1. 필요한 클래스들

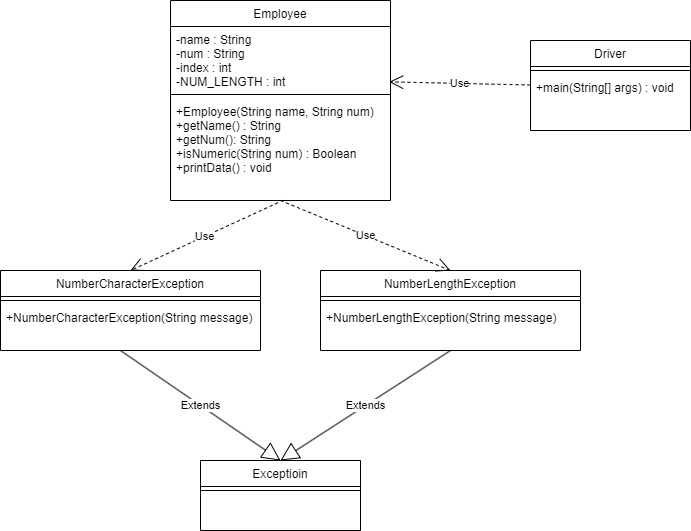
(1) Calculator: 입력을 받고 계산을 수행한다.

(2) DivisionByZeroException : 0또는 0에 가까운 수로 나눌 때의 예외를 나타낸다

(3) UnknownOpException : 허용되지 않은 연산자가 입력될 때의 예외를 나타낸다

(4) Driver: Employee클래스를 이용한다.

클래스 다이어그램



2. 클래스 설계

(1) Calculator 클래스

- 메소드

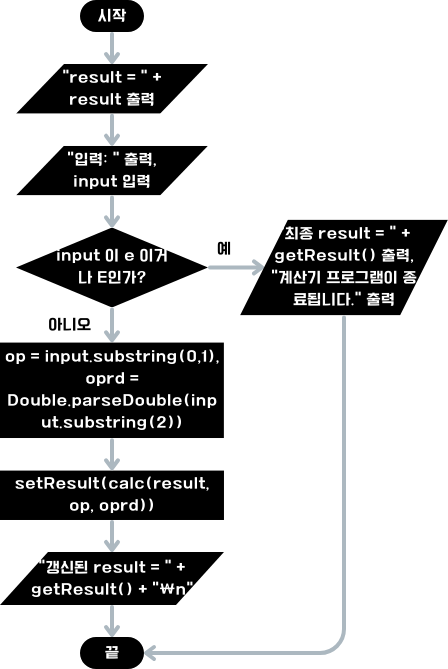
- inputData 메소드

- 목적: 입력을 받는다

- 매개변수: 없음

- 반환값: 없음

- 알고리즘(순서도)



- calc 메소드

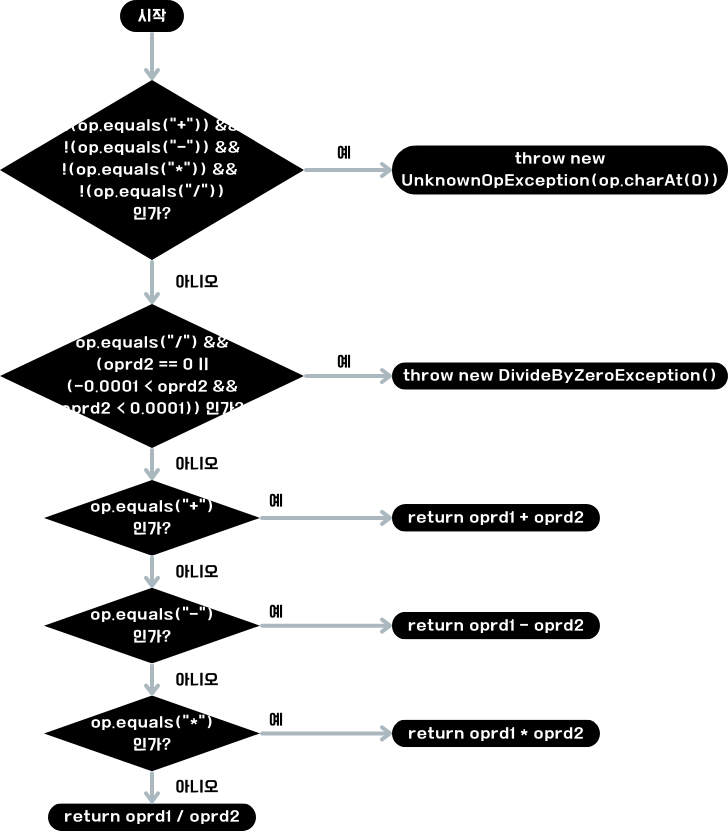
- 목적: 연산자를 확인하고, 연산을 수행한다

- 매개변수: double oprd1, String op, double oprd2

- 반환값: oprd1 + oprd2, oprd1 - oprd2,

oprd1 \* oprd2, oprd1 / oprd2 (연산자 종류에 따라)

- 알고리즘(순서도)



- handleDivisionByZeroException 메소드

- 목적: DivisionByZeroException 을 처리한다.

- 매개변수: 없음

- 반환값: 없음

- 알고리즘

1. "0 혹은 0에 아주 가까운 값으로 나누려고 시도하여\n프로그램이 중단되었습니다." 를 출력한다.

- handleUnknownOpException 메소드

- 목적: UnknownOpException 을 처리한다

- 매개변수: 없음

- 반환값: 없음

- 알고리즘

1. e.getMessage() 를 출력한다

2. "처음부터 다시 시작하세요." 를 출력한다

3. "연산식을 나타내는 각 라인(line)의 포맷: 연산자 숫자" 를 출력한다

4. "예:+ 3" 를 출력한다

5. "끝내려면 e(혹은 E)를 입력하세요.\n" 를 출력한다

6. setResult(0) 을 호출한다.

(1) Driver 클래스

- 변수: 없음

- main 메소드

- 지역 변수: 없음

- 알고리즘(순서도)

