



Object – Oriented System Design

Programming Assignment #02

● Programming Assignment

- Template : 제공하지 않음.
- 업로드 : 각자의 프로젝트 폴더 (기존의 Self-Test업로드 방식과 동일)
- Commit Message : Assignment02 (git commit -m "Assignment02")
- 제출기한 : 5월 29일 밤 12시까지

CSLAB

● Price List Program

- 각 지역의 기름 가격을 유종 별로 기록하는 프로그램
- 아래의 상황을 가정한다.
 - 휘발유 가격은 공급가 + 공급가 * 부가가치세율
 - 경유 가격은 공급가 + 환경세 + 공급가 * 부가가치세율
 - 지역마다 부가가치세율이 다름.

CSLAB

● Class Oil

- private String company (공급회사)
- private String oilType (기름종류)
- private double consumerPrice (소비자 가격)
- Company, oilType을 인자로 받는 생성자 (consumerPrice는 0으로 초기화)
- consumerPrice 에 대한 getter
- consumerPrice 에 대한 setter
- 공급회사 이름과 유종을 String 형태로 반환하는 toString()
형식 -> company:
oil type :
- double getOilPrice() : 추상 클래스 (소비자 가격 계산)

CSLAB

● Class Gasoline

- Extends Oil
- private int price (공급가)
- private double VAT (부가가치세율)
- price, company를 인자로 받는 생성자
 - 인자로 받은 값으로 price 값과 company값을 초기화 한다.
 - oilType은 "gasoline"으로 초기화
 - VAT는 1~30사이의 임의의 정수 / 100 으로 초기화
- price를 반환하는 getOilPrice() 작성
- 공급회사 이름, 유종, 공급가를 String 형태로 반환하는 toString()
형식 -> company :
oil type :
supply price :

CSLAB

● Class Diesel

- Extends Oil
- private int price (공급가)
- private double VAT (부가가치세율) <- 랜덤으로 생성 (N %, N은 1~30 정수)
- private double envTax (환경세) <- 생성자 호출시 500으로 초기화
- price, company를 인자로 받는 생성자
 - 인자로 받은 값으로 price 값과 company값을 초기화 한다.
 - oilType은 "diesel"으로 초기화
 - VAT는 1~30사이의 임의의 정수 / 100 으로 초기화
 - envTax는 500으로 초기화
- price를 반환하는 getOilPrice()작성
- 공급회사 이름, 유종, 공급가를 String형태로 반환하는 toString() 형식 -> company :
oil Type :
supply price :

CSLAB

● Class FullArrayException

- FullArrayException은 exception클래스다.
 - ListArray가 Full 일 경우 발생
 - 인자가 없는 생성자
 - String 인자를 받는 생성자
 - Exception이 발생할 경우
 - List is full!
 - Try to expand...
 - How much will it expand?? 출력

CSLAB

● Class PriceList

- `private Oil[][] pricelist` (오브젝트를 저장할 클래스 타입 이중 배열)
`private int vertical` (배열의 세로 길이)
- `priceList` 의 행은 지역들 이며, 1열은 휘발유, 2열은 경유이다.
- 인자 없는 생성자
 - `vertical`을 2로 초기화
 - `priceList`를 2 x 2 배열로 선언
- `vertical`에 대한 `getter`
- `setPriceList`
 - parameter : `int i, int j, Oil oil`
 - return : `void`
 - `pricelist[i][j]`에 oil오브젝트를 초기화하고 소비자 가격을 초기화한다. (`setOilPrice()`)
- `extendList`
 - parameter : `int amount`
 - return : `void`
 - `priceList`를 인자로 받은 `amount`만큼 세로로 늘린다.
ex) `amount`가 2일 경우 기존 `priceList[2][2]` 에서 `priceList[4][2]`로...

CSLAB

● Class PriceList

- **method getRegionInfo()**
 - parameter : int i
 - return : 지정된 지역의 정보 (String 형식) <- toString() 사용
- **method printList()**
 - parameter, return : void
 - 전체 priceList를 출력한다

CSLAB

● Class PriceListApp

• Main Method

- 스캐너를 사용하여 input을 받는다
- PriceList 객체 생성
- 메인 메뉴 생성
 - (1) Add a oil to price list
 - (2) View Region Information
 - (3) Print the entire price list
 - (4) Exit the program

• 1~4 외의 숫자를 입력할 경우, "insert 1~4"출력

• (1)

- Try-catch 문을 이용하여 list가 full일 경우 exception이 발생하도록 한다
-> exception 처리 시 list 확장 후 "list is extended" 출력
- 스캐너로 추가할 oil의 정보를 입력받는다
- gasoline, diesel 순차적으로 입력받아서 초기화 한다 -> 초기화 후 "added" 출력

● Class PriceListApp

- (2)
 - 확인할 지역의 번호를 스캐너로 입력받는다
 - 정보출력
- (3)
 - price List출력
- (4)
 - “Exit Application”출력 후 프로그램 종료

CSLAB

● Sample Output

• 프로그램 첫 화면

```
=====Main Menu=====
(1) Add a oil to price list
(2) View Region Information
(3) Print the entire price list
(4) Exit the program
=====
```

• oil 추가하기

```
=====Main Menu=====
(1) Add a oil to price list
(2) View Region Information
(3) Print the entire price list
(4) Exit the program
=====
choose a menu : 1
company : wooyel-oil
supply price : 1200
gasoline added

company : sungjun-oil
supply price : 1000
diesel added
```

CSLAB

● Sample Output

- List가 full 일 때, 확장 후 추가

```
=====Main Menu=====
(1) Add a oil to price list
(2) View Region Information
(3) Print the entire price list
(4) Exit the program
=====
choose a menu : 1
list is full!
try to expand...
How much will it expand?
3
list is extended.

company : Flex-oil
supply price : 2400
gasoline added

company : moya-oil
supply price : 2200
diesel added
```

LAB

● Samle Output

• 2 단계

```
=====Main Menu=====
(1) Add a oil to price list
(2) View Region Information
(3) Print the entire price list
(4) Exit the program
=====
choose a menu : 2
region number to view : 3
company : Flex-oil
oilType : gasoline
Supply Price : 2400

company : moya-oil
oilType : diesel
Supply Price : 2200
```

```
=====Main Menu=====
(1) Add a oil to price list
(2) View Region Information
(3) Print the entire price list
(4) Exit the program
=====
choose a menu : 2
region number to view : 2
company : TheQ-oil
oilType : gasoline
Supply Price : 1500

company : zino-oil
oilType : diesel
Supply Price : 1300
```

CSLAB

● Samle Output

- 배열의 초과 index를 입력
 - 에러메시지 출력

```
=====Main Menu=====
(1) Add a oil to price list
(2) View Region Information
(3) Print the entire price list
(4) Exit the program
=====
choose a menu : 2
region number to view : 5
out of index
=====Main Menu=====
```

CSLAB

● Sample Output

● 3단계

```
=====Main Menu=====
(1) Add a oil to price list
(2) View Region Information
(3) Print the entire price list
(4) Exit the program
=====
```

choose a menu : 3

	Gasoline	Diesel
Region #1	1368.00	1620.00
Region #2	1695.00	1826.00
Region #3	2784.00	3140.00

Price list 조회

```
=====Main Menu=====
(1) Add a oil to price list
(2) View Region Information
(3) Print the entire price list
(4) Exit the program
=====
```

choose a menu : 9

Insert 1 ~ 4

1~4 외의 값 입력

CSLAB

● Sample Output

• 4단계

```
=====Main Menu=====
(1) Add a oil to price list
(2) View Region Information
(3) Print the entire price list
(4) Exit the program
=====
choose a menu : 4
Exit application
```

프로그램 종료

CSLAB