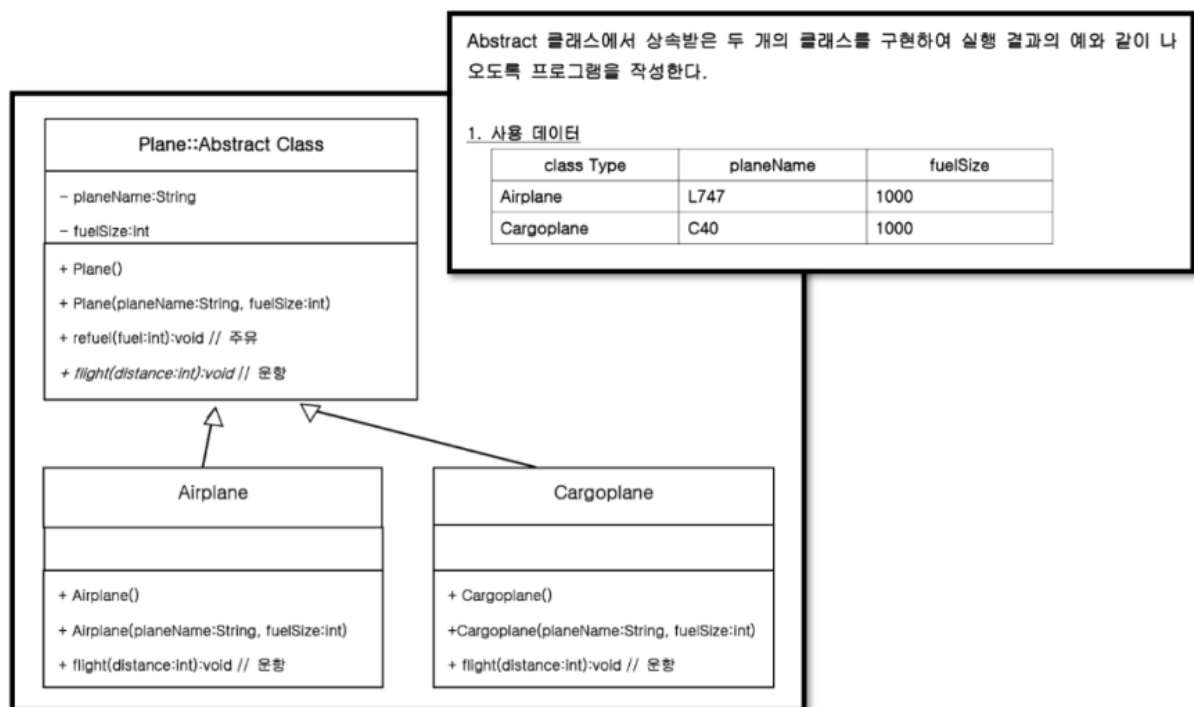




Question 3

Q. Inheritance Question 2



Package명	클래스명	메소드	설명
planetest	Plane	+Plane()	기본 생성자
		+Plane(planeName:String, fuelSize:int)	2개의 클래스 변수를 받는 생성자
		+refuel(fuel:int):void	일정 양의 연료 주입, 기존 연료가 증가됨
		+flight(distance:int):void	일정 거리 만큼 운항, 연료 감소 추상 함수 하위 객체에서 반드시 구현해야 함
	Airplane	+Airplane()	기본 생성자
		+Airplane(planeName:String, fuelSize:int)	2개의 클래스 변수를 받는 생성자
		+flight(distance:int):void	10 운항 시 연료 30감소
	Cargoplane	+Cargoplane()	기본 생성자
		+Cargoplane(planeName:String, fuelSize:int)	2개의 클래스 변수를 받는 생성자
		+flight(distance:int):void	10운항 시 연료 50감소
	PlaneTest	+ main(args:String[]):void	main 함수를 이용 하여 동작

4. PlaneTest 클래스 구조

```

public class PlaneTest {

    public static void main(String args[]) {
        // Airplane과 Cargoplane 객체 생성
        // 생성된 객체의 정보 출력
        // Airplane과 Cargoplane 객체에 100씩 운항
        // 운항 후 객체의 변경된 정보 출력
        // Airplane과 Cargoplane 객체에 200씩 주유
        // 주유 후 객체의 변경된 정보 출력
    }
}

```

5. 실행 결과 예

```

Plane      fuelSize
-----
L747      1000
C40       1000

100 운항
Plane      fuelSize
-----
L747      700
C40       500

200 주유
Plane      fuelSize
-----
L747      900
C40       700

```

A.

```
// Plane Code
package planetest;

public abstract class Plane {
    private String planeName;
    private int fuelSize;

    public Plane() {

    }

    public Plane(String planeName, int fuelSize) {
        this.planeName = planeName;
        this.fuelSize = fuelSize;
    }

    public void refuel(int fuel) {
        fuelSize += fuel;
        // this.setFuelSize(this.getFuelSize() + fuel);
    }

    public abstract void flight(int distance);

    public String getPlaneName() {
        return planeName;
    }

    public void setPlaneName(String planeName) {
        this.planeName = planeName;
    }

    public int getFuelSize() {
        return fuelSize;
    }

    public void setFuelSize(int fuelSize) {
        this.fuelSize = fuelSize;
    }

    @Override
    public String toString() {
        return "\t" + planeName + "\t" + fuelSize;
    }
}

```

```
// Airplane Code
package planetest;

class Airplane extends Plane {
    public Airplane() {

    }
}

```

```

public Airplane(String planeName, int fuelSize) {
    super(planeName, fuelSize);
}

public void flight(int distance) {
    setFuelSize(getFuelSize() - (distance * 3));
}
}

```

```

// Cargoplane Code
package planetest;

class Cargoplane extends Plane {
    public Cargoplane() {

    }

    public Cargoplane(String planeName, int fuelSize) {
        super(planeName, fuelSize);
    }

    public void flight(int distance) {
        setFuelSize(getFuelSize() - (distance * 5));
    }
}

```

```

// PlaneTest Code
package planetest;

public class PlaneTest {
    public static void main(String[] args) {
        Plane airplane = new Airplane("L747", 1000);
        Plane cargoplane = new Cargoplane("C40", 1000);

        System.out.println("\tPlane\tfuelSize");
        System.out.println("-----");
        System.out.println(airplane);
        System.out.println(cargoplane);

        airplane.flight(100);
        cargoplane.flight(100);

        System.out.println("\n100 운항\n\tPlane\tfuelSize");
        System.out.println("-----");
        System.out.println(airplane);
        System.out.println(cargoplane);

        airplane.refuel(200);
        cargoplane.refuel(200);

        System.out.println("\n200 주유\n\tPlane\tfuelSize");
        System.out.println("-----");
    }
}

```

```
        System.out.println(airplane);
        System.out.println(cargoplane);
    }
}
```

Plane	fuelSize
L747	1000
C40	1000

100 운항

Plane	fuelSize
L747	700
C40	500

200 주유

Plane	fuelSize
L747	900
C40	700