

## 구현 기술 기록서

### 1. 숫자

- 각 숫자의 private void 함수를 선언
- If와 bool을 통해 연산이 아닌 단순 숫자 입력임을 확인

### 2. 사칙연산

- +, -, \*, /와 같은 연산자를 enum을 통해 저장
- 오류를 방지하기 위해 e나 E를 통해 표현하는 크고 작은 숫자들의 연산을 제한 및 오류 메시지 출력
- if문을 통해 -를 제외한 연산자를 빈칸에 입력오류 메시지 출력, 네 연산자 모두 ++, +- 등과 같이 연속적으로 입력 시 오류 메시지를 출력할 수 있도록 함
- 만약 빈칸에 -연산자를 입력 시 음수를 입력 받을 수 있도록 함

### 3. 소수점, 거듭제곱

- 소수점 버튼이나 키보드를 통해 소수점을 입력 받아 소수의 연산이 가능하도록 함
- 만약 현재 계산식의 끝이 연산 기호일 경우 오류 메시지를 출력하고 제한
- Split을 통해 문자열을 분할 해 소수점의 연속적인 사용만을 제어해 오류 메시지를 출력하고 소수끼리 연산은 가능하도록 함
- 거듭제곱을 사칙연산과 비슷한 형태로 구성해 밑과 지수를 입력받아 연산 E나 e를 포함한 크고 작은 숫자, 빈칸, 연산자 중복의 경우 각각 오류 메시지를 출력하고 제한
- 조건에 걸리지 않는 경우 enum에 저장된 power를 불러와 Math.Pow 함수를 이용해 연산 진행

### 4. 부호변환, 퍼센트, 역수, 절댓값, 제곱, 제곱근

- 각각 e나 E를 사용한 크고 작은 수, 빈칸이거나 연산자에 실행 시 오류 메시지를 출력하고 제한
- 제곱근의 경우 음수의 제곱근을 제한하고 오류 메시지를 출력
- 각각 -, \*1/100, 1/x, Math.Abs, Math.Pow, Math.Sqrt 함수나 연산을 이용해 연산 진행

### 5. =

- 빈칸에 입력, 연산자 이후 숫자 미입력 시 오류메시지 출력
- 각 연산자에 따라 switch-case를 통해 문자열을 분리해서 a와 b로 입력
- Switch-case를 통해 연산자에 따른 연산을 진행
- .ToString()을 통해 연산 결과값을 출력 및 SaveResult메서드를 통해 저장

### 6. AC, C

- AC는 All Clear로 현재 창에 있는 모든 숫자를 지우고 C는 한글자씩 지우는 기능을 가진 버튼
- AC 입력 시 string.Empty를 textbox에 적용해 입력 값 초기화

- C 입력 시 숫자나 연산기호가 존재 시 하나씩 지우고 빈 상태일 경우 빈 상태를 유지하도록 if-else 사용

#### 7. Memory 기능

- M, MS, MC, M+, M- 버튼을 통해 값을 저장, 단순 연산, 출력 등을 가능하도록 도움

- M버튼 입력 시 저장된 값을 MessageBox를통해 보여주고 현재 저장되어 있는 값을 클립보드에 복사

- MS버튼 입력 시 현재 창에 존재하는 값을 M에 저장

- MC버튼 입력 시 현재 저장되어 있는 M값을 0으로 초기화

- M+버튼 입력 시 현재 창에 존재하는 값을 M에 더해줌

- M-버튼 입력 시 현재 창에 존재하는 값을 M으로부터 빼줌

#### 8. 계산이력

- 현재까지 계산을 최대 5개까지 계산식 및 결과 값을 저장

- 이를 출력하기 위해 Form2를 만들어 Listbox를 팝업 창과 같은 형태로 출력할 수 있도록 함

- 계산 이력을 나타내기 위해 5칸이 있는 String배열을 선언 후 각각 “로 빈 칸 없이 채우고 앞서 결과값을 저장한 SaveResult함수를 선언

- SaveResult함수 내에는 for를 이용해 배열 첫 번째 자리부터 결과값 및 식을 저장하고, 밀어내는 식으로 최대 5번의 연산 기록을 저장

- 문자열 op를 Enum값 operationType으로 변환 받아 그 값의 정수 인덱스를 확인

- 인덱스를 이용해 연산에 사용된 연산자를 확인 후 배열에 식과 결과값을 저장 앞서 만들어둔 Form2에 계산 이력 배열을 넘겨 식과 결과를 출력할 수 있도록 함

#### 9. 기타

- 폰트와 계산기 아이콘, 색상을 Form의 디자인-속성을 통해 수정

- 속성을 통해 버튼에 마우스가 올라갈 시 손모양으로 바뀌게 설정 및 버튼 팝업을 통해 인지에 도움