휴먼컴퓨터인터페이스(HCI)

과제 #2

대화형 프로토타입



학과 : 컴퓨터소프트웨어학과

학번 : 2014726021

이름 : 김혜린

제출일 : 04/29

1. 개요

-모든 기능적 요구조건에 대한 구현 완성도

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 정수 | 벡터 | 행렬 | sin | x |
| 실수 | 벡터 내적 | 행렬의 역행렬 | cos | y |
| 복소수 | 벡터 외적 | 행렬식 | tan | z |
| 산술연산 | f | pi | exp | sqrt |
| 비교연산 | g | e | log | 결과출력 |

행렬의 역행렬을 제외한 모든 기능이 다 구현되어 있다.

- 오픈소스 라이브러리 의존성

자바스크립트 라이브러리인 JQuery 만 사용하였다.

- 사용성 향상에 기여하는 핵심적인 상호작용 방식

1) 비슷한 기능을 하는 여러 버튼을 합쳐서, 클릭할 때마다 변경된다.

대표적으로 (sin, cos, tan), (x, y, z), (<=, <, >, >=) 버튼들이 있다.

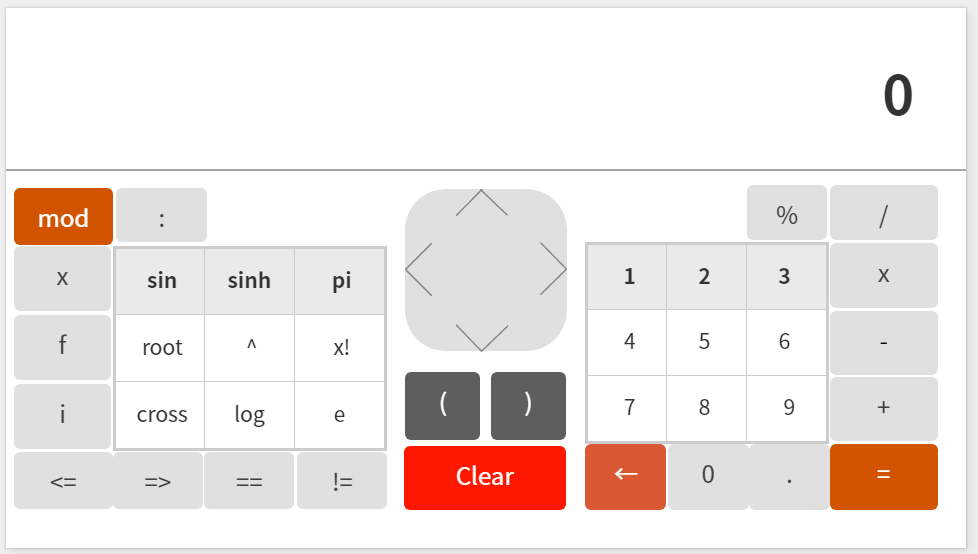
2) 백스페이스로 입력 받은 문자를 하나씩 지울 수 있게 한다.

3) 커서 위치를 움직여 해당 위치에서 입력이나 지우기를 할 수 있게 한다.

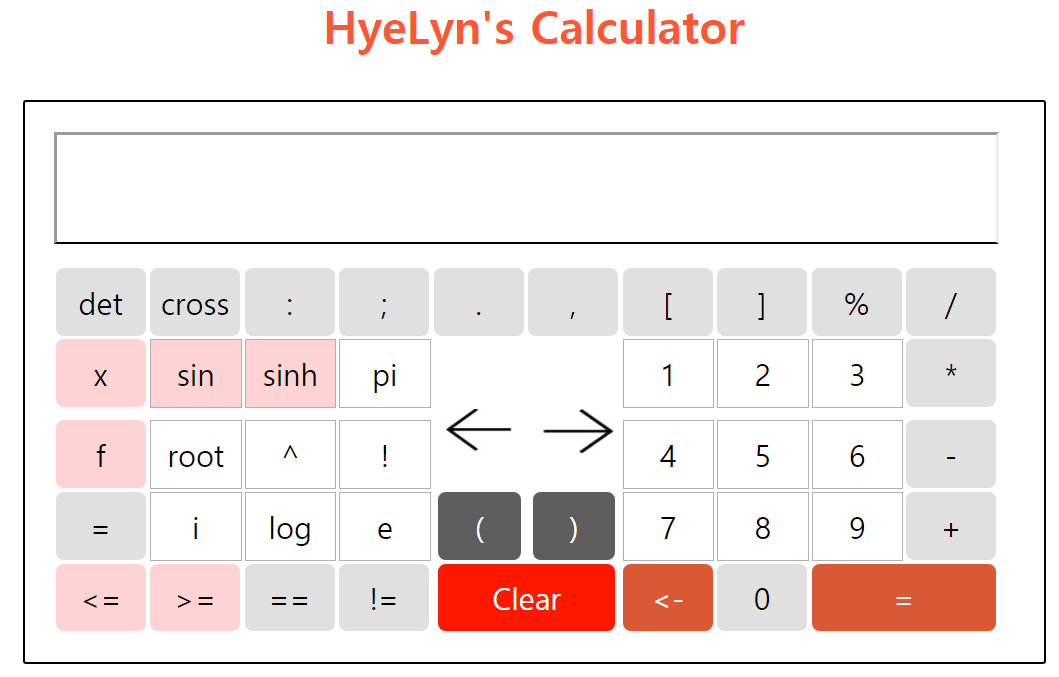
1. 본문

- 기존 계획으로부터 변경된 부분과 그 이유

기존에 계획했던 계산기의 모습은 다음과 같다.



그리고 구현한 계산기의 모습은 다음과 같다.

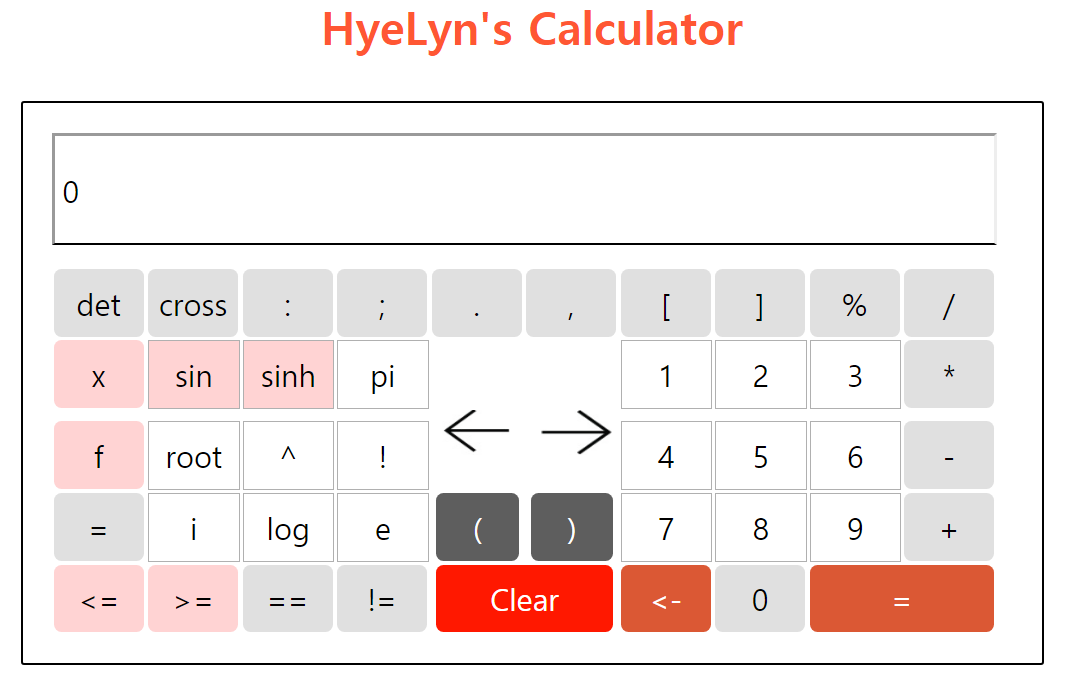


우선 계산기에서 값을 나타내는 방식이 처럼 위 아래로 나누어지는 방식이 아니라 1/2처럼 좌우로만 써지는 방식이기 때문에, 방향키의 위 아래가 필요 없게 되어 뺐다. 이동 해야 할 부분은 좌우밖에 없어서 좌우 방향키는 남겨두었다.

그리고 계획했던 계산기의 모습에서 최소 기능이 몇 개 빠져있어서 빠진 기능들을 위쪽으로 더 넣어주었다. 다행히 위아래 방향키를 빼면서 공간이 좀 더 생겨 딱 맞게 넣을 수 있었다.

Mod는 없애기로 했다. 삼각함수의 역함수는 최소 기능 조건은 아니었고, 또한 간소화가 목표인 계산기에서 자주 쓰지 않는 기능까지 넣어주어 복잡하게 해 놓을 필요는 없다고 생각하였다.

- 사용자 인터페이스의 구성 요소 및 사용 방법



1

2

3

4

1. 계산기 웹 어플리케이션의 인터페이스

전체 계산기의 모습을 직관적으로 보여주고 있습니다.

화면 전환 및 모드 변경 기능은 없습니다.

1. 결과 창

현재 클릭한 값들이 표시되는 부분입니다. 결과값을 계산해주는 = 키와 Clear 외의 다른 키를 입력했을 경우 입력 결과가 계속 결과 창에 갱신 됩니다.

1. 다중 키

배경이 연한 분홍을 띄는 키들입니다. 비슷한 키들이 합쳐진 버전이라고 생각하면 되겠습니다. 이를테면 동그라미 쳐 진 sin의 경우 한 번 누르면 sin이 결과 창에 입력되지만, 두 번 눌렀을 경우 cos, 세 번 눌렀을 경우 tan가 결과 창에 보여지는 것을 볼 수 있습니다. tan상태에서 다시 sin을 클릭하면 tan에서 sin으로 돌아옵니다.

f같은 키들도 모두 같이 방식으로 사용할 수 있습니다. f는 또 눌렀을 경우 g가 됩니다. g에서 다시 클릭했을 때 f로 돌아옵니다.

1. 화살표 방향키

커서 위치를 이동시켜 줍니다. 커서의 기본 위치는 결과의 제일 끝부분으로 설정되어 있습니다. 왼쪽 방향키를 누르면 커서의 위치를 하나 왼쪽으로 옮길 수 있습니다. 오른쪽 방향키도 마찬가지로 작동합니다. 키가 더 입력되거나 아니면 지워졌을 때, 커서는 다시 결과의 제일 끝 부분으로 돌아갑니다.

1. 백스페이스

커서의 위치에서 입력 값을 하나만 지웁니다. 원하는 부분으로 찾아가서 지울 수 있습니다.

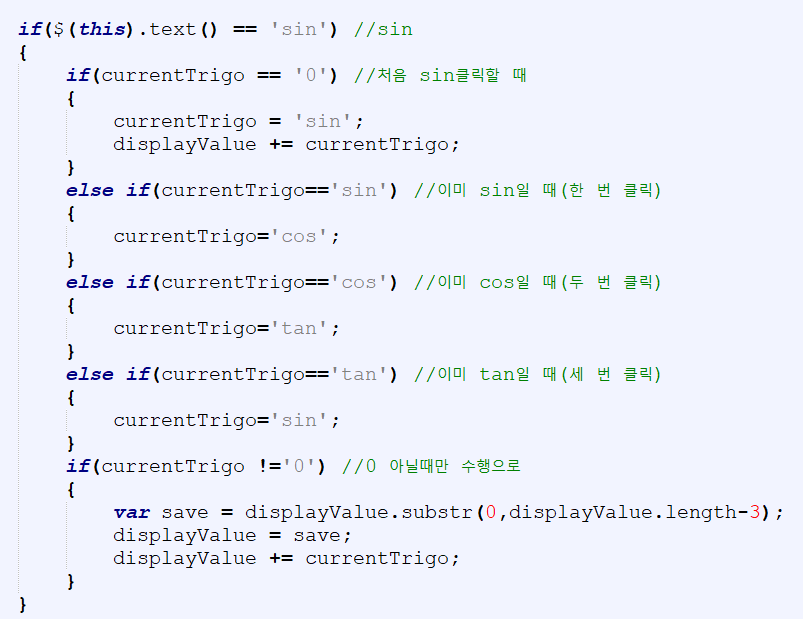
※) 사용방법

계산하고 싶은 수식을 차례대로 입력하여 원하는 수식을 만들고, = 버튼을 클릭하여 값을 구합

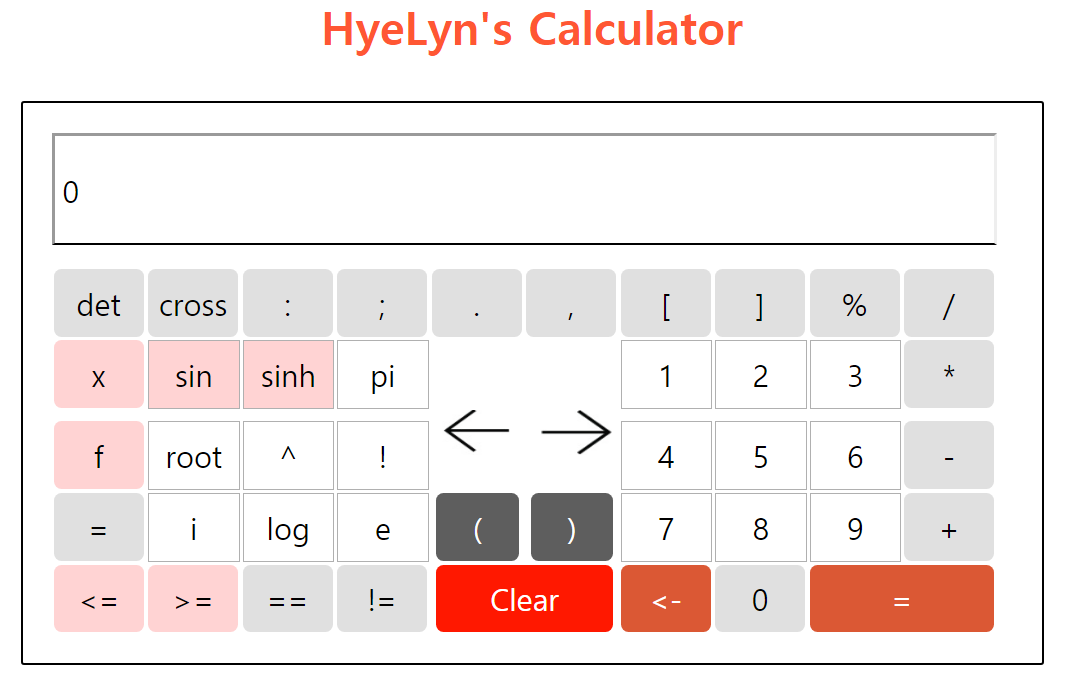
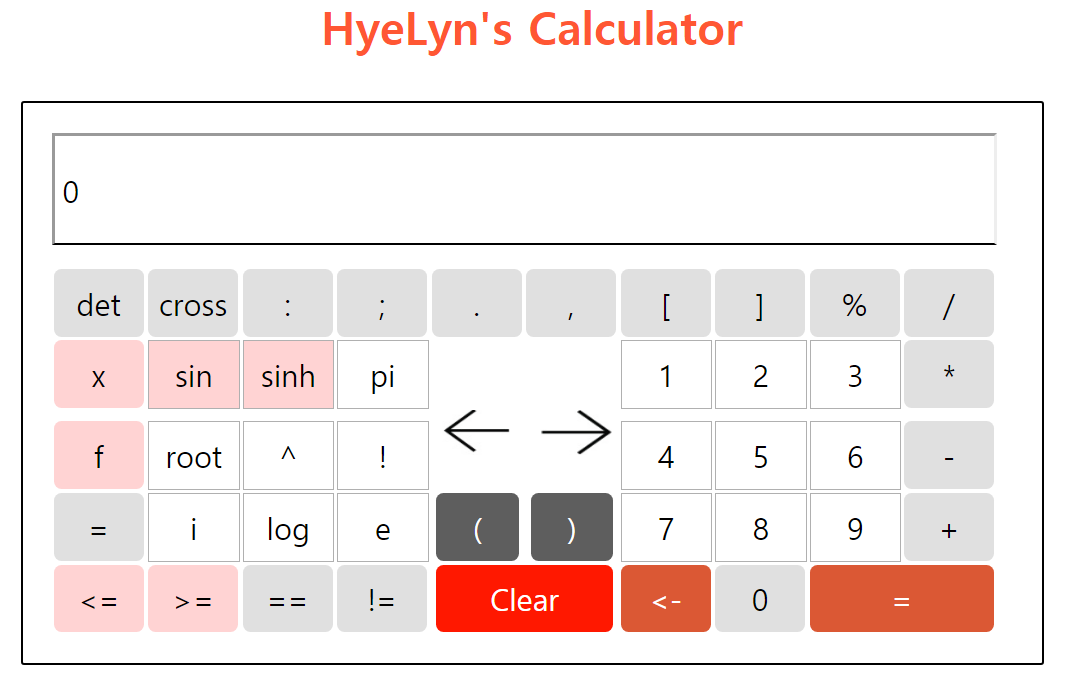
니다. 이 때, 사이즈가 크고 배경색이 진한 = 버튼이 결과를 계산하는 버튼이고 좌측 아래쪽에 있는 =는 할당을 의미합니다. 이를 테면 x=5의 수식을 만들고 싶을 때 사용합니다.

- 특징적인 상호작용 방식들에 대한 세부 구현 방법

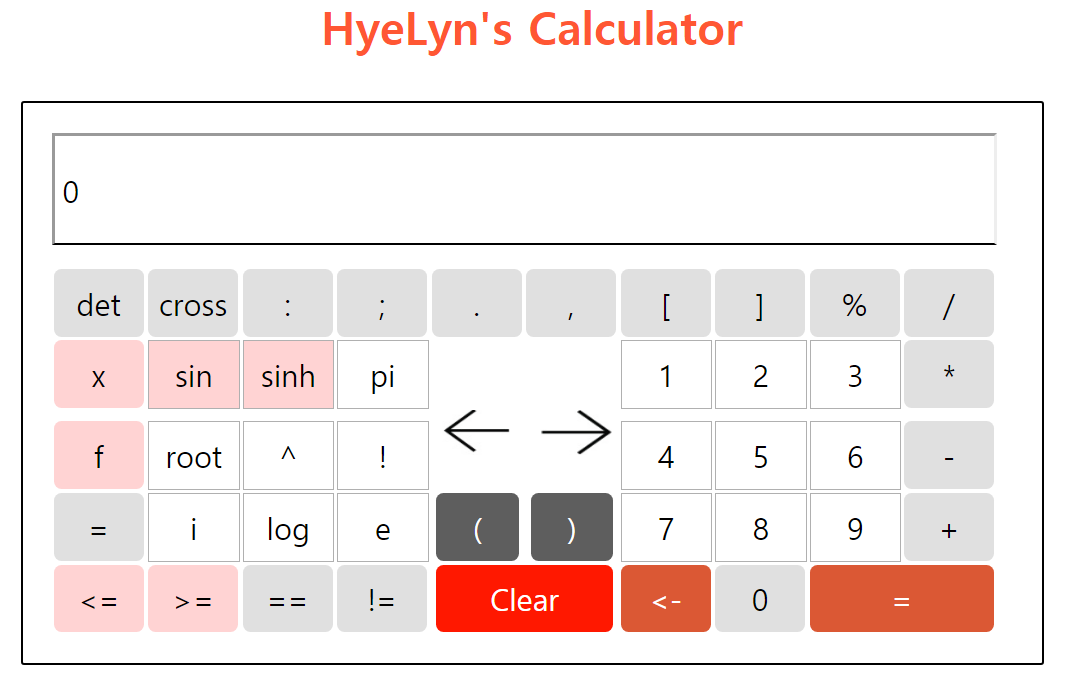
1) 삼각함수



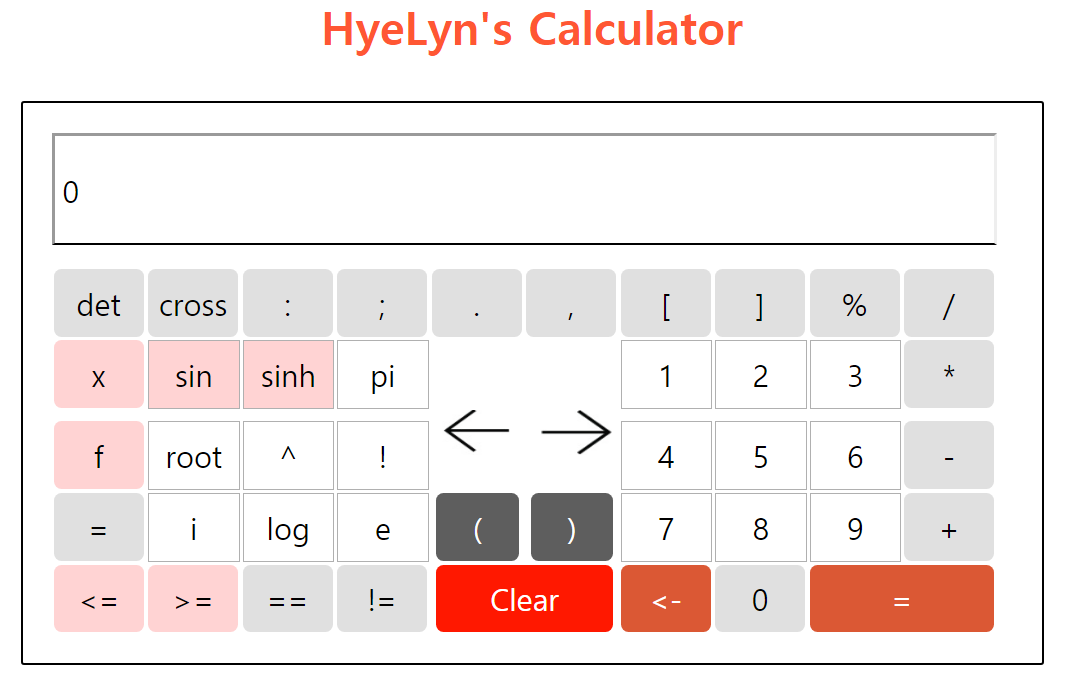
Sin, cos, tan 버튼을 하나로 합친 인터페이스의 구현 부분입니다. 현재 입력된 삼각함수를 기억하고 있다가 또 클릭했을 시에 다음 삼각함수로 바꿔주면서 누를 때마다 sin, cos, tan가 차례로 나오도록 하였습니다. 삼각함수는 길이가 모두 3이므로 다시 눌렸을 때 원래 있던 3개의 문자열을 지우고 새로운 삼각함수가 붙여지도록 구현하였습니다.



sin

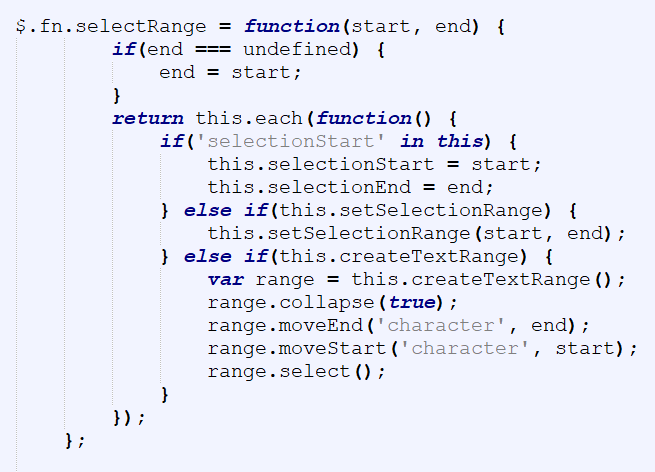


cos

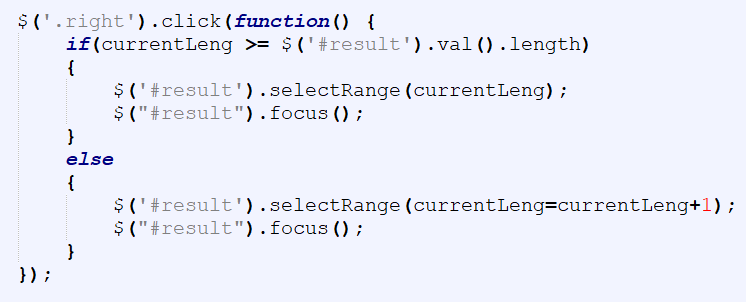


tan

2) 방향키의 커서

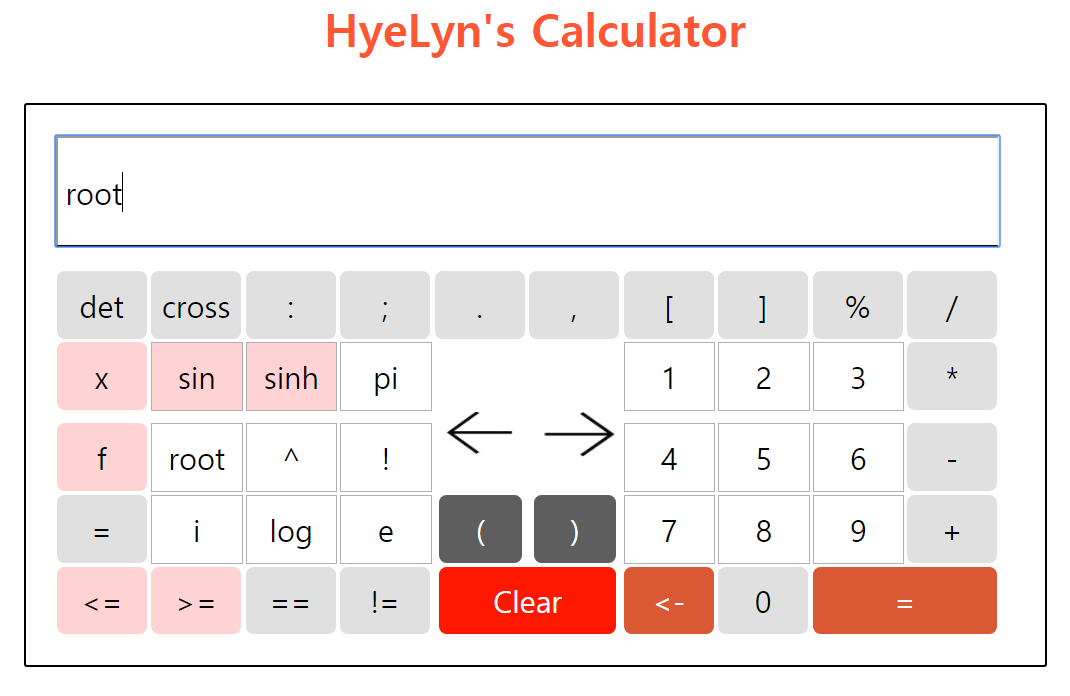
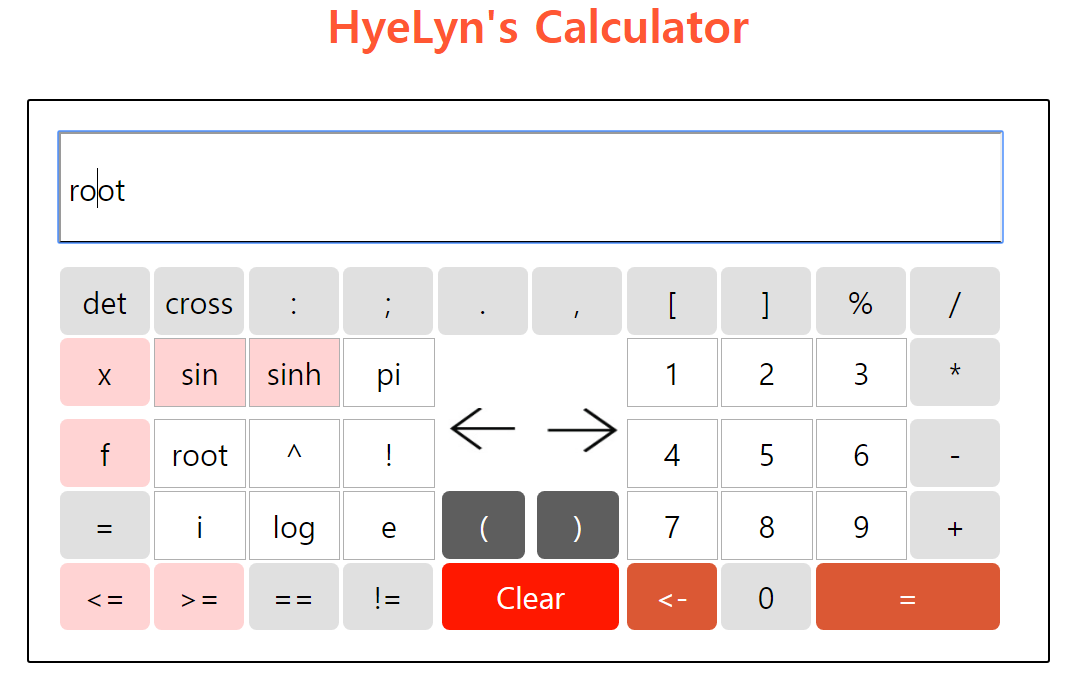
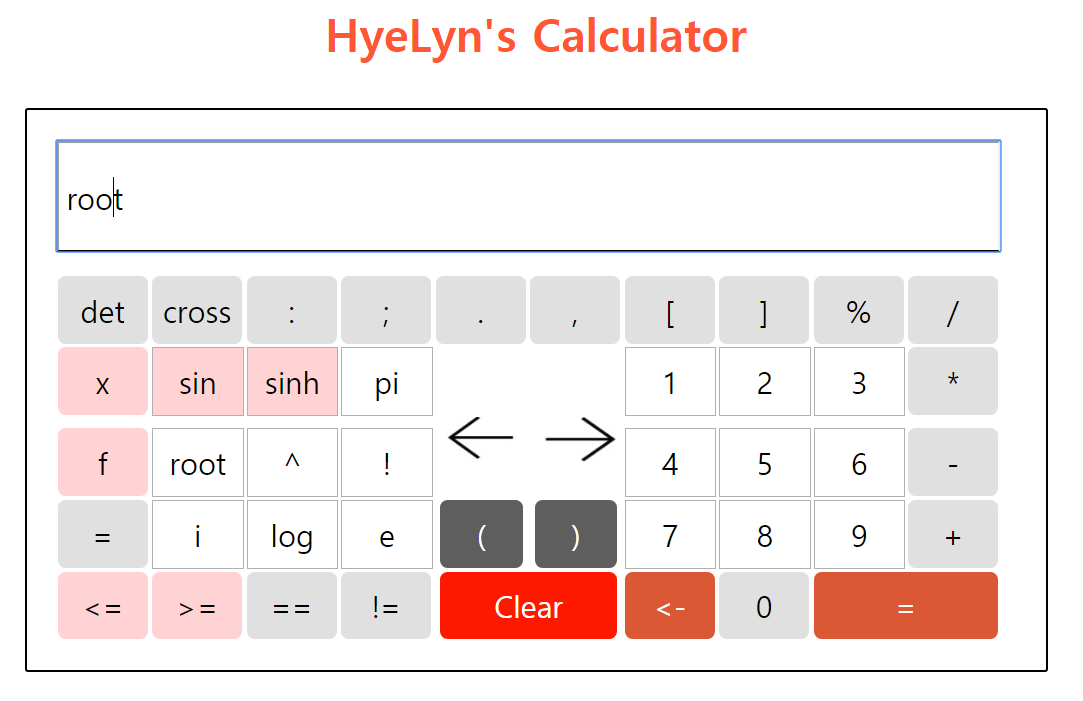
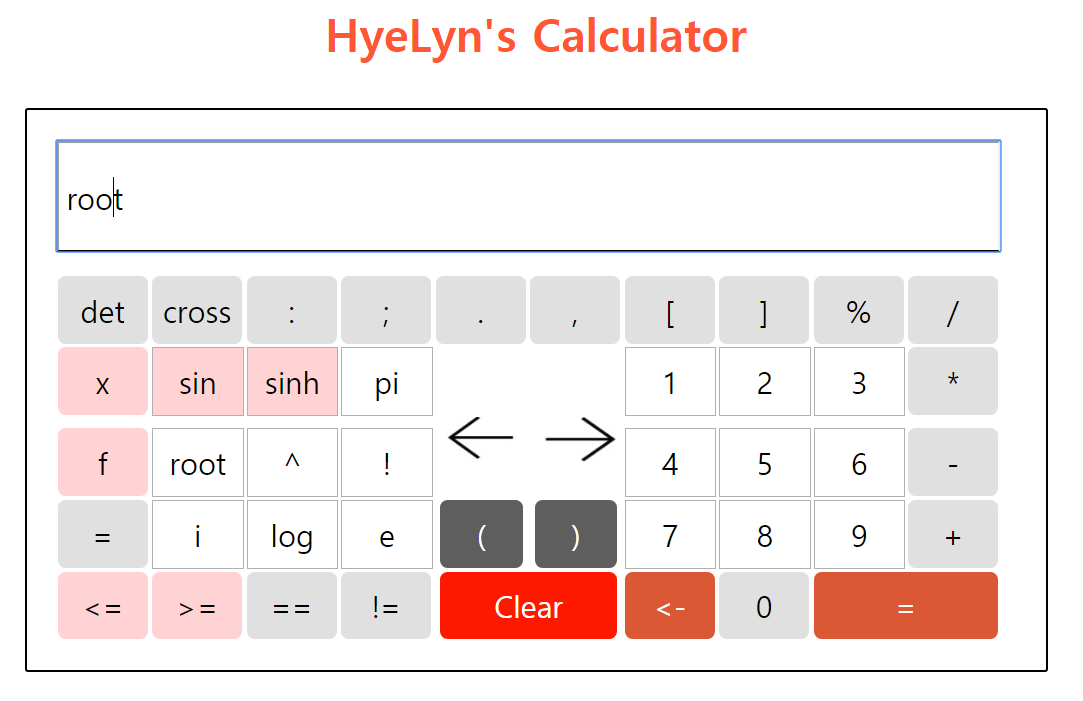


결과 창에서 커서의 범위를 선택할 수 있는 함수 구현 내용입니다.

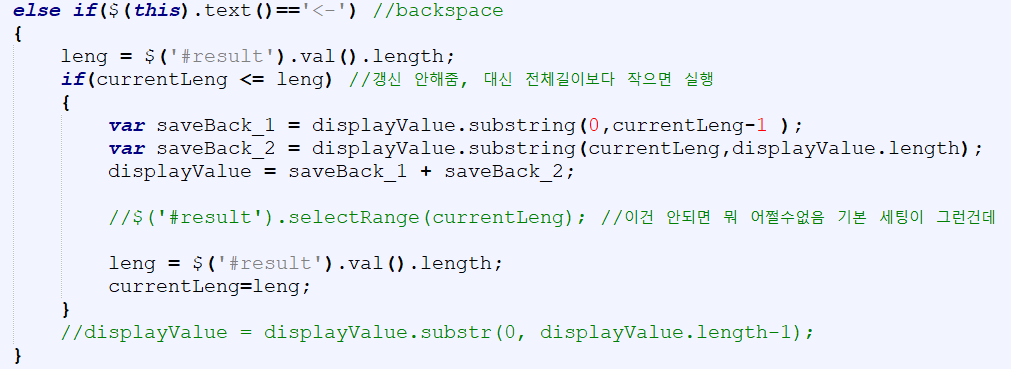


방향키를 클릭해서 이벤트가 발생했을 때 구현 내용입니다. 결과 창에서의 커서의 위치를 오른쪽으로 옮겨줍니다. 이 때, 이미 제일 오른쪽에 커서가 위치해있다면 그 자리에서 다시 포커스를 줍니다.

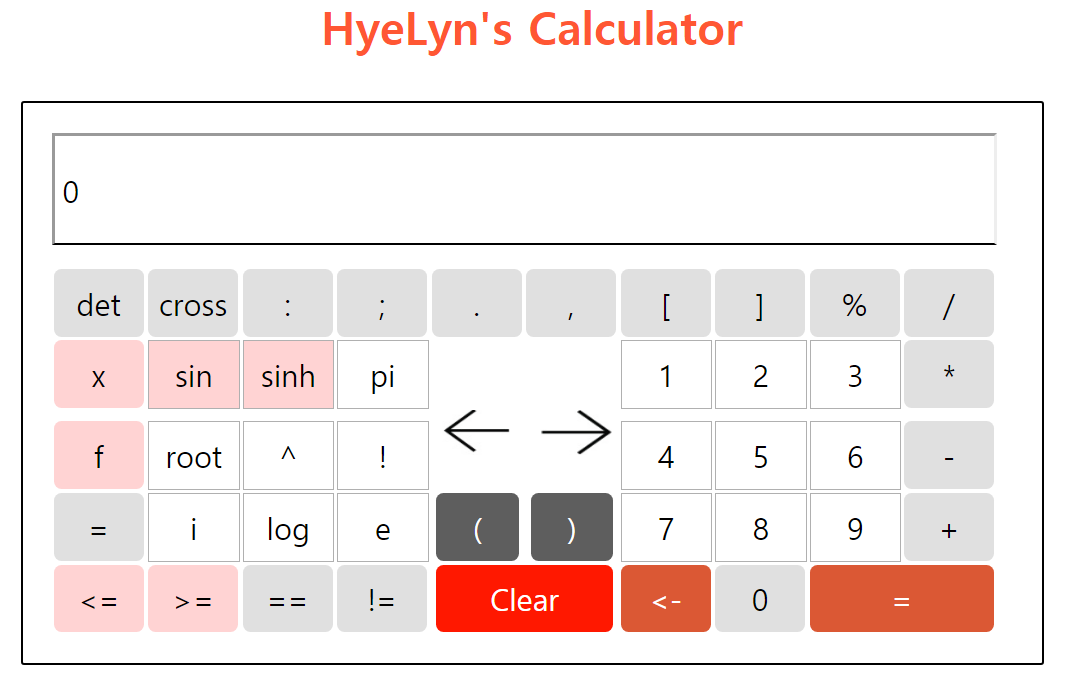
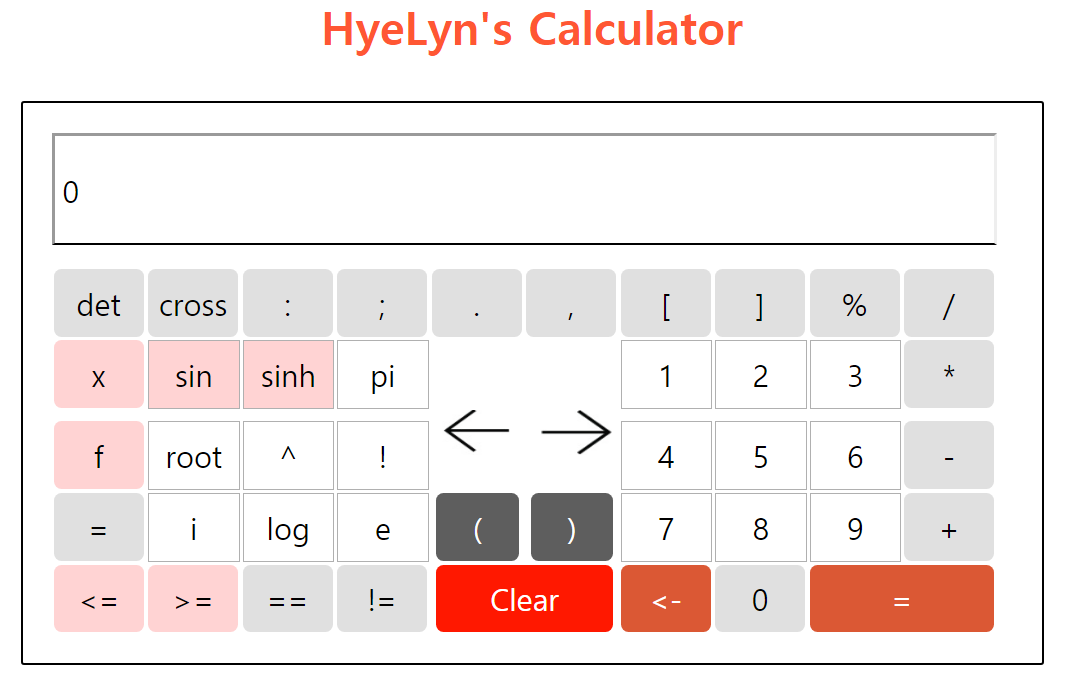
왼쪽의 경우에도 마찬가지로 커서가 제일 왼쪽에 있을 때를 제외하곤 같은 방법으로 커서를 왼쪽으로 이동시켜줍니다.



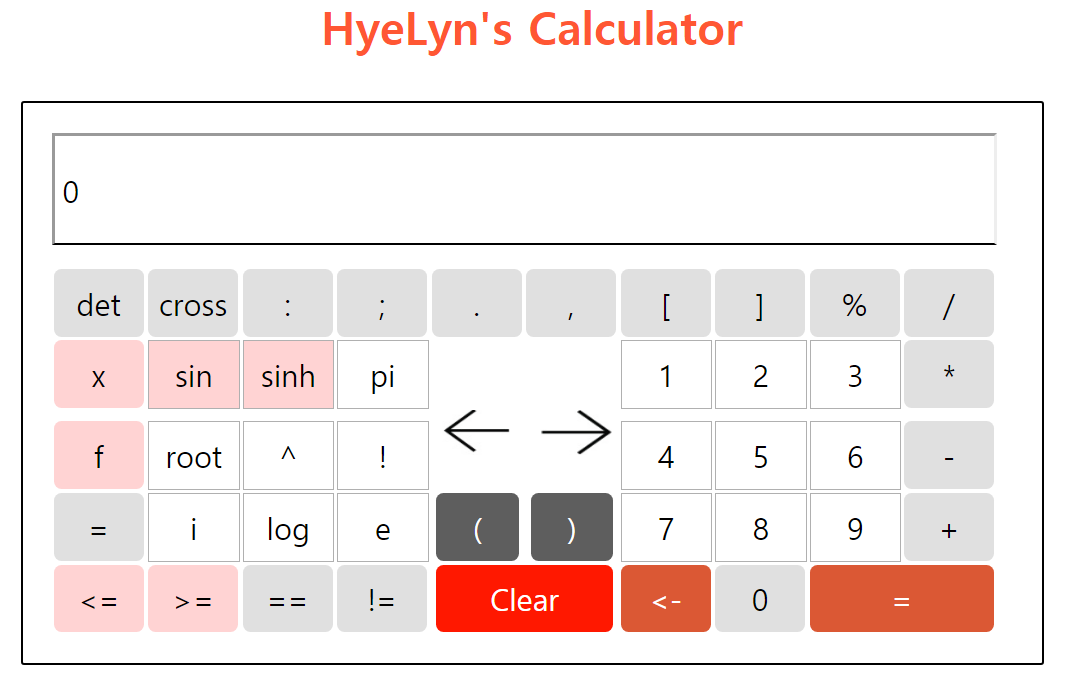
3) 백스페이스



백스페이스 키가 눌렸을 때의 구현 내용입니다. 전체길이를 결과 창에 있는 값의 길이로 받아주고, 그 길이보다 현재 커서 위치가 더 작을 때만 다음과 같은 과정을 실행합니다. 현재 displayValue에 있는 값을 0부터 커서 위치 전까지, 그리고 커서 위치에서부터 끝까지로 나누어 저장해주고 다시 합쳐줍니다.

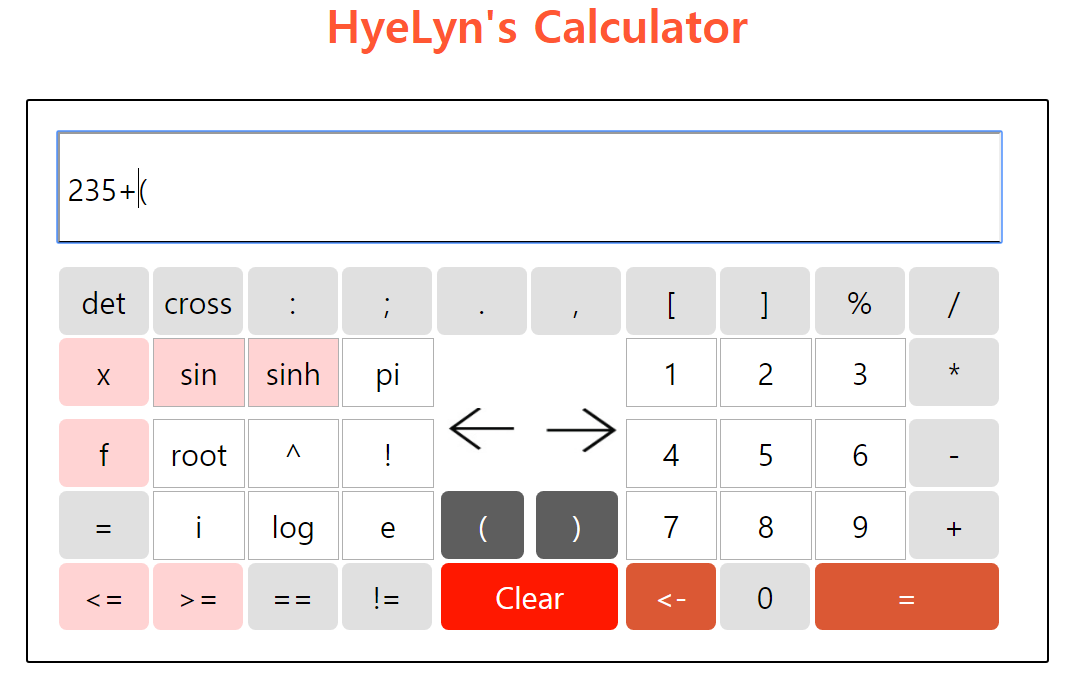


235+()



235(

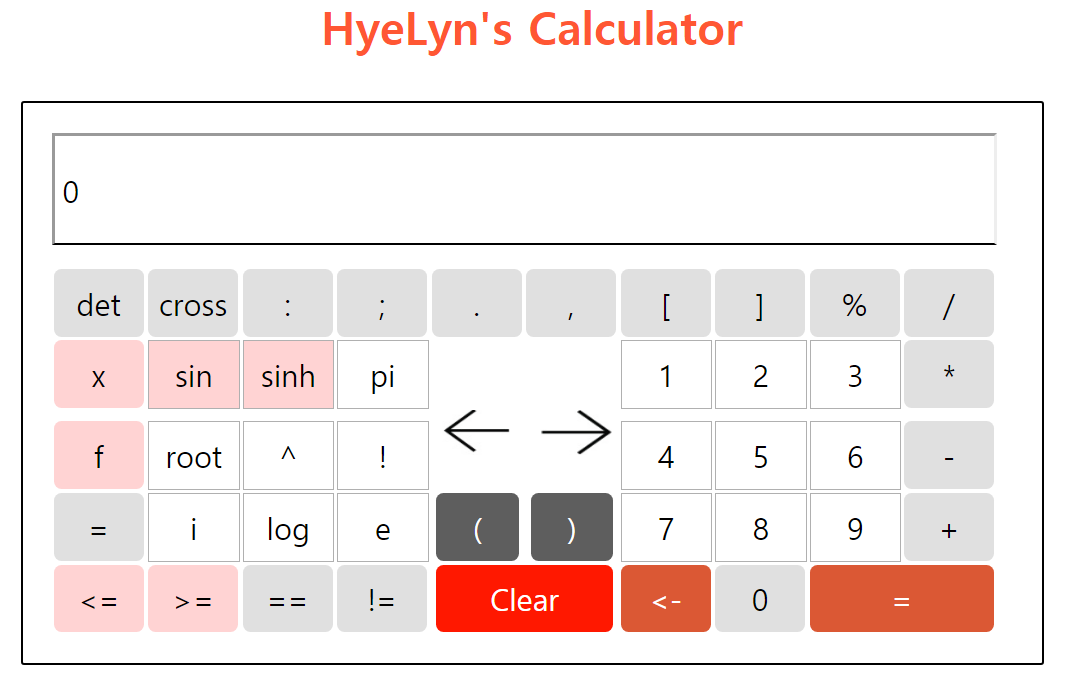
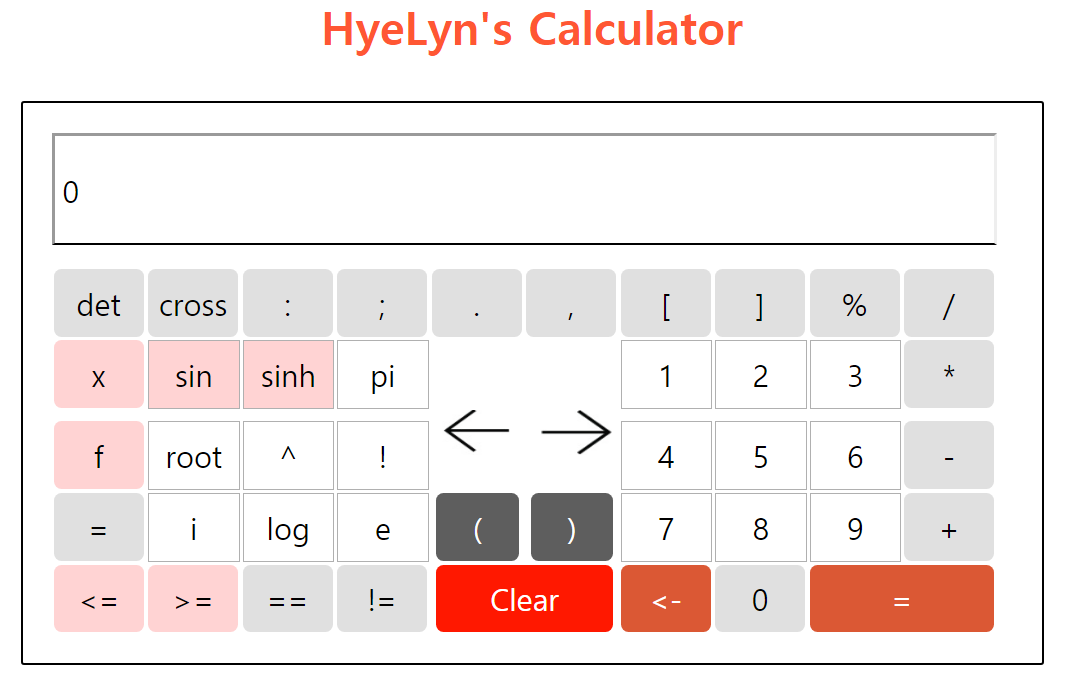
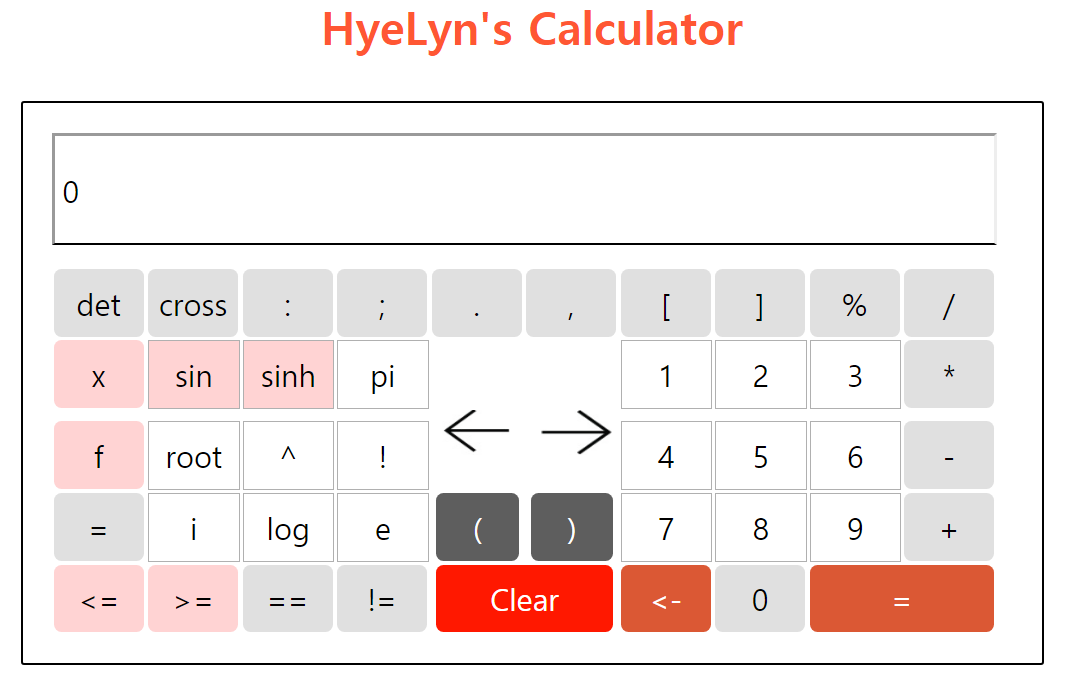
235+(



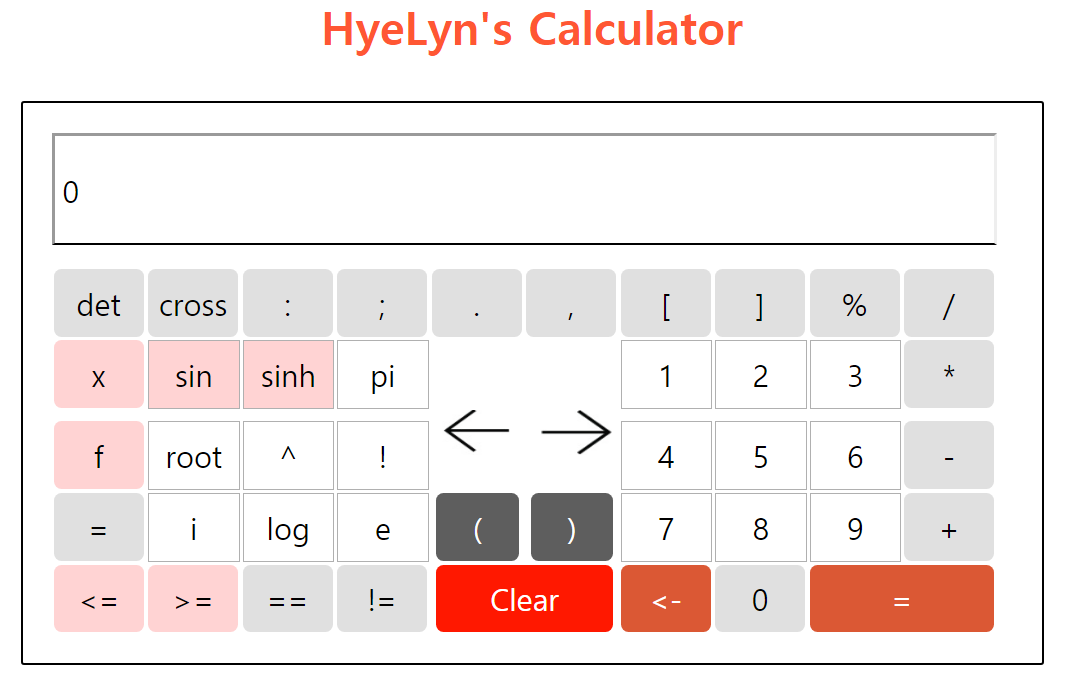
- 실제 문제에 대한 사용 예시

<문제 해결 과정 스토리보드>

1) x=20, y=10일 때, 2sin(x)+(4+e^2)+2cos(y)의 값은?



x



x=20

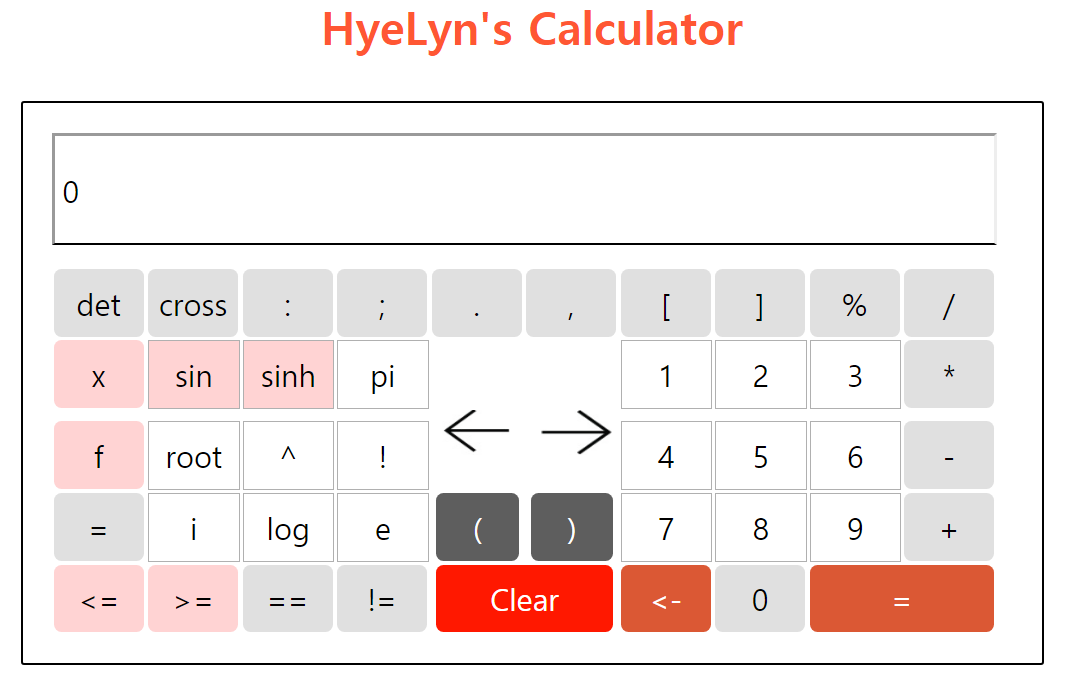
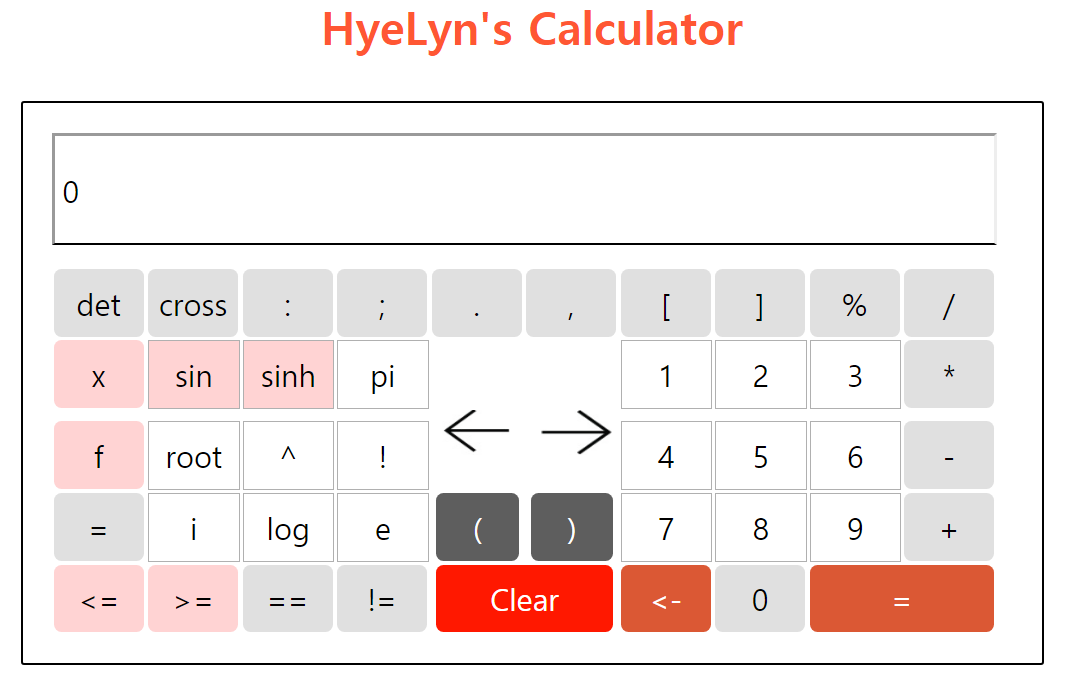
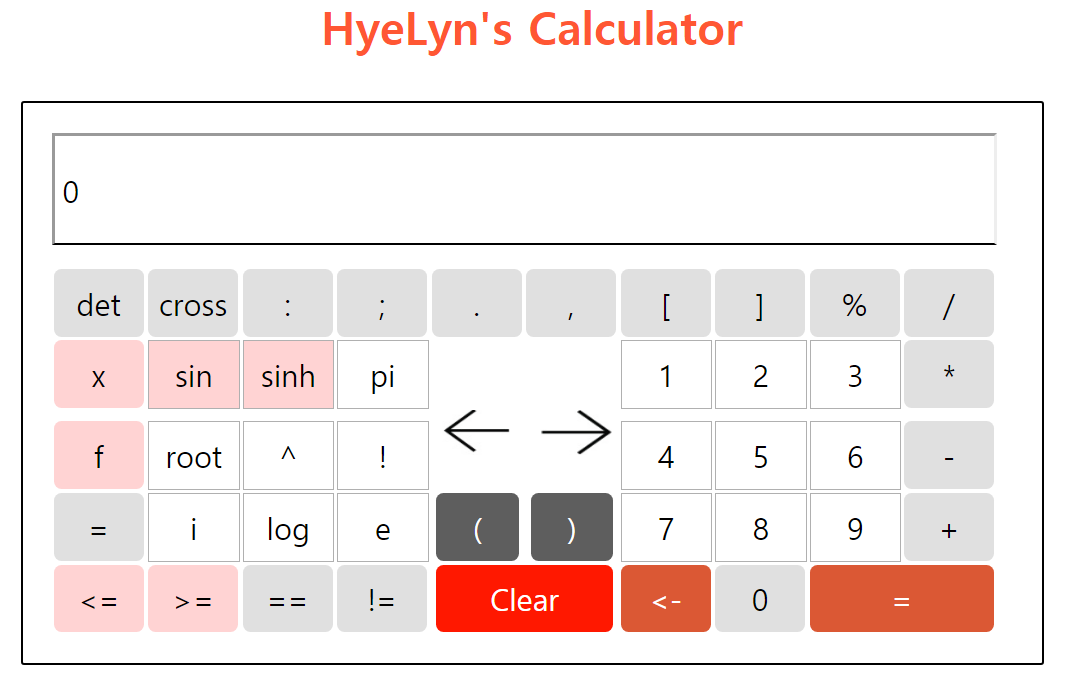
x=



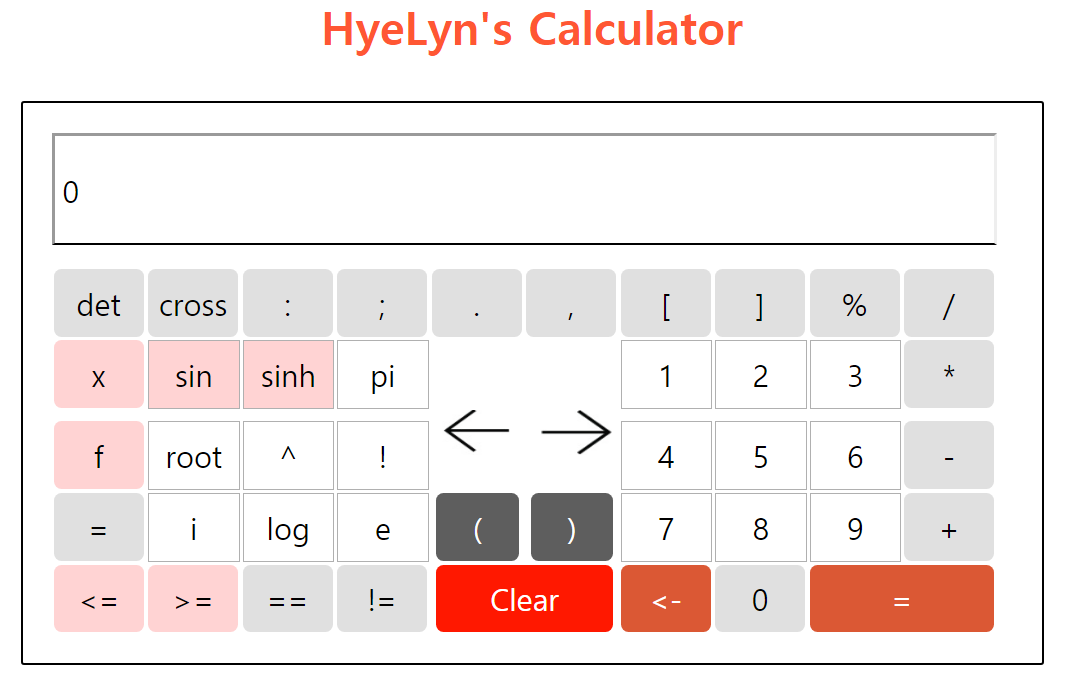
x=2



X=20을 먼저 할당해준다, 마찬가지의 방법으로 y=10도 설정



2



2cos(x

2cos



2cos(



X2

X2

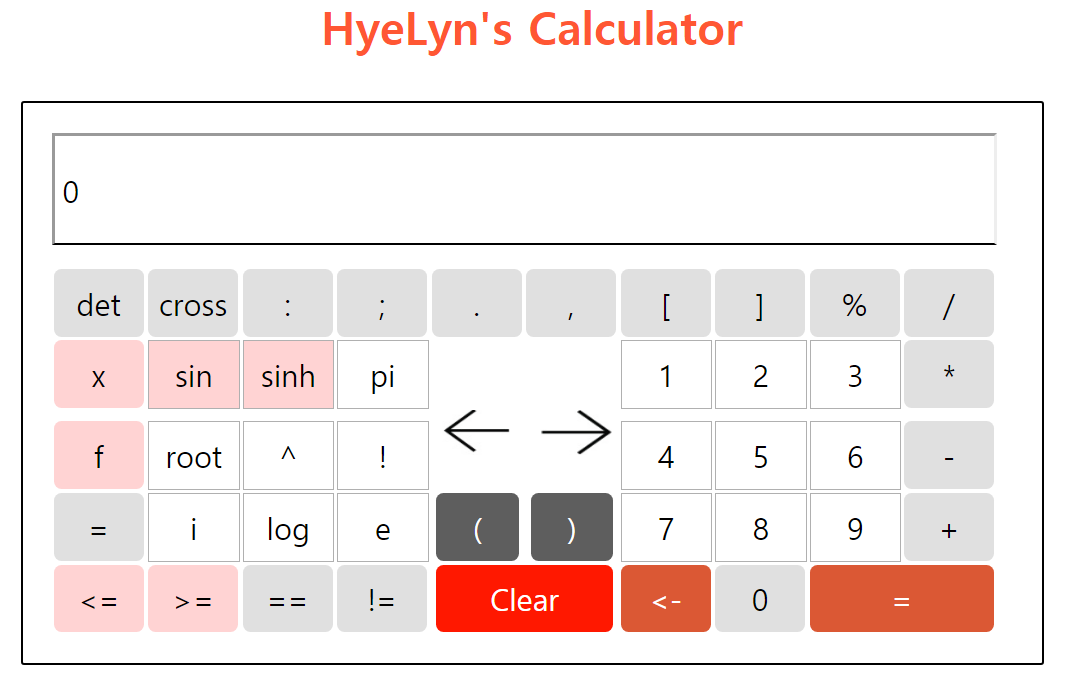
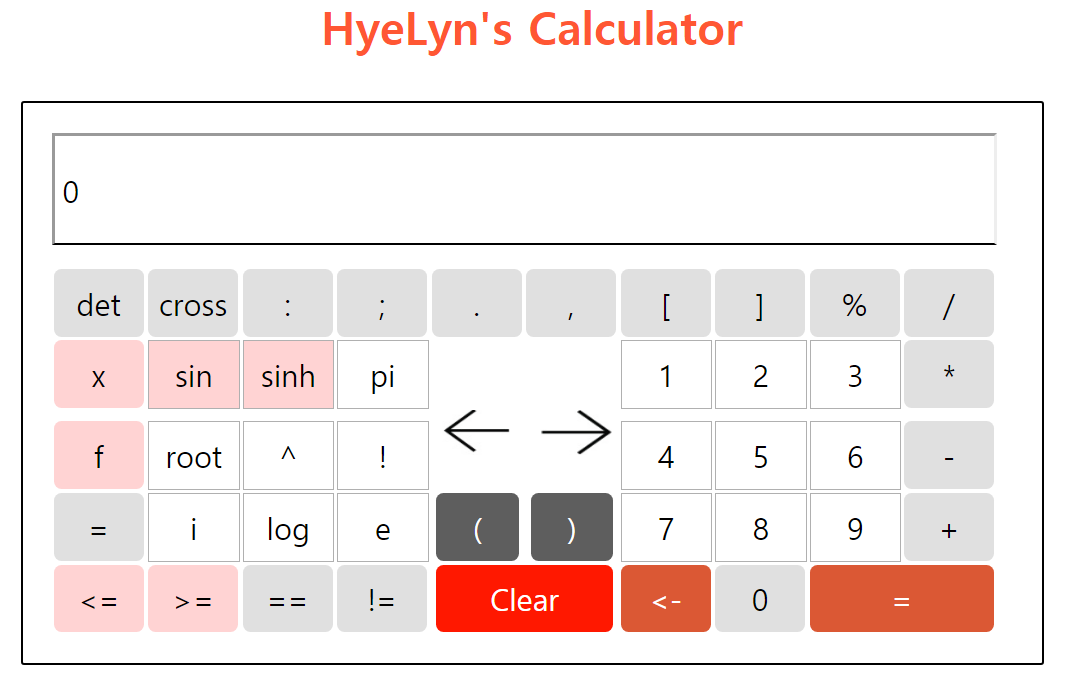
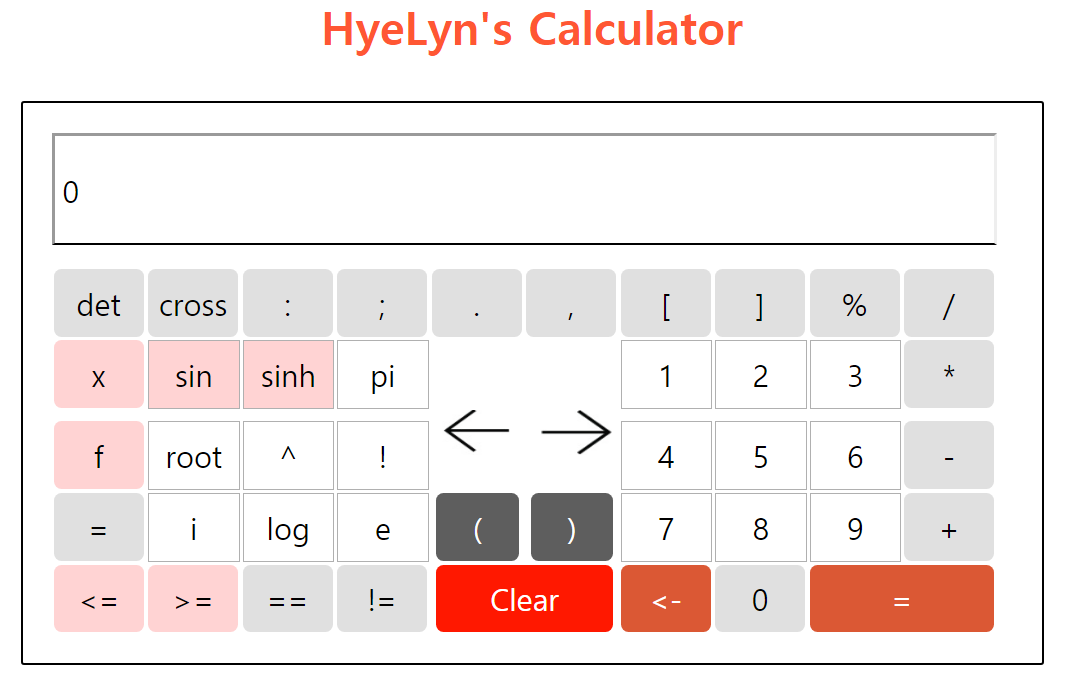
2cos(y)를 만드는 과정이다. Sin은 해당 작업을 다 한 번 씩만 눌러서 해주면 된다.

따라서 처음에 x=20, y=10을 할당해주고 2sin(x)+(4+e^2)+2cos(y) 식의 값을 구했다.

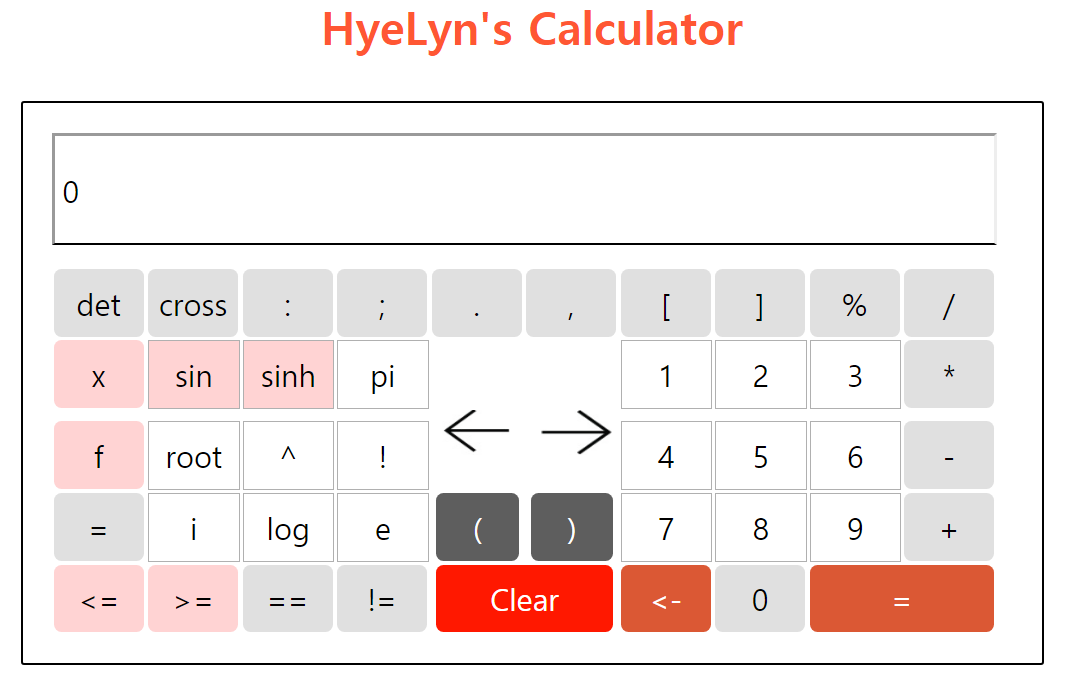
2cos(y)를 만드는 방법은 위와 같고 나머지는 있는 그대로 클릭해주면 된다.

답 : 11.536803542232999

2) 16진수 915D2를 10진법으로 변환하라.



16



16^4\*

16^



16^4



16^4\*9+ … 를 수행해주면 변환할 수 있다. 있는 그대로 클릭해주면 완성된다. 가끔 오타가 났을 경우 커서를 이동해서 수정해주거나 백스페이스로 지웠다.

답 : 595410

3) {5+(6-11i)}+(-5+4i)

있는 그대로 클릭해주면 완성된다. 보이는 숫자 그대로 입력하면 되기 때문에 스토리보드는 생략한다. 마찬가지로 오타가 났을 경우 이동해줄 수 있다.

답 : 6-7i

1. 논의

- 구현 측면에서 성공적인 부분과 실패한 부분

우선 인터페이스의 구현이 성공적이었다. 몇 개의 버튼이 추가되거나 사라지긴 했지만, 원래 생각했던 인터페이스대로 전체적인 모양 구현이 가능했다. 스타일 구현도 거의 차이 없이 잘 했다.

그리고 기능적으로는 삼각함수처럼 버튼을 합치려는 구현들은 모두 성공적이었다. 방향키도 좌우 커서 움직임이 자유로워서 좋고, 백스페이스도 잘 작동한다.

하지만 아쉬운 부분은, 커서의 위치가 좀 더 융통성 있으면 좋았겠다는 것이다. 현재 구현 방식은 이렇다. Sin(2)에서 2의 위치로 커서를 움직여 2를 지웠다. 그러면 괄호사이에 커서의 위치가 남아있는 것이 아니라 다시 맨 끝으로 이동한다. 구현할 때 매번 현재 커서 위치를 전체 길이로 초기화해주기 때문이다. 계속 커서의 위치를 기억해서 구현했으면 좋았을 것 같다.

- 사용성 측면에서 긍정적인 측면과 부정적인 측면

기존의 계산기 인터페이스보다 사용성을 높이기 위해 위치를 많이 바꾸었다. 방향키를 중심으로 오른쪽은 숫자와 기본연산들을, 그리고 왼쪽으로는 공학용으로 자주 사용하는 함수들을 배치하였다. 그래서 기존의 계산기보다는 인터페이스가 직관적이어서 사용성에 도움이 되었다고 생각한다.

그리고 삼각함수나 버튼을 합칠 수 있는 함수들은 합쳐서 간소화시키고 그 기능을 사용자가 알아 볼 수 있도록 배경색을 조금 다르게 하였다.

백스페이스와 방향키도 마찬가지로 원하는 위치에서 사용자가 입력할 수 있게 해줌으로써 사용성 측면에서 더 좋아진 것이라고 생각한다. 거의 구현이 잘 된 부분이 사용성을 향상시키고 있는 것 같다.

때문에 백스페이스의 위치가 융통성이 없는 부분이 역시 아쉬운 점이 되는 것 같다. 이 부분은 부정적인 측면이라고 생각한다.   
 또한 결과값을 구해주는 evaluate 키도 = 로 표시하고 할당해주는 키도 =로 표시하여 사용자가 헷갈릴 수 있는 부분도 생각해봐야 할 듯하다. 물론 사이즈와 색상을 다르게 하여 구분을 주려고 노력했지만, 실제로 시연하는 과정에서 여러 번 왔다갔다 했기 때문이다.

- 과제 #2에 대한 전반적인 자체 평가 및 향후 개선 계획

인터페이스 및 기능적인 부분에 대해서 생각하는대로 대부분의 구현을 했고 디자인도 마음에 들기 때문에 굉장히 만족한다. 10점 만점에 8점정 도를 주고 싶다.

하지만 간소화시킨만큼 다른 공학용계산기보다 기능이 많이 빠져있어서 그 부분을 어떻게 해결할 것인지를 생각해야 할 것 같고, 커서의 위치를 보다 융통성 있게 만들어주어야 좀 더 편할 것이다.

<유튜브 링크> <https://youtu.be/EYs9PZKzecA>