- 다음을 프로그래밍하시오.
- id와 pw가 각각 root, 1234와 일치하면 새로운 일기장 프레임 오픈
- 일기장 내용 입력하고 오늘의 날짜로 파일을 생성하여 날짜, 제목, 내용을 저장

swing이 아닌 cli방식으로 변경  
가능



- 다음 그림을 설명하시오.



## - 다음을 구현하시오.

### Student 클래스:

- id: 학번
- name: 이름
- grade: 학년
- department: 학과

### 프로그램 명세서

### Subject 클래스:

- name: 과목명
- score: 점수

### Main 클래스:

- main 메서드에서 다음을 수행합니다.
  - 새로운 과목 및 학생을 생성하고, 강좌에 추가합니다.
  - 학생이나 과목을 제거하고, 강좌 정보를 출력합니다.
  - 각 학생들의 평균 점수를 출력합니다.

### Course 클래스:

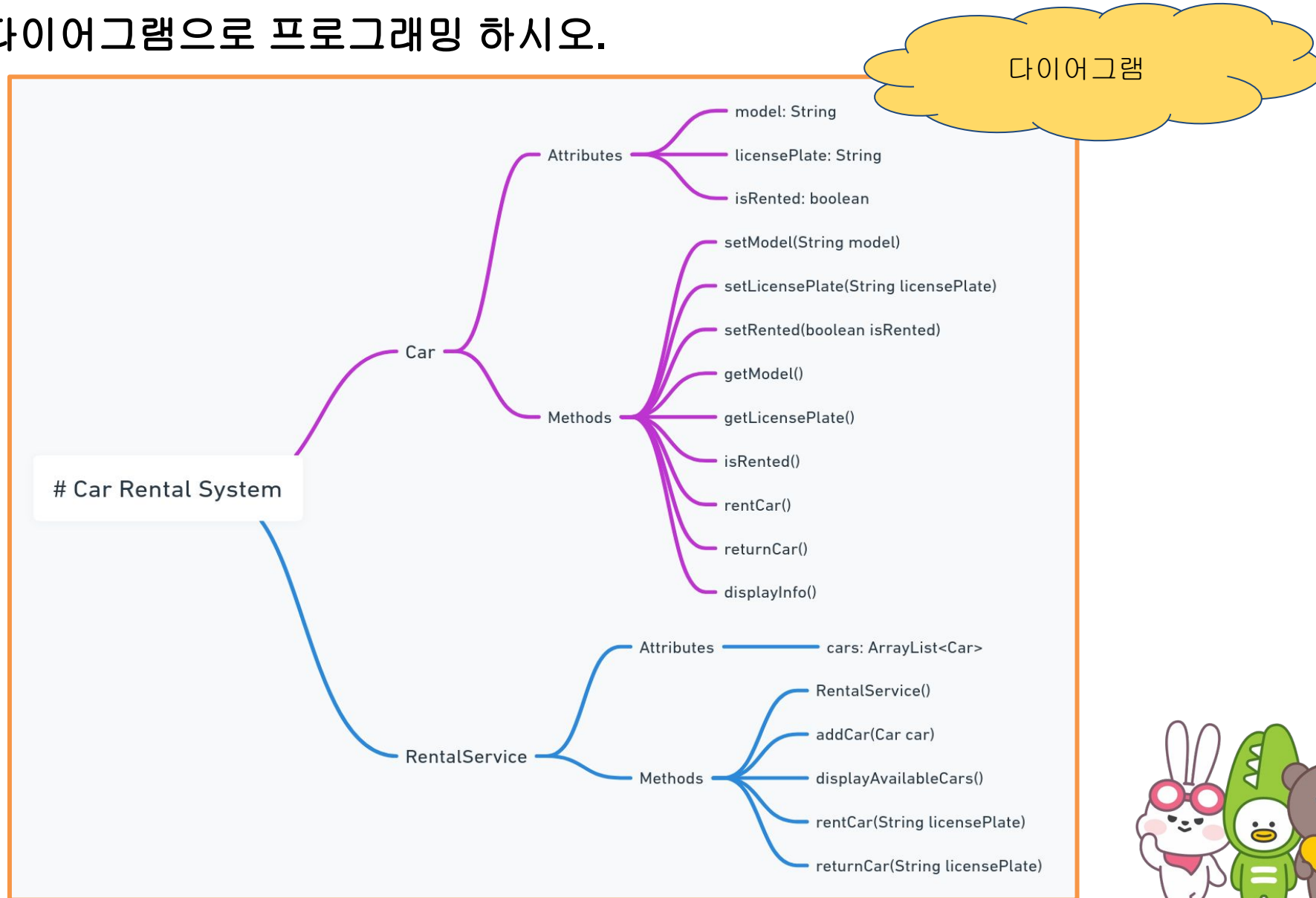
- courseName: 강좌명
- maxStudents: 최대 수강 인원
- currentStudents: 현재 수강 인원
- students: 수강 신청한 학생 목록 (Student 객체 배열)
- subjects: 과목 목록 (Subject 객체 배열)
- addStudent(Student student): 학생 추가 메서드
- removeStudent(Student student): 학생 삭제 메서드
- addSubject(Subject subject): 과목 추가 메서드
- removeSubject(Subject subject): 과목 삭제 메서드
- getAverageScore(): 각 학생들의 평균 점수를 계산하는 메서드
- printCourseInfo(): 강좌 정보를 출력하는 메서드

```
Run
basic_q4_array x Main x

/Library/Java/JavaVirtualMachines/jdk-17.jdk/Contents/Home/bin/java
-----
강좌명: 자바 프로그래밍
수강 인원: 5/30
수강 학생 목록:
홍길동0
홍길동1
홍길동2
홍길동3
홍길동4
과목 목록:
[_0619_class_deep.deep_q2_class.Subject@2a84aee7, _0619_class_deep.
자바 기초
jsp
python
spring
웹 기본

Process finished with exit code 0
```

- 다음 다이어그램으로 프로그래밍 하시오.



```

Run
Main x
/Users/alicia/Library/Java/JavaVirtualMachines/corretto-17.0.9/Content:
<<<<<<렌탈 차량 등록>>>>>>
Model: Toyota Camry, License Plate: 123ABC, Rented: false
Model: Honda Civic, License Plate: 456DEF, Rented: false

<<<<<<렌탈 시작>>>>>>
Car rented: 123ABC
현재까지의 렌탈 차량 목록-----
Model: Honda Civic, License Plate: 456DEF, Rented: false
Car returned: 123ABC
현재까지의 렌탈 차량 목록-----
Model: Toyota Camry, License Plate: 123ABC, Rented: false
Model: Honda Civic, License Plate: 456DEF, Rented: false

Process finished with exit code 0
|

```

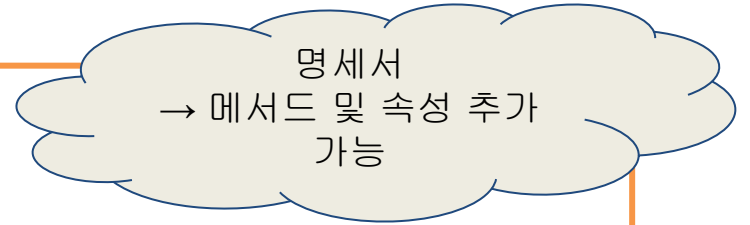


## Car 클래스

- 속성
  - ``model``: 자동차 모델 (String)
  - ``licensePlate``: 자동차 번호판 (String)
  - ``isRented``: 렌탈 상태 (boolean)
- 메서드
  - ``void setModel(String model)``: 모델 설정
  - ``void setLicensePlate(String licensePlate)``: 번호판 설정
  - ``void setRented(boolean isRented)``: 렌탈 상태 설정
  - ``String getModel()``: 모델 반환
  - ``String getLicensePlate()``: 번호판 반환
  - ``boolean isRented()``: 렌탈 상태 반환
  - ``void rentCar()``: 차 렌탈
  - ``void returnCar()``: 차 반환
  - ``void displayInfo()``: 자동차 정보 출력

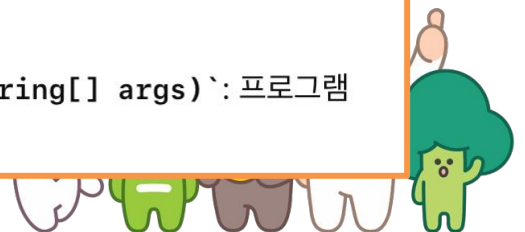
## RentalService 클래스

- 속성
  - ``ArrayList<Car> cars``: 자동차 목록
- 메서드
  - ``RentalService()``: 생성자, 자동차 목록 초기화
  - ``void addCar(Car car)``: 자동차 추가
  - ``void displayAvailableCars()``: 이용 가능한 자동차 정보 출력
  - ``void rentCar(String licensePlate)``: 특정 번호판의 자동차 렌탈
  - ``void returnCar(String licensePlate)``: 특정 번호판의 자동차 반환



## Main 클래스

- 메서드
  - ``public static void main(String[] args)``: 프로그램 실행





- 팀별로 주제를 정하십시오.
- 클래스를 정의하여 명세서를 완성하십시오.
- 클래스간의 관계를 UML이나 다이어그램으로 작성하십시오.
- 프로그램으로 완성하십시오.

수고하셨습니다!

