Class Diagram Description

nitezh.ministock.activities.configure
ConfigureActivity#x# : Sets the id/size of the widget.
ConfigureActivityBase : Displays installation and configure message when widget is being added to screen and gets the size of the widget.

nitezh.**ministock.activities.widget**
WidgetColors: Contains the colors the widget uses
WidgetProvider#x#: Specific instance of WidgetProviderBase based on id/size
WidgetProviderBase: Main widget class containing inherited behavior from the Android API's AppWidgetProvider
WidgetRow: Object representation of a stock as a row in the widget
WidgetView: Decorates the widget depending on size and data

nitezh.**ministock.activities**
MinistocksActivity: Displays the initial menu
PortfolioActivity: Object Displays the stocks that the user is following
PreferencesActivity: Displays Preferences of the users for stocks
SymbolSearchableActivity: Displays symbol of the stock

nitezh.**ministock**.**dataaccess**
FxChangeRepository : Currency converter
GoogleStockQuoteRepository : Pulls data from google finance
IexStockQuoteRepository : Pulls data from iex trading
YahooStockQuoteRepository : Pulls data from yahoo finance
YahooStockQuoteRepository2 : Pulls data from yahoo finance with query url.

nitezh.ministock.domain
AndroidWidget: The app's widget that displays the stocks on the user's home screen
AndroidWidgetRepository: Contains the data pertaining to the stocks to be displayed on the widget.
PortfolioStock : Data to be displayed on the widget (price , symbol etc)
PortfolioStockRepository : Identity map class for all the chosen user stocks.
StockQuote : Object representation of a stock, inside the application
StockQuoteRepository: Identity map class for all the available stocks.
Widget: contains the initializations of the methods of the widget

WidgetRepository: contains all of the data pertaining to the widget, in other words stocks being tracked by the user.

WidgetStock: Keeps track of the trends of the stocks selected by the user and the stocks are displayed on the widget itself on the home screen

nitezh.ministock.utils
Cache: Short-term identity map for remotely obtained stock data
CurrencyTools: ADT containing static currency information
DateTools: Helper methods to facilitate date and time manipulations
InputTools: Unused class
NumberTools: Helper methods for number rounding and consistency
ReflectionTools: Used to give IDs to stock rows
StorageCache: Short term identity map for Storage subclasses
UrlDataTools: Helper methods for dealing with URLs
VersionTools: Contains only the current version number

nitezh.ministock
BackupAgent: Provides a complete backup of the app state (UserData), that can be restored on another device. Currently a beta feature
CustomAlarmManager: Manages time-based events such as refreshing data on an interval
DialogTools: Creates custom dialogs for displaying messages to users
PreferenceStorage: Stores app preferences to be retrieved on startup
StockSuggestions: Suggests stocks dynamically based on letters typed in the search bar
Storage: ADT interface for general storage purposes
SymbolProvider: Content provider for stock symbols
UserData: Contains user preferences and app state
WidgetProvider: Exactly the same as WidgetProviderBase (empty extension)