SKIT Hackathon 2019

Problem Statement:-

“Expression evaluator for current generation trader”

As we know Indian stock market is the best place to invest. Indian trading ratio is increased tremendously recent days where traders transiting from normal to advance trading techniques which opens the space to create more advance trading tools. Every trader have their personal preference and formula in their mind. Systems today allow the trader to create their own formulae, but it’s easy to mess up or make mistakes. To handle this challenge we need mechanisms that help user to create their formulae or expressions, in an easy way.

This challenge, is to create an HTML5 based Expression Editor, with AutoComplete, and Expression Validation, Parentheses matching, etc.

We will give N lists - like Instrument List, Position List, etc. Each list item is in turn a list - with first item of list being primary field name and other (optional) list entries being aliases for that field name. As user starts typing some field name, your code should search in these lists, and show fields that are relevant matches in Instrument, and which are the matches in Position. And if user selects specific field from Instrument, it should replace the field typed by user, with Ins.PrimaryFieldName.

Sample List,

Instrument List		**Position List**	
Main Field	**Optional Fields**	**Main Field**	**Optional Fields**
LTP	Last, Last Traded Price	PNL	P&L, Profit & Loss
Volume	Vol	Day PNL	MtoM, Day Gain
Yr Hight	Year High, 52wk high	Realised PNL	Booked Profit, Booked PNL
Yr Low	Year Low, 52wk low	BEP	Break Even Price

So if user is typing Last it should show

Instrument
    LastTradedPrice LTP
    LastClose
Position
    LastDividendReceived

If user selects LastTradedPrice, it should replace what user typed, with Ins.LTP

The list can have multiple aliases for each field, but whatever is typed, it is replaced by the first FieldName.

Supported operations are:
+, -, /, *, ^

Supported functions sample list:
log(), exp(), log10(), exp10(), sin(), cos(), tan(), min(...), max(...), avg(...), etc.
It should work in both mouse/keyboard as well as Touch based system.

AutoComplete should appear near or below where the field is being typed.

Expression should be evaluated for full proper valid expression.

If there are any expressions that aren't correct - then they should be either highlighted (acceptable), or corrected (desirable).

Implementation will be evaluated on following points -

- smoothness of user experience
- Code quality
- Feature coverage
- Demonstration of error handling
- Presentation
- Ease and grace of approach - for this teams will be evaluated on continuous basis, not just at the end.