**Input /Output:-**

Input: Numeric values (combination of 0-9). 11 digit display.

‘+, -,\*, /, =’ operations allowed.

Delete and decimal point functionality allowed.

Output: Numeric values as result of the numeric input and operation selected. 11 digit display.

**Errors:** 1.First input should be a number then operation. If not, then it will display ‘ERROR’ message.

2. For first number and then “=” operation, it will give the result as the number.

3. If the input is first number then operation and then again operation, it will replace the

previous operation with the latest one.

4. Any other operation put other than mentioned above will generate ERROR as output.

5. If a number is divided by zero then it will display ‘cannot divide by zero’.

6. For input N, if it is not combination of number or invalid input like ‘9.3+…’, then it will give Error.

**Finite State Machine for Simple Calculator:-**

0…99

+,-,/,\*,.

0…99

INVALID

=

INVALID

INVALID

Error

=

0…99

Result

Fig (1)

**Finite State Machine:-**

**INPUT 1:**

If one digit or combination of digits are succeeded by ‘equals to’ then it will be the number itself.

For valid combination of 0 to 9 digits, it will go to operation state or else it will be invalid and output will be ERROR.

If the numbers entered for calculation are incorrect or need to be changed, then delete key will be used.

If first input entered is not a valid numeric value, then it will generate ERROR.

**OPERATION:**

If the operation is other than ‘+,-, /,\*,., =’ then it will be invalid and output will be Error.

For Valid operation, it will wait for the next Input.

If a user doesn’t want to do the calculation after entering the operation, then user can click clear to reset.

If there is sequence of digits with multiple operations like 1+2\*3, then the calculation will be in loop between INPUT N and OPERATION till the time the expression is completed validly (proper sequence of digits and operations) or else it will give an ERROR.

User also can delete a number out of the two operands to rectify any one of them

**INPUT N:**

If user enters a valid numeric expression to be calculated, then it will display the RESULT.

If there is a sequence of digits with multiple operations, then it will be as mentioned under OPERATION above.

For non numeric input, ERROR will be displayed.

**ERROR:**

ERROR will be displayed for any invalid input, or invalid expression.

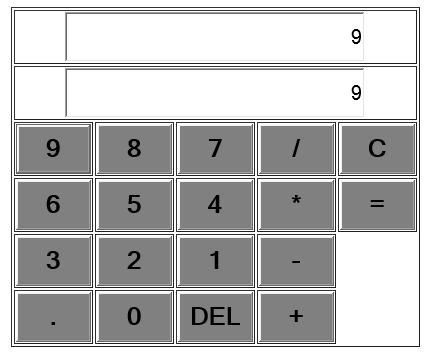
**RESULT:**

The RESULT will be the final output of the calculation.

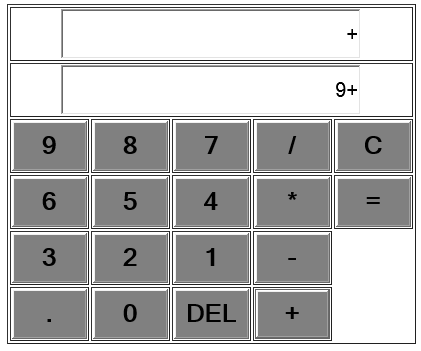
**Implementation Details:-**

Below screenshots show the example of add operation performed :-

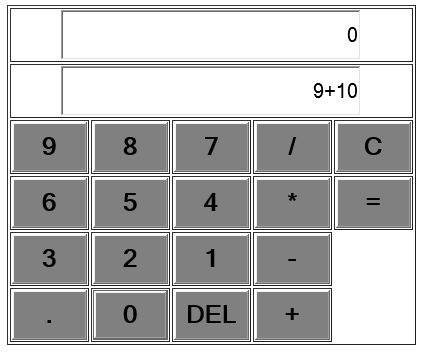
**1)** First operand is typed.



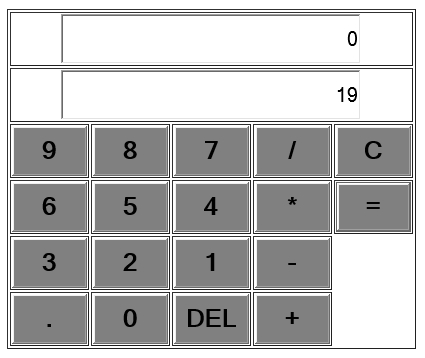
**2)** Select '+' operator for add operation



**3)** Select the next operand to be added



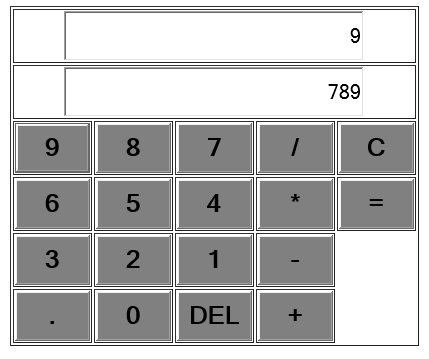
**4)** Final result of the Operation



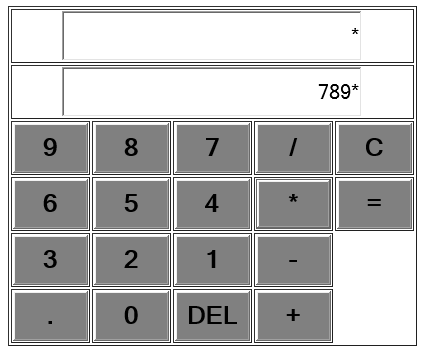
Similarly, multiplication, subtraction and division.

**Delete operation:-**

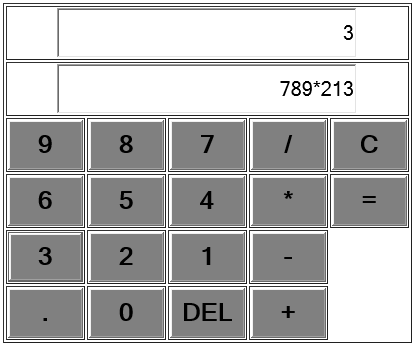
**1)** First operand entered for addition



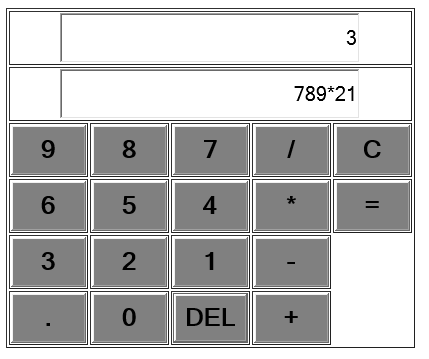
**2)** Operation selected "\*"



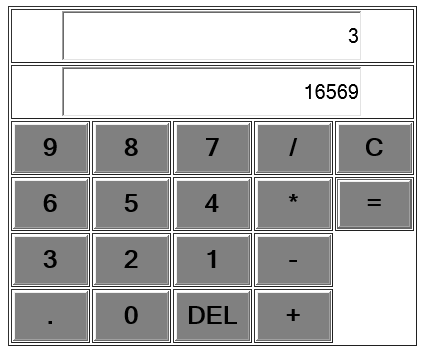
**3)** Second Operand entered but entered wrong i.e. instead of 21, it was entered 213.



**4)**Rectified the mistake by delete last entered digit.

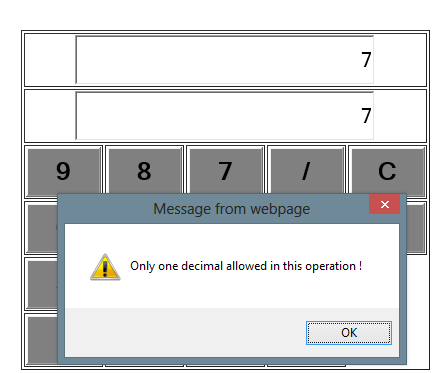


**5)** Final result after rectification

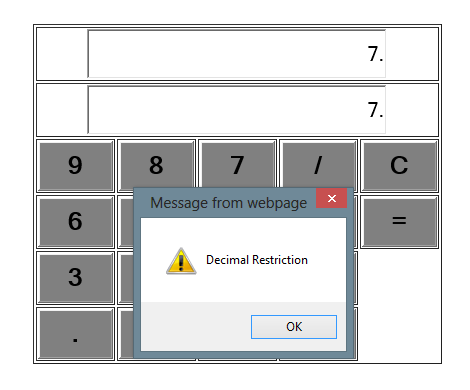


**Some Errors and Alerts tested :-**

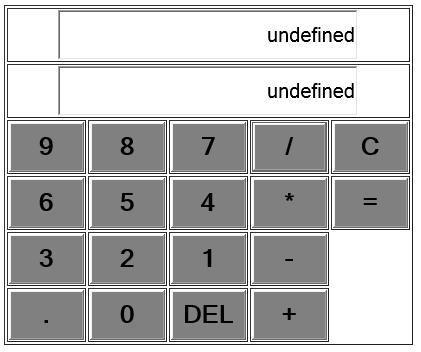
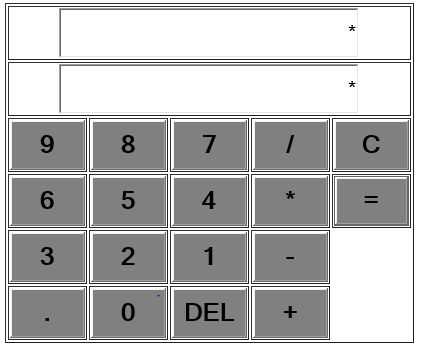
**1)** Alert generated on first use of decimal



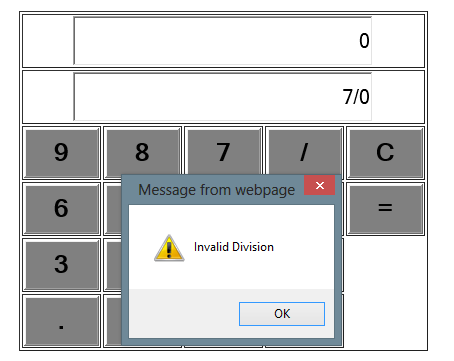
**2)**Error of decimal restriction generated if decimal used second time in same operation.



First is only operator without operand like multiply and any other operation will cause undefined error :-



If number is divided by 0 :-



Similarly, for other invalid operations and calculations, error messages are generated.