다음과 같은 내용으로 Multiplication 클래스가 있다.

class Multiplication {

private int dan;

private int number;

Multiplication() {}

Multiplication(int dan) {

this.dan = dan;

}

Multiplication(int dan, int number){

this.dan = dan;

this.number = number;

}

public void printPart() {

if (number == 0) {

for(int n=1; n <= 9; n++)

System.out.print("\t"+dan + "\*" + n+ "="+dan\*n);

System.out.println();

} else {

System.out.println(dan \* number);

}

}

}

1. 상속 구문을 적용하여 GuGuDanExpr 클래스를 구현한다.

- Multiplication 클래스를 상속한다.

- GuGuDanExpr 클래스의 생성자 사양

GuGuDanExpr()

GuGuDanExpr(int dan)

GuGuDanExpr(int dan, int number)

- GuGuDanExpr 클래스의 메서드 사양

static void printAll()

다음에 제시된 출력 방식으로 1단부터 9단까지 모두 출력

1\*1=1 1\*2=2 ……………….

2\*1=2 2\*2=4 ………………..

:

9\*1=1 9\*2=2 ……………….

2. 다음에 제시된 내용을 수행하는 메인 클래스 GuGuDan 을 구현한다.

1부터 20사이의 난수를 2개를 추출하여 각각 dan 변수와 number 변수에 담는다.

(1) dan 과 number 이 모두 1~9 사이이면 dan\*number 의 구구단을 출력한다.

GuGuDanExpr 객체를 생성(생성자를 통해서 dan과 number에 대한 데이터를 전달하여 초기화한다.)하고 printPart() 를 호출한다. 단이 3, number가 4로 추출된다면 3 \* 4 = 12 를 출력한다.

(2) dan 은 1~9 사이이고 number 가 10 이상이면 GuGuDanExpr 객체를 생성

(생성자를 통해서 dan에 대한 정보를 전달하여 초기화한다.)하고 printPart()

를 호출한다.

추출된 dan의 숫자가 2 라면….

2단 : 2 \* 1 = 1 2 \* 2 = 2 2 \* 3 = 6 ………………

(3) dan 의 값이 10 이상이면 GuGuDanExpr 의 static 메서드 printAll() 을 호출

하여 1단부터 9단까지의 값들을 행 단위로 출력한다.

1 \* 1 = 1 1 \* 2 = 2 1 \* 3 = 3 ………………..

2 \* 1 = 1 2 \* 2 = 2 2 \* 3 = 6 ………………...

……………..

9 \* 1 = 9 9 \* 2 = 18 9 \* 3 = 27………………...

|  |
| --- |
| Multiplication |
| - dan : int  - number : int |
| Multiplication ()  Multiplication (dan : int)  Multiplication ( dan : int, number : int)  + printPart() : void |

|  |
| --- |
| GuGuDan |
|  |
| + main(String args[]) : void |

|  |
| --- |
| GuGuDanExpr |
|  |
| GuGuDanExpr ()  GuGuDanExpr (dan : int)  GuGuDanExpr ( dan : int, number : int)  + printAll() : void |