**Derivatives Pricing Workflow Project**

We are going to build a program for trader to price financial products. Here we focus on IR Swaps as an example. But please build the system in such a way that it's easy to **extend** to any other financial products.

The workflow is like this: the trader type in a ticker (in string format), and the system need to correctly parse and identify the right financial product for the input ticker. And then it will do the pricing based on the financial product type.

## Part 0:

Learn Swaps, explain its characteristics. How to price Swaps?

## Part 1:

**Create a ticker match machine** for various financial products. The ticker format is similar to Bloomberg ticker, but with our own customized pattern.

Let's starts with Rates IR Swap products.

IR Swap ticker format is:

*(ccy)(swap_type)[forward term](maturity)*

- ccy can be a list of configurable currencies, eg 'us', 'eu', 'cd'...
- Similarly, swap_type can be 'sw', 'ois', 'ff', 'mmab' etc.
- Maturity can be '3m', '6m', '1y', '10y', '30y' etc
- Forward term is optional, and have same format as Maturity.

Design and implement **the TickerMatcher** class, which shows whether the input ticker string is a valid ticker, and parse it accordingly.

- Input: ticker is a string.
- Output: s is a string to show whether the ticker is valid or not, and show all the components values if it's a valid ticker.

Eg, run below code:

```
if __name__ == '__main__':
  # Ticker Input
  matcher = TickerMatcher()
  for ticker in ('usois10y', 'bpff5y10y', 'ussw10x'):
     s = matcher.match(ticker)
     print(s+'\n')
```

The output should be below:

```
### usois10y matched successfully with IRSwap template ###
ccy:     us
swap_type:     ois
forward:
maturity:        10y

### bpff5y10y matched successfully with IRSwap template ###
ccy:     bp
swap_type:     ff
forward:         5y
```

maturity: 10y

*** ussw10x can NOT match with any template! ***

## Part 2:

**Price IR (Interest Rate) Swap.**

- 2.1: Explain the IR Swap pricing process.
  What are the factors/inputs for IR Swap pricing?
- 2.2: Implement the IR Swap pricing in Python.

We are going to use the USD Libor swap curve as at December 31 2018. Picture below shows the swap curve.


The hypothetical interest rate swap is as follows,
**Maturity**: 10 years
**Notional**: 10 Million USD
**Fixed rate**: 2.5%
**Floating rate**: Libor

# Help Information

**IR Swap In a nutshell:**

An interest rate swap's (IRS's) effective description is a derivative contract, agreed between two counterparties, which specifies the nature of an exchange of payments benchmarked against an interest rate index. The most common IRS is a fixed for floating swap, whereby one party will make payments to the other based on an initially agreed fixed rate of interest, to receive back payments based on a floating interest rate index. Each of these series of payments is termed a 'leg', so a typical IRS has both a fixed and a floating leg. The floating index is commonly an interbank offered rate (IBOR) of specific tenor in the appropriate currency of the IRS, for example LIBOR in USD, GBP, EURIBOR in EUR or STIBOR in SEK. To completely determine any IRS a number of parameters must be specified for each leg; the notional principal amount (or varying notional schedule), the start and end dates and date scheduling, the fixed rate, the chosen floating interest rate index tenor, and day count conventions for interest calculations.

**Python Library:**

**QuantLib**: https://www.quantlib.org/