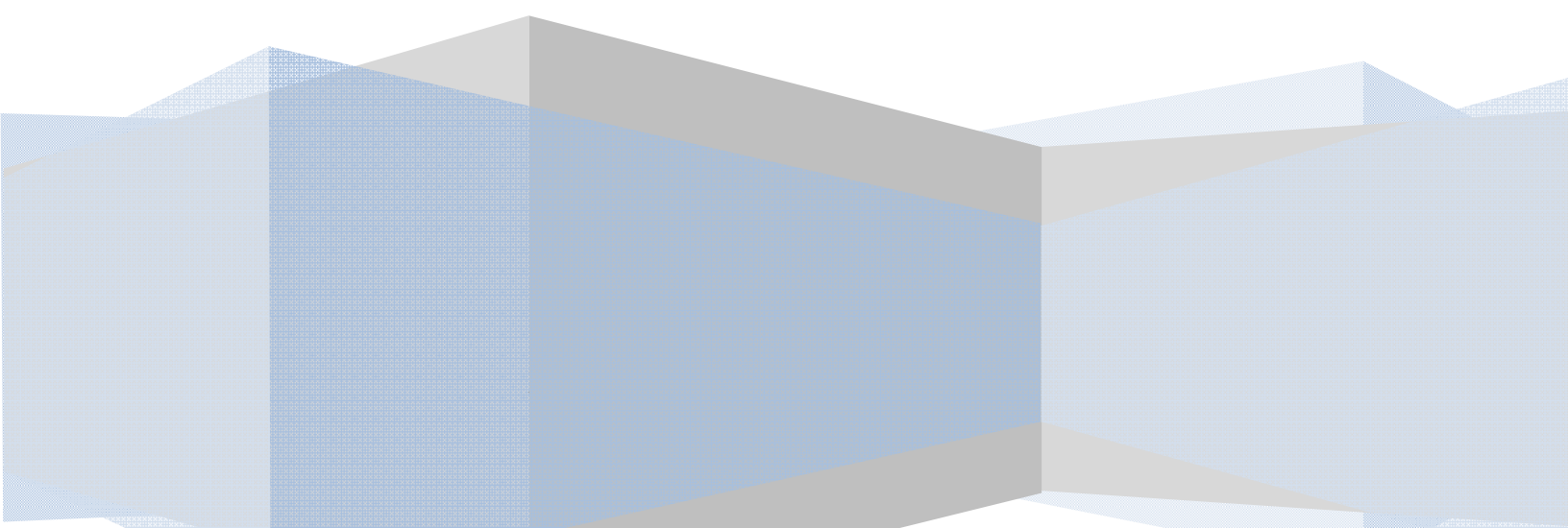


www.iGuideStocks.com

IGS SQL



How to write IGS SQL:

iGuideStocks has its own proprietary robust and powerful Query Engine which will allow you to use existing or create your own filters to get any filtered scan with custom output fields at lightning speed.

The iGuideStocks tool has provided the full query support called as IGS SQL. The IGS SQL looks like SQL but it is different than SQL. IGS SQL runs only on iGuideStocks and does not require any database installation etc. The IGS SQL is very powerful and you can use any function, data variable or customized indicator as you use them in IGS Formula Language.

The structure of IGS SQL looks like this

Select <column names separated by comma> from table where <condition statement> sortby <expression>

Example

Select open[0], close[0] from table where open[0] > close[0]

This will return all rows from selected category where current opening price is greater than closing price.

Example

Select open[0], close[0] from table where open[0] > close[0] sortby close[0]

This will return all rows from selected category where current opening price is greater than closing price and then it will sort data using closing price.

Note: All indicators used in sortby column must be present in select statement .

You can write any expression in sortby column to sort your data for display purpose.

Advanced IGS SQL Support:

In IGS SQL you can use inbuilt function, inbuilt library or inbuilt indicators while making a query. Please first go through IGS Formula Language manual to use advanced IGS SQL.

The IGS SQL is built on top of IGS FL and works the same way as IGS FL works. But it is SQL where you can filter or control displaying of data. Please note this is not a full fledged SQL that is used in databases but structured language specifically for stock market. It helps to filter and control display of stock market data. (User can save the data in csv format for further research)

Now suppose you need to use a customized indicator then before using it in SQL you need to define it and then use it in IGS SQL. For doing it first define your indicator like this

#indname1:<expression>

#indname2:<expression>

.

.
<sql query>

Now in SQL query you can use indicator with reference from today

Suppose you need to display stock with percentage change in price and would like to show the top gainer on top (descending order).

Here is a IGS SQL example using customized indicator.

```
#perchange: -1 * ( ( close[0] – close[-1] ) * 100 ) / close[-1]
```

```
select symbol,open[0], open[-1], perchange[0] from table sortby perchange[0]
```