

Basel III 市場風險標準法試算實作班

使用 Python 與 QuantLib 套件

昀騰金融科技

技術長

董夢雲 博士

dongmy@ms5.hinet.net

主題二、Python 開發環境與 QuantLib 套件使用說明

- 一、QuantLib 程式庫的由來
- 二、QuantLib 程式庫的下載
- 三、QuantLib-Python Package 下載與安裝
- 四、QuantLib Package 使用與學習

程式範例：`hello.py`、`FXFwd.py`

一、QuantLib 程式庫的由來

- ◆ QuantLib 是一個開放源碼程式庫(Open-Source Software Library)。
- 目的在提供對金融工具評價和相關主題(交易避險、風險計算)有興趣的軟體開發人員，適合的開發工具。
- 可在 QuantLib 的網站，<https://www.quantlib.org>，了解專案的相關內容。

檔案 (F) 編輯 (E) 檢視 (V) 歷史 (S) 書籤 (B) 工具 (I) 說明 (H)

Q QuantLib, a free/open-sou X +

https://www.quantlib.org/index.shtml

110% G+ 分享

QuantLib

A free/open-source library for quantitative finance

Get QuantLib

Head to our [download](#) page to get the latest official release, or check out the latest development version from our [git](#) repository. QuantLib is also available in [other languages](#).

Documentation

[Documentation is available](#) in several formats from a number of sources. You can also read our [installation instructions](#) to get QuantLib working on your computer.

Need Help?

If you need to ask a question, subscribe to our [mailing list](#) and post it there. Before doing that, though, you might want to look at the [FAQ](#) and check if it was already answered.

Found a bug?

Open an issue on [GitHub](#); if you have a patch, [open a pull request](#) instead.

Want to contribute?

Just fork our repository on [GitHub](#) and start coding (instructions are [here](#)). Please have a look at our [developer intro](#) and [guidelines](#).

More info

Here is the QuantLib [license](#), the [list of contributors](#), and the [version history](#).

 OSI certified

Hosted on
GitHub

Supported by


The QuantLib project is aimed at providing a comprehensive software framework for quantitative finance. QuantLib is a [free/open-source](#) library for modeling, trading, and risk management in real-life.

◆ QuantLib 是以 C++ 開發的程式庫，以此為基礎，被轉寫成其他不同的語言。

- 包括 C#、Java、Perl、Python、R、Ruby 與 Scheme。

The screenshot shows a web browser window with the URL <https://www.quantlib.org/extensions.shtml>. The page title is "QuantLib in other languages". The main content area has a yellow sidebar on the right.

Other languages and platforms

QuantLib is available as a C#, Guile, Java, MzScheme, Perl, Python, and Ruby module by means of SWIG. Experimental bindings to GNU R and Objective Caml are also available; feedback is welcome.

[QuantLibAddin](#) exports a procedural interface to a number of platforms including Microsoft Excel (see the [QuantLibXL](#) site) and OpenOffice/LibreOffice Calc. The [LibreOffice QuantLib Addin](#) project also exports QuantLib to Calc.

A modified QuantLib C++ library enabling adjoint automatic differentiation (AAD) is available from <https://github.com/compatibl/QuantLibAdjoint>.

[Deriscope](#) is another project that aims at exporting QuantLib functionality to Excel.

GNU R support is provided by means of [RQuantLib](#) by Dirk Eddelbuettel.

An initial web API for QuantLib (also usable from Google Sheets) is available from [quantra.io](#).

A project for porting QuantLib to C# has started as [QLNet](#) and is looking for developers.

The [JQuantLib](#) project aims at a 100% Java port. At this time, it provides about 65% of QuantLib 0.9.7.

An alternative set of Python wrappers is provided in the [PyQL](#) project by Didrik Pinte and Patrick Henaff.

[QuantLib.jl](#) is a port of QuantLib to the Julia language. The project is in its initial stages and welcomes contributions.

Bindings for Node.js are available from the [quantlibnode](#) project.

[QLDDS](#) is a project that allows the functionality of the QuantLibAddin for C++ to be distributed via OpenDDS across multiple computers running different operating systems.

Get QuantLib
Head to our [download](#) page to get the latest official release, or check out the latest development version from our [git](#) repository. QuantLib is also available in [other languages](#).

Documentation
Documentation is available in several formats from a number of sources. You can also read our [installation instructions](#) to get QuantLib working on your computer.

Need Help?
If you need to ask a question, subscribe to our [mailing list](#) and post it there. Before doing that, though, you might want to look at the [FAQ](#) and check if it was already answered.

Found a bug?
Open an issue on [GitHub](#); if you have a patch, [open a pull request](#) instead.

(一)QuantLib 的歷史

- ◆ 由一群在 Cabota Banca Intesa，利率衍生商品交易台工作的數量分析專家們，於西元 2000 年時，所開創的一個財務金融程式庫開發專案。
 - 目前這些專家們已經成立了一家公司，該公司之前稱之為 RiskMap，現在則命名為 StatPro Italia。
 - ✓ 不少學術界發表的論文，以 QuantLib 進行計算或以之作為對照。
 - ✓ 2019 年 StatPro 被 Confluence 以兩億美金收購。
- ◆ QuantLib 專案目前是由 Luigi Ballabio 與 Ferdinando Ametrano 兩位專家所領導。
 - Ferdinando Ametrano 分享他在另一個結構化專案中，實作 Black-Scholes 公式的開發經驗。
 - 該專案允許一種新的合作方式，專案中的任何人可以改進、更正來開發一個免費的共用基礎框架 (Free Common-base Framework)。
 - ✓ 類似極限或敏捷開發的方式。
 - 此一意見為 RiskMap 所支持並加以採用。

- ◆ QuantLib 的創建者秉持著這樣的信念，公開的標準最適合於科學與技術的演進。
 - 尤其在學術界之內，好的實務技能與工具只有被教育界所接受，方可在長期的演化中勝出。
 - 雖然最早專案的目標對象為學術界與實務界，然而最後卻是進一步促成這兩方人士進一步的交流。

- ◆ QuantLib 所提供的工具，不論對實務上的實作或是進階的建模，都是相當有用的。
 - 財金學者 Mark Joshi 在其專著(C++ Design Patterns and Derivatives Pricing, 2nd Ed. 2008)中指出，《熟悉使用 QuantLib 已經成為財工人員未來的一項重要技能》(page 177)。
 - 德國工業銀行(IKB Deutsche Industriebank AG, <https://www.ikb.de/>)積極贊助其活動，並使用 QuantLib 作為中台模型驗證工具。
 - ✓ 國內多家大型銀行交易室的財工人員，也有在使用。

◆ 下表列示了 QuantLib 各版本釋出的時間，目前最新的版本是 1.24(至 2021/11/12)

- 除了 C++之外，其他語言的 QuantLib 專案也路續成立運作，
 - ✓ 使用 C# 的 QLNet，<https://sourceforge.net/projects/qlnet/>；
 - ✓ 使用 R 的 RQuantLib，<https://dirk.eddelbuettel.com/code/rquantlib.html>；
 - ✓ 使用 Java 的 JQuantLib，<https://www.jquantlib.org>；
 - ✓ 使用 Python 的 PyQL，<https://github.com/enthought/pyql>；
 - ✓ 以及可配合 Excel 使用的增益集，本專案的子專案 QuantLibXL。
- 透過 SWIG，可供多種語言使用的模組，本次演說使用此版本。
 - ✓ 包含 C#，Java，Perl，Python，R，Ruby，Scala。

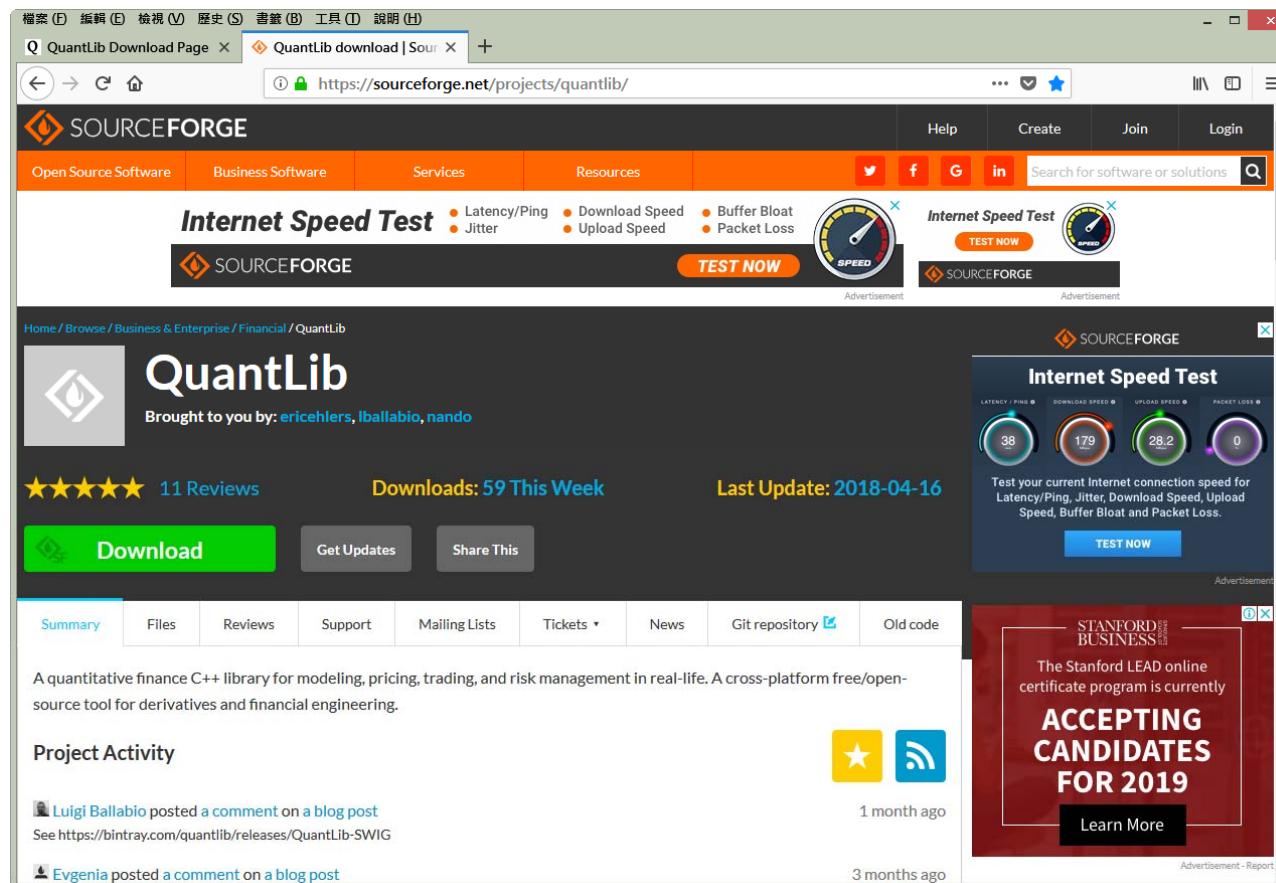
◆ QuantLib 各早期版本的演進與推出時間表(2021/11/2)。

Version	Release date	Notes
0.1.1	Nov 21, 2000	
0.2.0	Sep 18, 2001	
0.3.4	Nov 21, 2003.	本人首次發現，在 KGI 。
0.3.7	Jul 23, 2004.	此版之後 QuantLib 需要 Boost 程式庫。
0.4.0	Feb 20, 2007.	
0.8.0	May 30, 2007.	版本的大幅更動，目的在加快收斂到 1.0 版。
0.9.0	Dec 24, 2007.	
0.9.9	Nov 2009.	
1.0.0	Feb 24, 2010	正式釋出
1.0.1	Sep 17, 2010	
1.1.0	May 23, 2011	
1.2.0	March 6, 2012	
...	...	
1.9.0	Nov 8, 2016	開始使用 VS2008
1.12	Feb 1, 2018	
1.13	May 24, 2018	最後使用 VS2008 ,
1.14	Oct 1, 2018	開始使用 VS2010 ,
1.15	Feb 19, 2019	

1.16	Aug 5, 2019	開始使用 VS2019 ,
1.17	Dec 3, 2019	
1.18	Mar 23, 2020	
1.19	Jul 20, 2020	
1.20	Oct 26, 2020	
1.21	Jan 20, 2021	
1.22	April, 2021	
1.23	July 14, 2021	
1.24	Oct 19, 2021	

二、QuantLib 程式庫的下載

- ◆ 可在 Open Source Software 網站內的 QuantLib 專案網頁下載，只到 1.12 版。
 - <http://sourceforge.net/projects/quantlib/> , 2018/09/16。



◆ 點 Files，可以看到下面目錄結構。

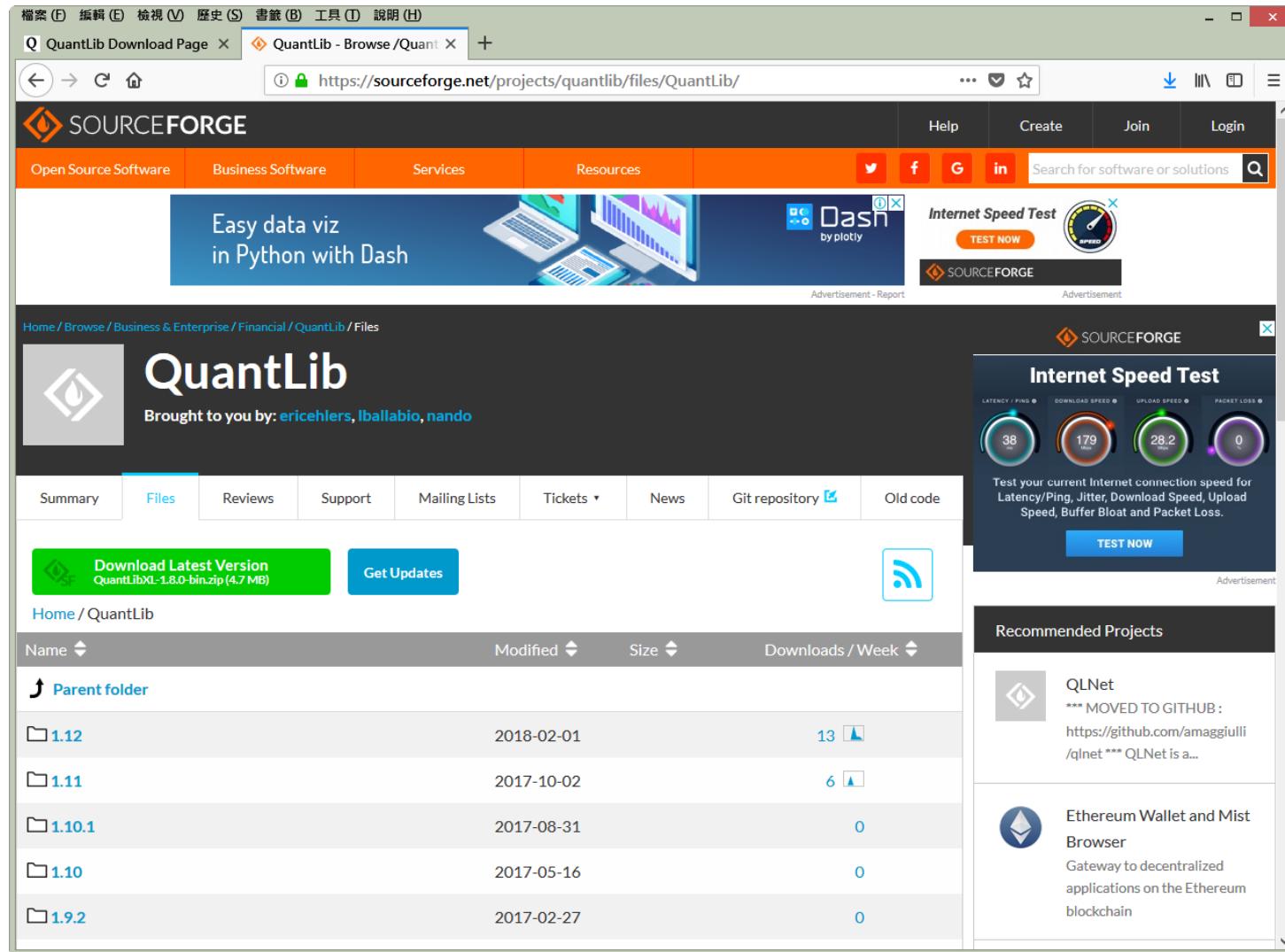
➤ 點選 QuantLib，

The screenshot shows a web browser window displaying the SourceForge project page for QuantLib. The URL in the address bar is <https://sourceforge.net/projects/quantlib/files/>. The page header includes the SourceForge logo and navigation links for Help, Create, Join, and Login. A search bar is also present. A prominent advertisement for TestRail is displayed at the top. Below it, another advertisement for Internet Speed Test is shown, featuring four circular performance metrics: Latency/Ping (38), Download Speed (179), Upload Speed (28.2), and Packet Loss (0). The main content area is titled "QuantLib" and shows the following file structure:

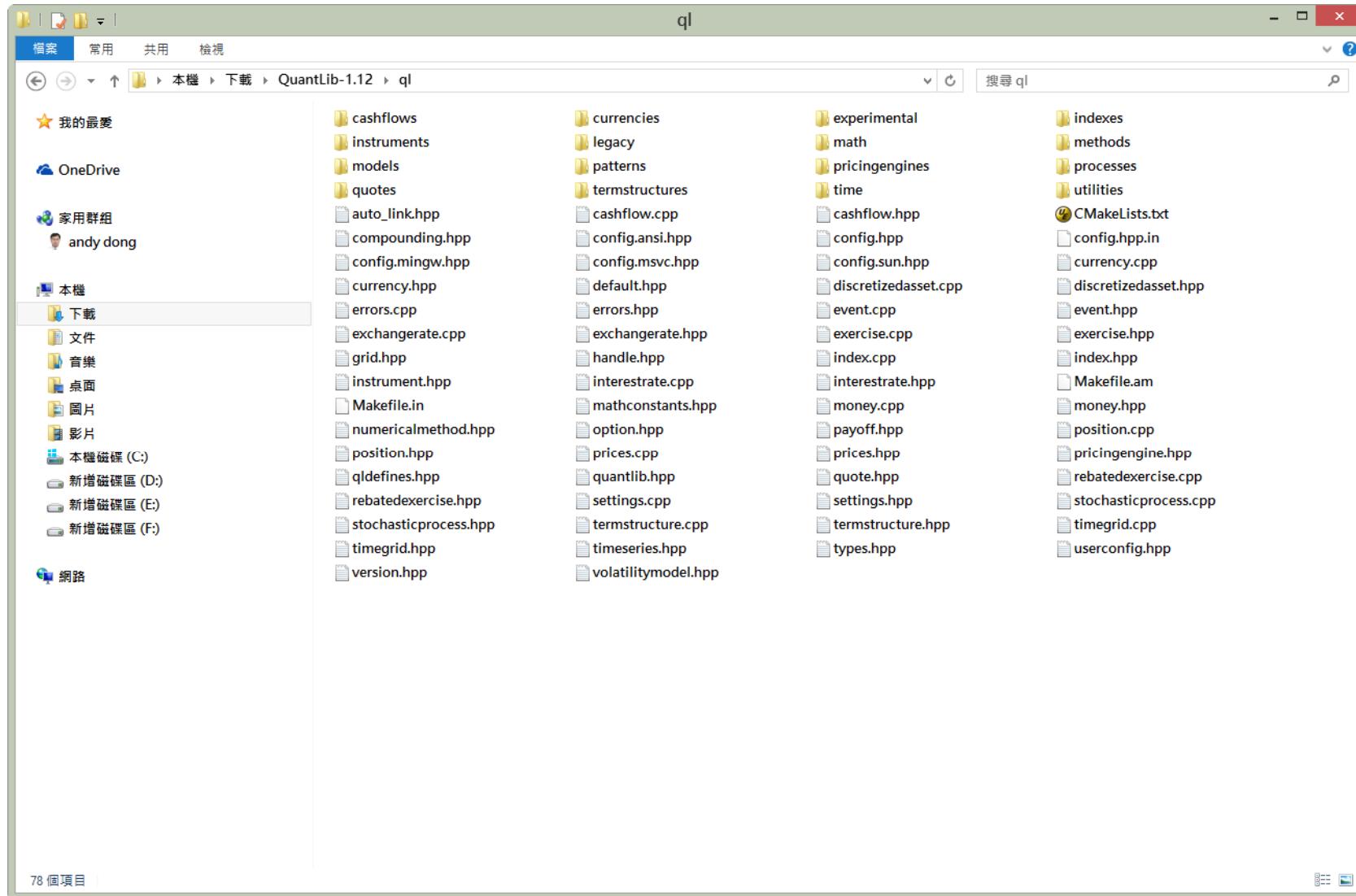
Name	Modified	Size	Downloads / Week
QuantLib	2018-04-16		30
QuantLibAddin	2018-02-27		2
QuantLibXL	2018-02-27		6
ObjectHandler	2018-02-15		12
test	2017-04-12		2
reposit	2016-05-29		7

On the right side of the page, there are two recommended projects: QLNet and Ethereum Wallet and Mist Browser.

◆ SourceForge 上最後的版本，1.12 版。



◆ 下載安裝後，完整的目錄，有 2 千多個檔案。



◆ QuantLib 的 Wiki 網頁為，<http://en.wikipedia.org/wiki/QuantLib>。

The screenshot shows a Microsoft Edge browser window displaying the English Wikipedia article for "QuantLib". The title bar indicates the URL is <https://en.wikipedia.org/wiki/QuantLib>. The page content starts with the heading "QuantLib" and a summary paragraph about the library. A prominent orange warning box is present, stating: "This article has multiple issues. Please help improve it [hide] or discuss these issues on the talk page. (Learn how and when to remove these template messages)" followed by a bulleted list of issues. Below the summary, there is a section about the library's purpose and its implementation in C++. On the left side, the Wikipedia sidebar contains links like Main page, Contents, and Recent changes.

◆ 新版可在 <https://github.com/lballabio/QuantLib/>，網站下載，2021/11/12。

The screenshot shows a Firefox browser window with the following details:

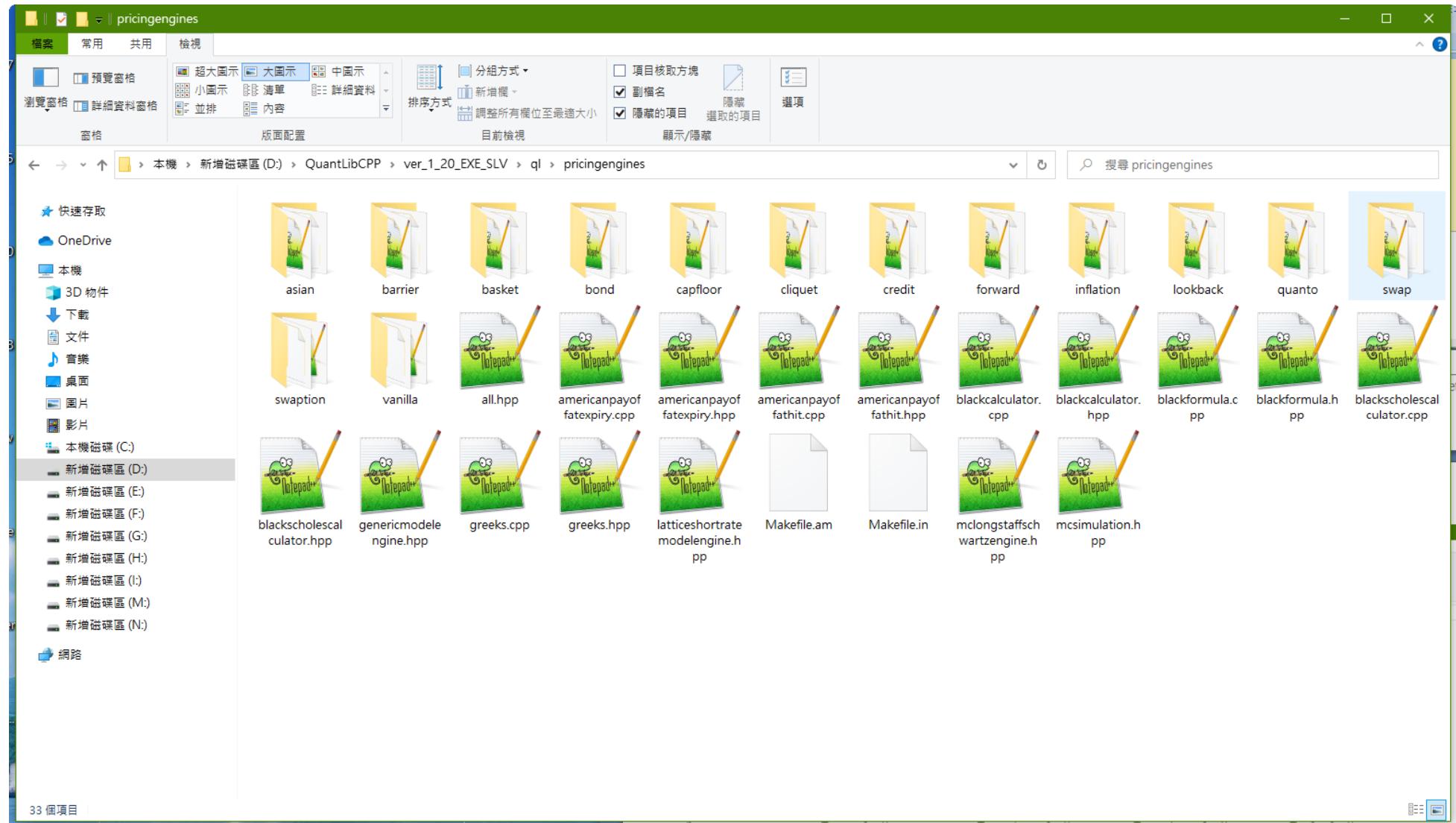
- Address Bar:** https://github.com/lballabio/QuantLib/releases/tag/QuantLib-v1.24
- Page Title:** Release 1.24 · lballabio/QuantLib
- Header:** Search or jump to... (with a magnifying glass icon), Pull requests, Issues, Marketplace, Explore.
- Profile Bar:** Watch (with dropdown), 234, Star (with star icon), 2.9k, Fork (with fork icon), 1.2k.
- Section Headers:** Releases / QuantLib-v1.24, 1.24 (Latest).
- Release Details:** lballabio released this 24 days ago, 41 commits to master since this release, QuantLib-v1.24, 88e3a1b.
- Downloads:** QuantLib-1.24.tar.gz, QuantLib-1.24.zip.
- Changes:** Changes for QuantLib 1.24: QuantLib 1.24 includes 25 pull requests from several contributors.

(一)QuantLib 程式庫的內容

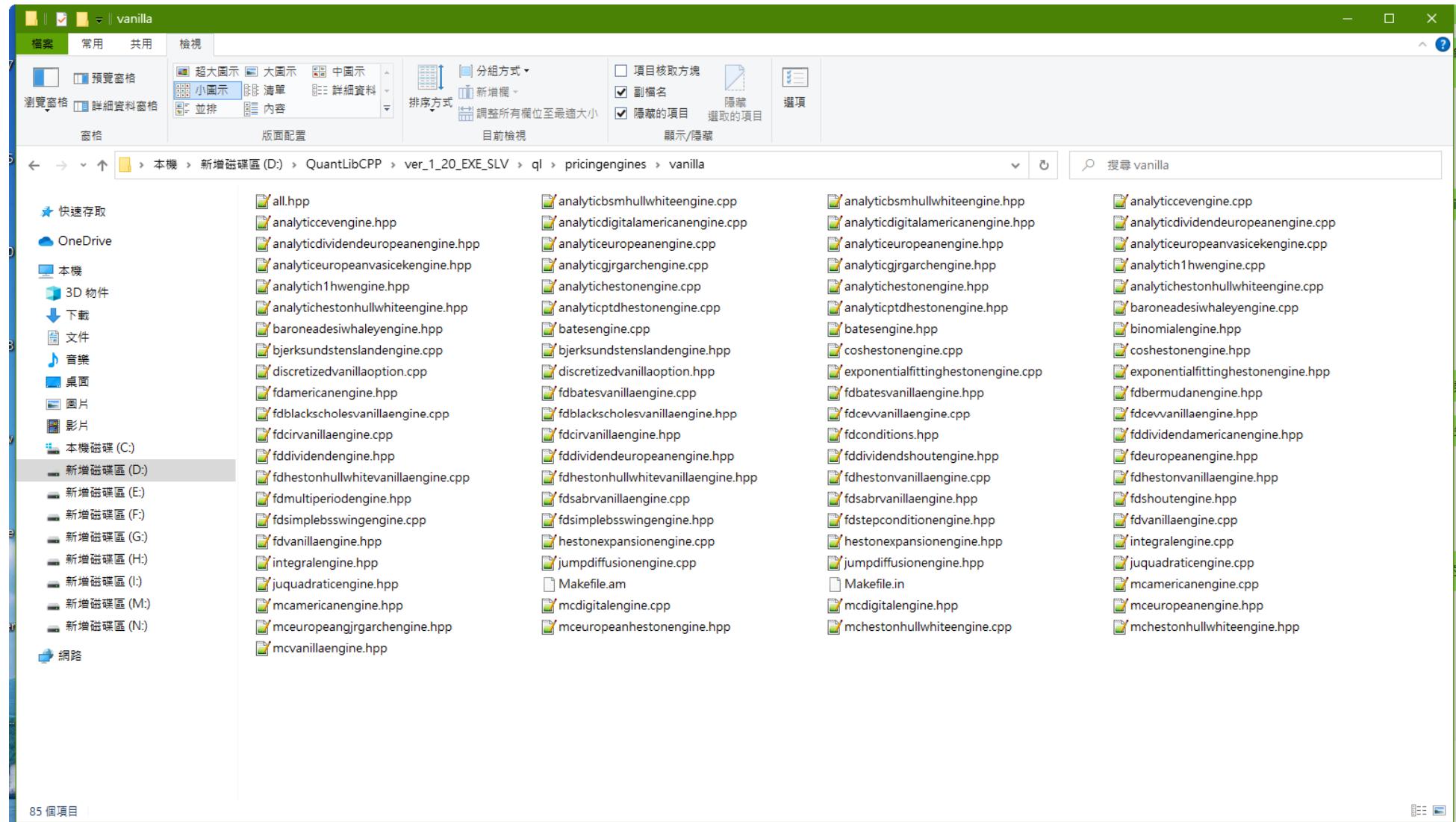
- ◆ QuantLib 程式庫內容豐富，包含的檔案兩千餘個，可以概分 15 類模組。
 - 1.Numeric types：主要在定義各類資料的資料型別，例如，利率(Rate)、利差(Spread)、波動性(Volatility)是實數(Real)。
 - 2.Currencies and FX rates：包括 66 種貨幣類別，以及匯率轉換計算與管理的物件。
 - 3.Date and time calculations：包括 37 種不同國家/地區日曆類別、7 種不同計息方式類別，以及相關日期計算的物件。
 - 4.Pricing engines：九大類評價引擎，包括 Asian option、Barrier option、Basket option、Cap/Floor、Cliquet option、Forward option、Quanto option、Swaption、Vanilla option 等。每一類都有解析解、二元樹、有限差分法與蒙地卡羅模擬的實作類別。
 - 5.Finite-differences framework：三大類有限差分法的實作類別。
 - 6.Short-rate modelling framework：包括單因子模型，Vasicel、CIR、Hull-White、Black-Karasinski、Extended CIR 等模型，以及雙因子模型的 G2 模型。

- 7.Financial instruments : 40 多種的金融工具，含 Swap、Vanilla Option、Exotic Option、Stock、Forward、Cap、Floor、Color、Bond、Future、Callable Bond 以及相對應的 Quanto 產品、Inflation Bond。
- 8.Lattice methods : 一維與二維的二元樹和三元樹模型。
- 9.Math tools : 包括分配、積分、相關性、內差、矩陣、最適化、亂數、求解、統計等九類數學工具。
- 10.Monte Carlo framework : 單變數與多變數的歐式與美式模擬方法物件。
- 11.Design patterns : Singleton、Observer/Observable、Lazy Object、Composite、Curiously Recurring Template、Acyclic Visitor 樣式。
- 12.Stochastic processes : 幾何布朗運動、隨機波動模型、Square Root Process、Ornstein Uhlenbeck Process、Hull White Process、G2 Process。
- 13.Term structures : 包括利率、波動性、信用與通膨的期限結構物件。
- 14.Models : 分別包括 Equity Model、Short Rate Model、Volatility Model 中使用的相關模型，以及針對 Libor/Swap Market Model 涉及的金融工具與模擬方法的相關物件上百個，。
- 15.QuantLib macros : 一些數值極限與除錯所需的 Macros。

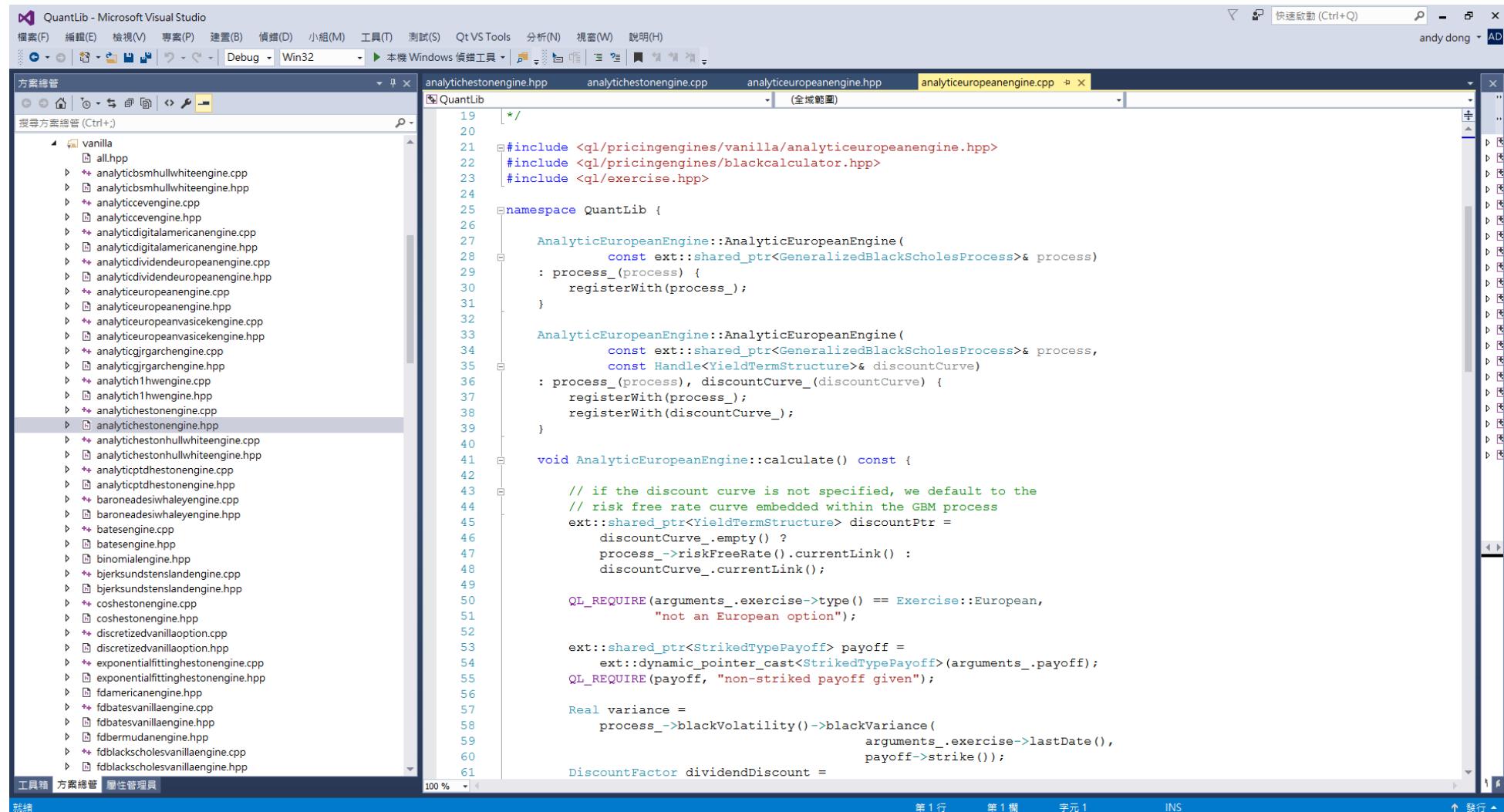
➤ 各類產品評價引擎接有。



➤ 解析解、樹狀模型、有限差分法、蒙地卡羅模擬法，各類模型皆有。



➤ 原始碼有提供，足以驗證了解，下面 Black-Scholes 解析解模型。



The screenshot shows the Microsoft Visual Studio interface with the QuantLib project open. The left pane displays the Solution Explorer with various files listed under the 'vanilla' folder. The right pane shows the code editor with the file 'analyticengine.cpp' open, which contains the implementation of the AnalyticEuropeanEngine class. The code includes imports for ql/pricingengines/vanilla/analyticeuropeanengine.hpp, ql/pricingengines/blackcalculator.hpp, and ql/exercise.hpp. It defines the AnalyticEuropeanEngine class with methods for calculating options based on different processes and discount curves. The code uses C++11 features like shared_ptr and lambda functions.

```
19  /*
20
21  #include <ql/pricingengines/vanilla/analyticeuropeanengine.hpp>
22  #include <ql/pricingengines/blackcalculator.hpp>
23  #include <ql/exercise.hpp>
24
25  namespace QuantLib {
26
27      AnalyticEuropeanEngine::AnalyticEuropeanEngine(
28          const ext::shared_ptr<GeneralizedBlackScholesProcess>& process)
29      : process_(process) {
30          registerWith(process_);
31      }
32
33      AnalyticEuropeanEngine::AnalyticEuropeanEngine(
34          const ext::shared_ptr<GeneralizedBlackScholesProcess>& process,
35          const Handle<YieldTermStructure>& discountCurve)
36      : process_(process), discountCurve_(discountCurve) {
37          registerWith(process_);
38          registerWith(discountCurve_);
39      }
40
41      void AnalyticEuropeanEngine::calculate() const {
42
43          // if the discount curve is not specified, we default to the
44          // risk free rate curve embedded within the GBM process
45          ext::shared_ptr<YieldTermStructure> discountPtr =
46              discountCurve_.empty() ?
47                  process_->riskFreeRate().currentLink() :
48                  discountCurve_.currentLink();
49
50          QL_REQUIRE(arguments_.exercise->type() == Exercise::European,
51                     "not an European option");
52
53          ext::shared_ptr<StrikedTypePayoff> payoff =
54              ext::dynamic_pointer_cast<StrikedTypePayoff>(arguments_.payoff);
55          QL_REQUIRE(payoff, "non-striken payoff given");
56
57          Real variance =
58              process_->blackVolatility()->blackVariance(
59                  arguments_.exercise->lastDate(),
60                  payoff->strike());
61
62          DiscountFactor dividendDiscount =
```

➤ 模型參考文件有提供，方便學習了解，下面 Heston 解析解模型。

The screenshot shows the Microsoft Visual Studio interface with the QuantLib project open. The Solution Explorer on the left lists various source files and header files for the project. The main code editor window displays the `analytichestonengine.cpp` file, which contains C++ code for an Heston engine. The code includes comments and references to academic papers and software papers. The status bar at the bottom indicates the current line (第 1 行), cursor position (第 1 列), character count (字元 1), and mode (INS).

```
52 // *! References:
53
54 Heston, Steven L., 1993. A Closed-Form Solution for Options
55 with Stochastic Volatility with Applications to Bond and
56 Currency Options. The review of Financial Studies, Volume 6,
57 Issue 2, 327-343.
58
59 A. Sepp, Pricing European-Style Options under Jump Diffusion
60 Processes with Stochastic Volatility: Applications of Fourier
61 Transform (<http://math.ut.ee/~spartak/papers/stochjumpvols.pdf>)
62
63 R. Lord and C. Kahl, Why the rotation count algorithm works,
64 http://papers.ssrn.com/sol3/papers.cfm?abstract\_id=921335
65
66 H. Albrecher, P. Mayer, W. Schoutens and J. Tistaert,
67 The Little Heston Trap, http://www.schoutens.be/HestonTrap.pdf
68
69 J. Gatheral, The Volatility Surface: A Practitioner's Guide,
70 Wiley Finance
71
72 F. Le Floc'h, Fourier Integration and Stochastic Volatility
73 Calibration,
74 https://papers.ssrn.com/sol3/papers.cfm?abstract\_id=2362968
75
76 L. Andersen, and V. Piterbarg, 2010,
77 Interest Rate Modeling, Volume I: Foundations and Vanilla Models,
78 Atlantic Financial Press London.
79
80 \ingroup vanillaengines
81
82 // test the correctness of the returned value is tested by
83 // reproducing results available in web/literature
84 // and comparison with Black pricing.
85
86 */
87 class AnalyticHestonEngine
88     : public GenericModelEngine<HestonModel,
89                             VanillaOption::arguments,
90                             VanillaOption::results> {
91
92     public:
93         class Integration;
94         enum ComplexLogFormula {
95             // Gatheral form of characteristic function w/o control variate
96         };
97     };
98 }
```

◆ Confluence 併購 statpro，使用 QuantLib 提供金融計算服務給全球主要金融機構。

The screenshot shows the Confluence website homepage. At the top, there's a green header bar with the Confluence logo and a search bar. Below the header, the main content area features a large banner with the text "We are Confluence" and "Empowering Knowledge." A sidebar on the left has a "Subscribe" button. In the center, there's a tablet displaying a financial dashboard with various charts and data points. Below the banner, there are four statistics: "400+", "40", "10/10", and "10/10". Each statistic has a corresponding descriptive sentence below it.

We are Confluence

Empowering Knowledge.

We are a global leader in data-driven investment management solutions partnering with our clients to deliver products and services designed to optimize efficiency and control.

[Learn more about us >](#)

400+

Over 400 clients use our services and solutions

40

Servicing clients in 40 countries worldwide

10/10

The top 10 asset managers rely on our solutions

10/10

The top 10 service providers rely on our solutions

➤ QuantLib 程式庫的品質受到市場的認同。

The screenshot shows a web browser window with the URL <https://www.confluence.com>. The page header includes a green navigation bar with Chinese menu items (檔案, 編輯, 檢視, 歷史, 書籤, 工具, 說明) and a tab labeled "Home - Confluence Technologies". Below the header is a dark grey header bar with the Confluence logo, a search bar, and social media links for Facebook, Twitter, and LinkedIn. The main content area features a large banner with the text "What we are doing" and "Latest News". On the left, there is a "Subscribe" button. The news section displays three articles with images and titles:

- Confluence and J.P. Morgan Expand Partnership, Integrating Additional Asset Coverage into Delta Risk and Performance Platform**
Pittsburgh, PA, April 4, 2021 – Confluence Technologies, Inc. ("Confluence"), a global technology solutions provider delivering innovative products to the worldwide money management industry...
- Futuregrowth Asset Management Expands Strategic Relationship with Confluence**
Pittsburgh, PA, April 20, 2021 – Confluence Technologies, Inc. ("Confluence"), a global technology solutions provider that helps the investment management industry solve complex investment da...
- Confluence Reaches Agreement in Principle to Acquire Fundpeak, Global Provider of Fund Portals and Factsheets**
Pittsburgh, PA, April 15, 2021 – Confluence Technologies, a global technology solutions provider helping the investment management industry solve complex investment data challenges, today announc...

三、QuantLib-Python Package 下載與安裝



The screenshot shows a Firefox browser window with the URL <https://www.quantlib.org/install/windows-python.shtml>. The page content is as follows:

QuantLib

A free/open-source library for quantitative finance

QuantLib-Python installation on Windows

Installation from PyPI

If you don't need to modify the wrappers, you might want to try installing a precompiled binary version. The availability of binaries depend on your operating system; to try to install them, run:

```
pip install QuantLib
```

(If you have multiple versions of Python installed, run the above with the one you want to use QuantLib with.) If a binary package is available for your system, it will be installed and you will be able to leave this page and use it right away; if not, you'll have to compile it yourself as described in the next section.

Installation from a released version

Get QuantLib

Head to our [download](#) page to get the latest official release, or check out the latest development version from our [git](#) repository. QuantLib is also available in [other languages](#).

Documentation

[Documentation](#) is available in several formats from a number of sources. You can also read our [installation instructions](#) to get QuantLib working on your computer.

◆ Source Code : 自行編譯 Python 套件

The screenshot shows a Firefox browser window with the address bar at <https://github.com/lballabio/QuantLib-SWIG>. The GitHub page for the repository is displayed, featuring the repository name 'lballabio / QuantLib-SWIG' and various navigation links like 'Code', 'Issues (9)', 'Pull requests (2)', 'Actions', 'Security', and 'Insights'. On the left, there's a list of recent commits from the 'master' branch, each with a small icon, the author's name ('lballabio'), the commit message, the commit hash ('98b4ec7'), the time of the commit ('2 days ago'), and the number of commits ('2,121 commits'). To the right of the commits, there's an 'About' section with a brief description of the repository, a 'quantitative-finance' tag, and links to 'Readme' and 'View license'. Below that is a 'Releases' section showing the latest release '1.22' (Latest, on 15 Apr) and a link to '+ 25 releases'.

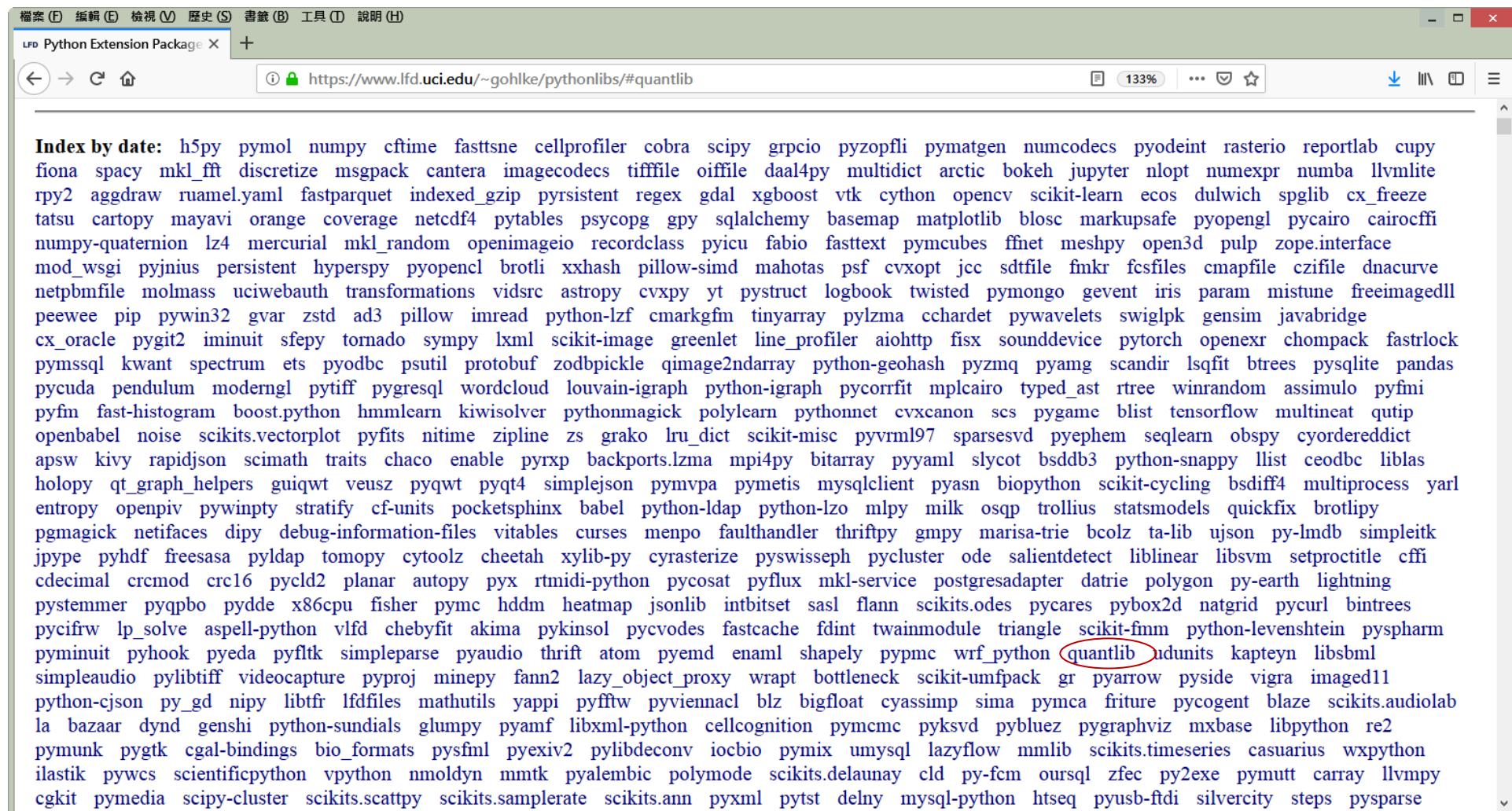
Commit	Message	Author	Hash	Time	Commits
.build	Remove support for Python 2.7	lballabio	98b4ec7	2 days ago	2,121
.github	Replace stale application with action.	lballabio		6 months ago	
CSharp	Move Type enumeration to base Swap class	lballabio		2 months ago	
Java	Remove features deprecated in version 1.17	lballabio		last month	
Python	Set version to 1.23 rc	lballabio		6 months ago	
R	Set version to 1.23-dev	lballabio		2 days ago	
SWIG	Merge pull request #393.	lballabio		3 months ago	
				2 days ago	

◆ 舊版 whl 檔下載網址

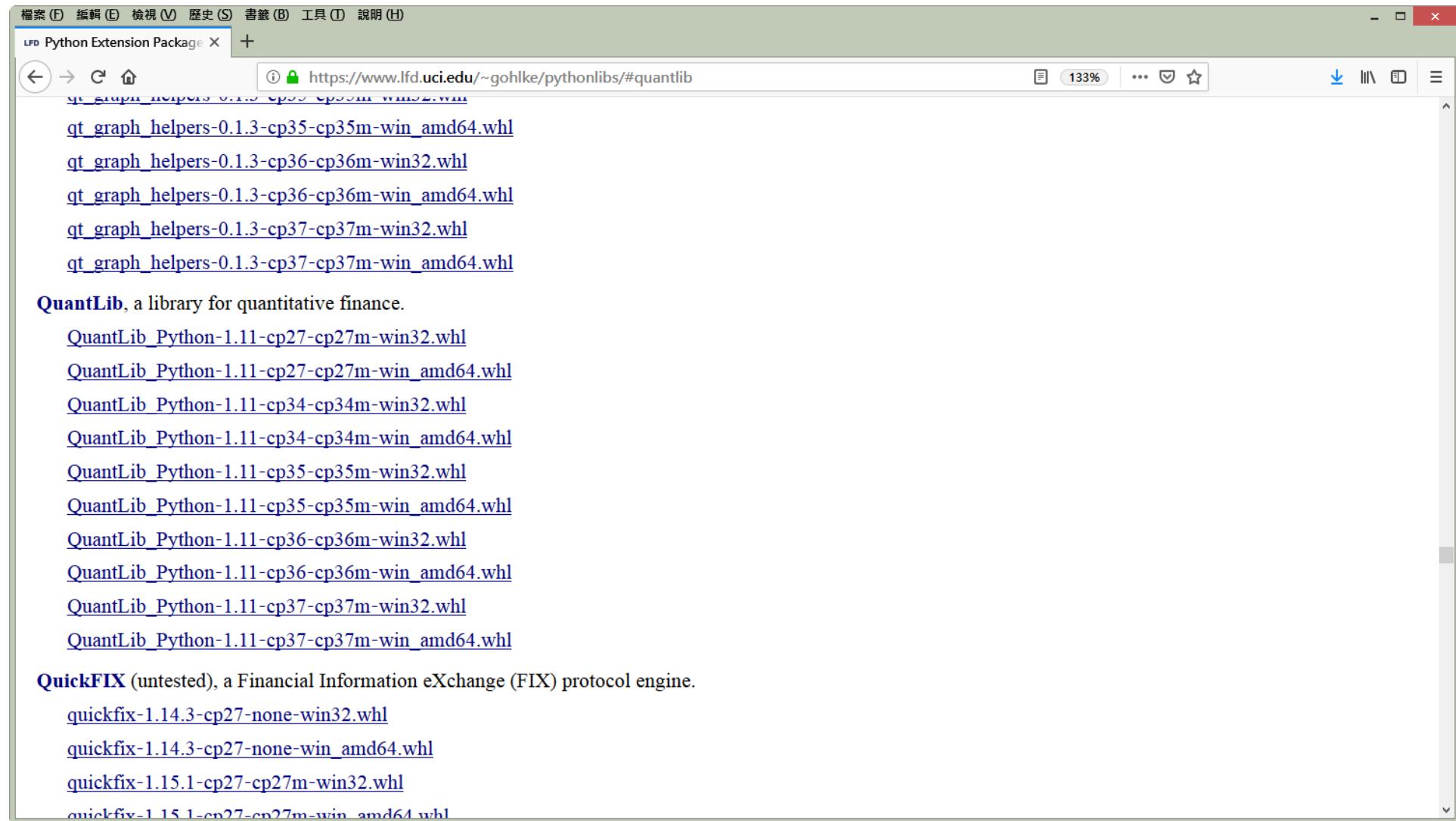
The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** LFD Python Extension Package
- Address Bar:** https://www.lfd.uci.edu/~ohlke/pythonlibs/#quantlib
- Content Area:**
 - # Unofficial Windows Binaries for Python Extension Packages
 - by **Christoph Gohlke, Laboratory for Fluorescence Dynamics, University of California, Irvine.**
 - This page provides 32- and 64-bit Windows binaries of many scientific open-source extension packages for the official [CPython distribution](#) of the [Python](#) programming language. A few binaries are available for the [PyPy](#) distribution.
 - The files are unofficial (meaning: informal, unrecognized, personal, unsupported, no warranty, no liability, provided "as is") and made available for testing and evaluation purposes.
 - Most binaries are built from source code found on [PyPI](#) or in the projects public revision control systems. Source code changes, if any, have been submitted to the project maintainers or are included in the packages.
 - Refer to the documentation of the individual packages for license restrictions and dependencies.
 - If downloads fail, reload this page, enable JavaScript, disable download managers, disable proxies, clear cache, use Firefox, reduce number and frequency of downloads. Please only download files manually as needed.
 - Use [pip](#) version 9 or newer to [install the downloaded .whl files](#). This page is not a pip package index.
 - Many binaries depend on [numpy-1.14+mkl](#) and the Microsoft Visual C++ 2008 ([x64](#), [x86](#), and [SP1](#) for CPython 2.7), Visual C++ 2010 ([x64](#), [x86](#), for CPython 3.4), or the Visual C++ 2017 ([x64](#) or [x86](#) for CPython 3.5, 3.6, and 3.7) redistributable packages.
 - Install [numpy+mkl](#) before other packages that depend on it.
 - The binaries are compatible with the most recent official CPython distributions on Windows >=6.0. Chances are they do not work with custom Python distributions included with Blender, Maya, ArcGIS, OSGeo4W, ABAQUS, Cygwin, Pythonxy, Canopy, EPD, Anaconda, WinPython etc. Many binaries are not compatible with Windows XP or Wine.
 - The packages are ZIP or 7z files, which allows for manual or scripted installation or repackaging of the content.
 - The files are provided "as is" without warranty or support of any kind. The entire risk as to the quality and performance is with you.**
 - The opinions or statements expressed on this page should not be taken as a position or endorsement of the Laboratory for Fluorescence Dynamics or the University of California.

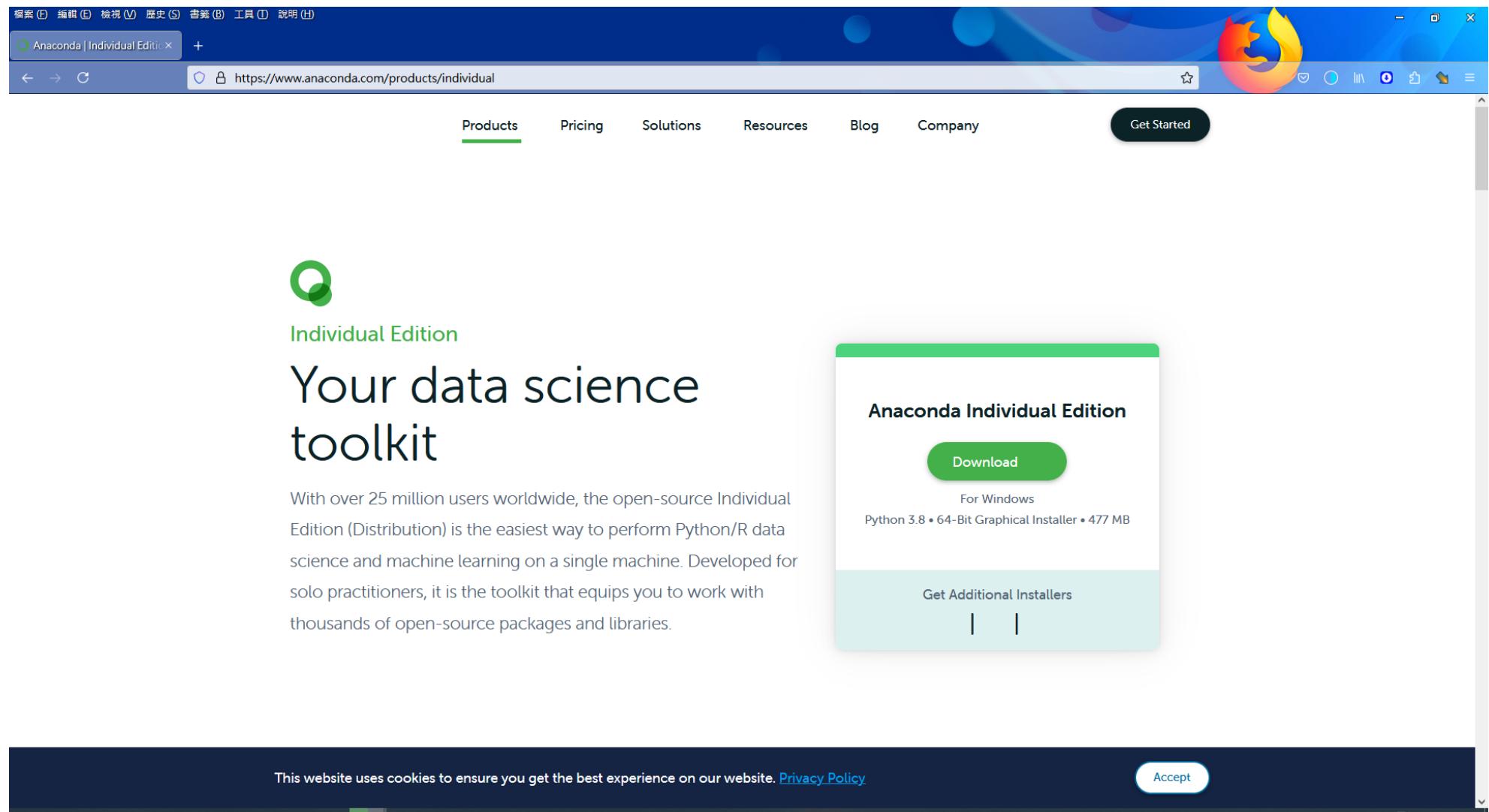
➤ Package 索引



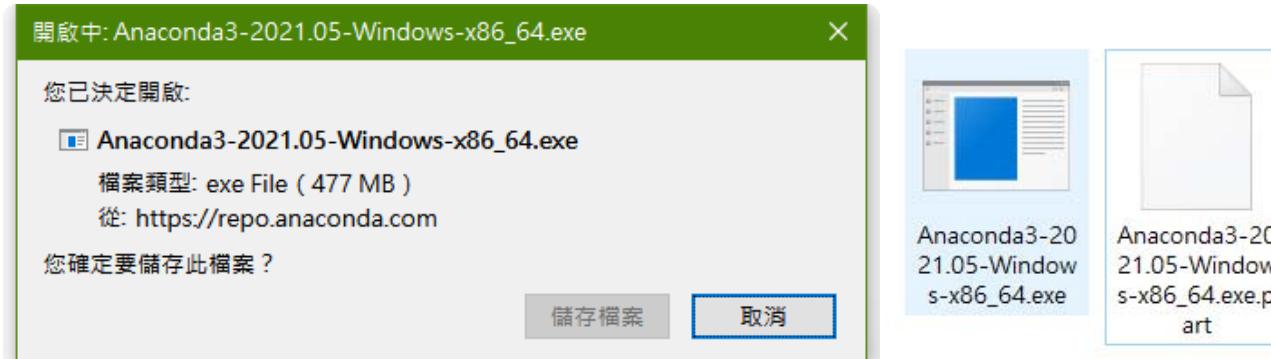
➤ QuantLib 檔案



◆ 本課程的 Python 使用 Anaconda 的發行版本。



➤ 下載的檔案



➤ 完成後，點選執行安裝，桌面出現 Anaconda3 的圖像。



◆ 在開始按鈕選擇



◆ 點選執行 Anaconda3 圖像，

```
(base) C:\Users\andyd>dir/w
磁碟區 C 中的磁碟沒有標籤。
磁碟區序號： 183C-35D4

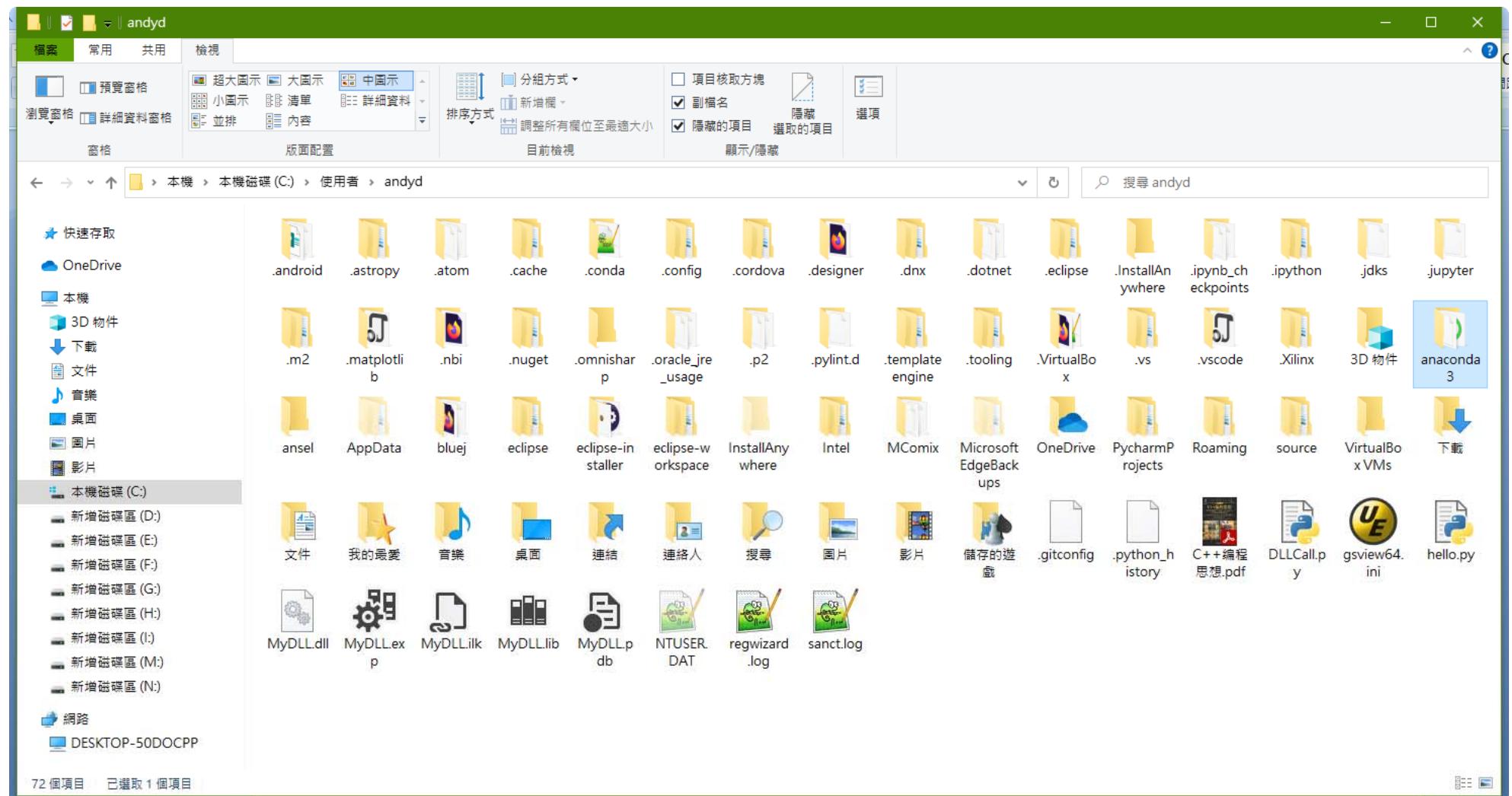
C:\Users\andyd 的目錄

[.] [..] [.android]
[.astropy] [.atom] [.cache]
[.conda] [.config] [.cordova]
[.designer] [.dnx] [.dotnet]
[.eclipse] .gitconfig [.InstallAnywhere]
[.ipynb_checkpoints] [.ipython] [.jdk]
[.jupyter] [.m2] [.matplotlib]
[.nbi] [.nuget] [.omnisharp]
[.oracle_jre_usage] [.p2] [.pylint.d]
.python_history [.templateengine] [.tooling]
[.VirtualBox] [.vs] [.vscode]
[.Xilinx] [3D Objects] [anaconda3]
[ansel] [bluej] [Contacts]
C_ 編程思想 (两卷合订本).pdf [Desktop] DLLCall.py
[Documents] [Downloads] [eclipse]
[eclipse-installer] [eclipse-workspace] [Favorites]
gsview64.ini hello.py [Intel]
[Links] [MComix] [Music]
MyDLL.dll MyDLL.exp MyDLL.ilk
MyDLL.lib MyDLL.pdb [OneDrive]
[Pictures] [PycharmProjects] regwizard.log
[Roaming] sanct.log [Saved Games]
[Searches] [source] [Videos]
[VirtualBox VMs]

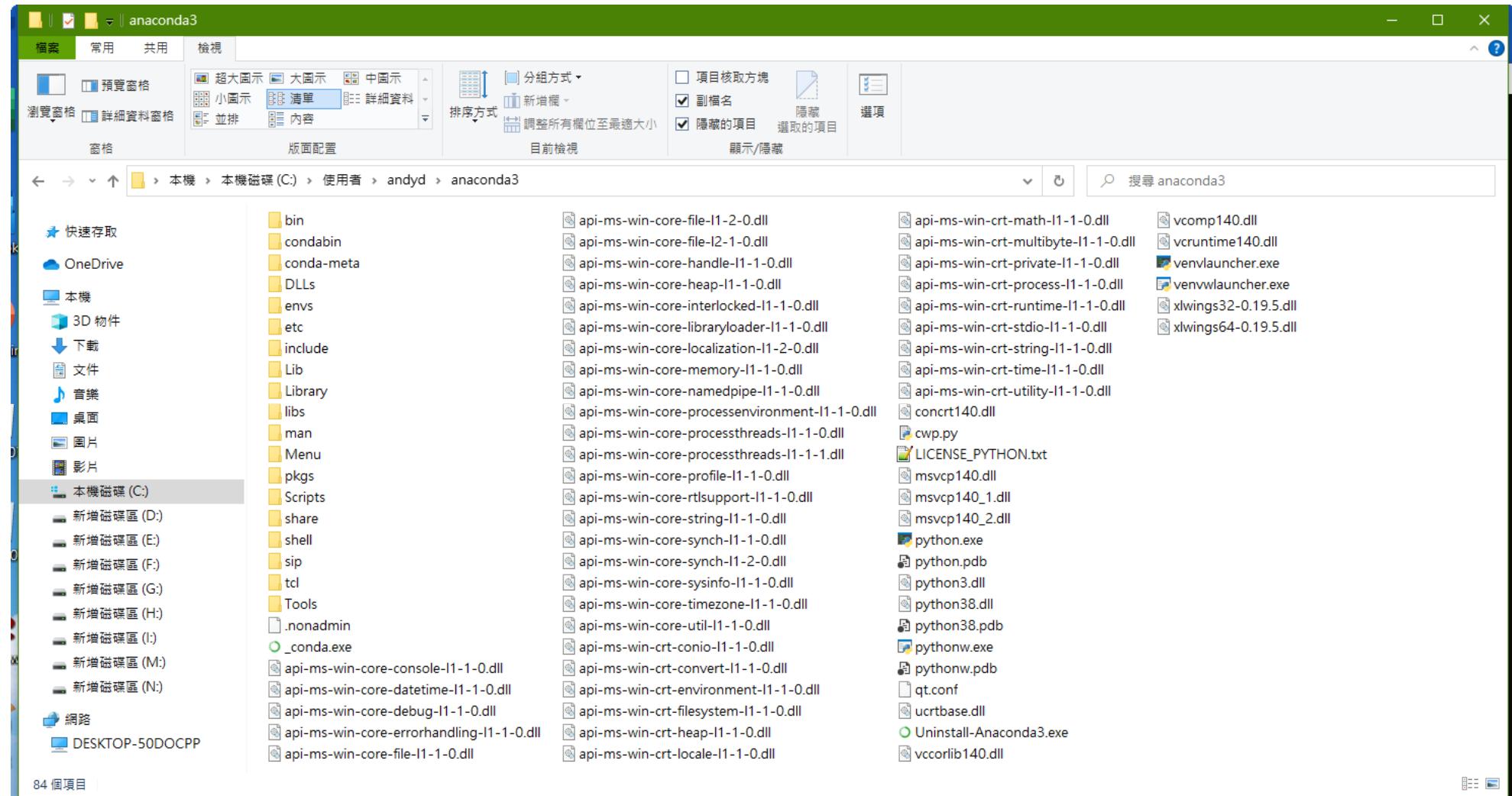
13 個檔案 62,695,322 位元組
57 個目錄 485,859,225,600 位元組可用

(base) C:\Users\andyd>
```

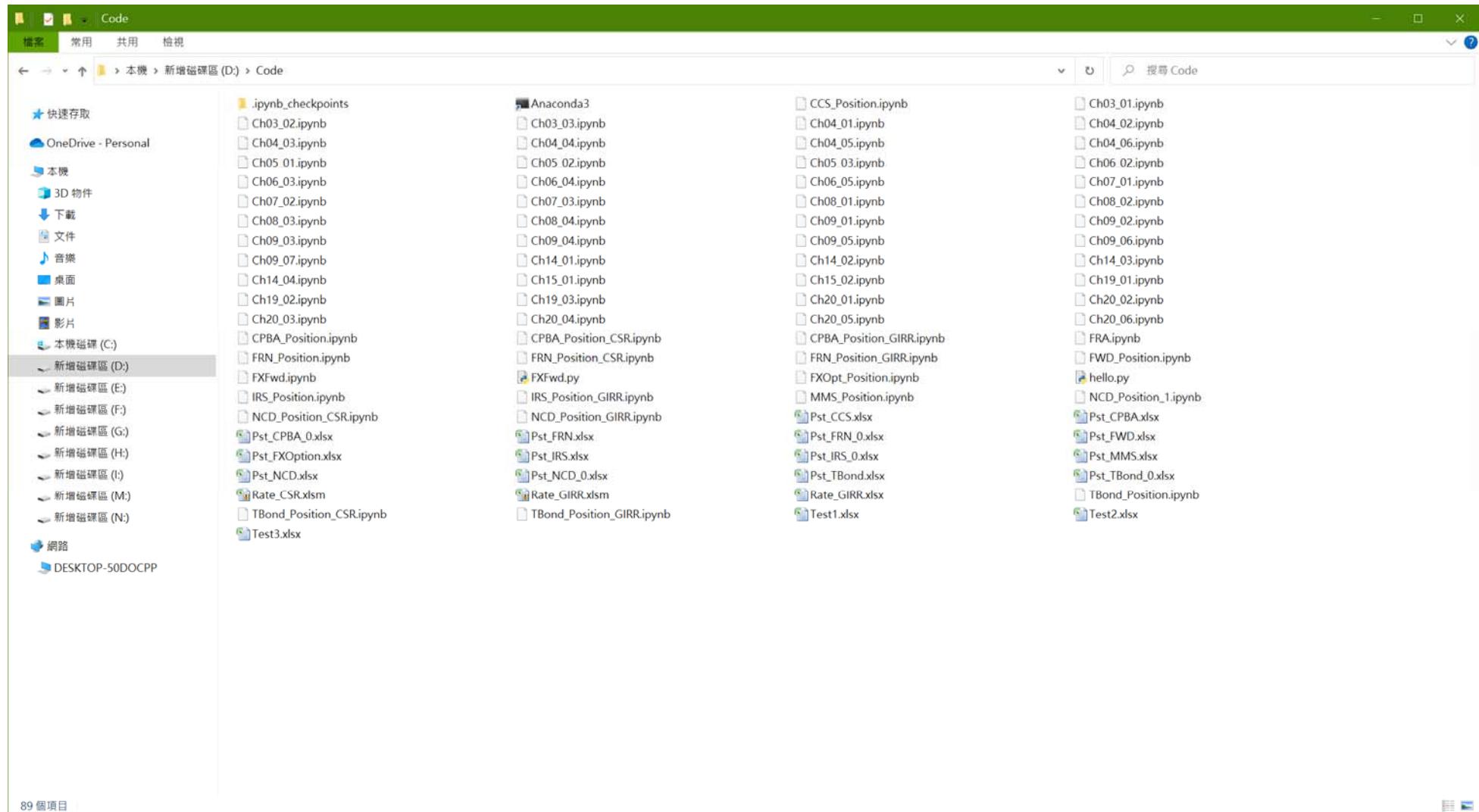
◆ 預設執行位址



➤ Anaconda 安裝的位址



◆ 將程式放在 D:\Code 目錄下，



◆ 進入 Notebook

```
(base) C:\Users\andyd>dir/w
磁碟區 C 中的磁碟沒有標籤。
磁碟區序號： 183C-35D4

C:\Users\andyd 的目錄

[.] [..] [.android] [.astropy] [.atom]
[.cache] [.conda] [.condarc] [.config] [.cordova]
[.designer] [.dnx] [.dotnet] [.eclipse] [.gitconfig]
[.InstallAnywhere] [.ipynb_checkpoints] [.ipython] [.jdks] [.jupyter]
[.m2] [.matplotlib] [.mozilla] [.nbi] [.nuget]
[.omnisharp] [.oracle_jre_usage] [.p2] [.pylint.d]
[.templateengine] [.tooling] [.VirtualBox] [.vs]
[.Xilinx] [3D Objects] [.anaconda3] [.ansel]
C++編程思想.pdf [Contacts] CudaDLLCall.ipynb CudaLib.lib [.vscode]
DLLCall.py [Documents] [Downloads] [dwhelper]
[eclipse-installer] [eclipse-workspace] [Elvis] [Favorites]
gsview64.ini hello.py [Intel] [Links]
[Music] MyDLL.lib [OneDrive] [Pecu_QL_Python]
Pricing the Future.pdf [PycharmProjects] PyCUDA_Test.ipynb PyQt5 中文教程.pdf [.Pictures]
regwizard.log RNG.ipynb [Roaming] sanct.log ReadYahooData.ipynb
[Searches] [source] stock_id.csv [Saved Games]
[Videos] [VirtualBox VMs] [workspace] VanillaOpt.ipynb
VanillaSwap.ipynb

21 個檔案 66,238,478 位元組
62 個目錄 380,336,058,368 位元組可用

(base) C:\Users\andyd>
```

➤ 切換到 notebook 目錄下，進入 jupyter notebook。

(base) C:\Users\andyd>D:
(base) D:\>cd Code
(base) D:\Code>dir /w
磁碟區 D 中的磁碟是 新增磁碟區
磁碟區序號: A624-91C0
D:\Code 的目錄

[.]	[.]	[.ipyynb_checkpoints]	Anaconda3.lnk
CCS_Position.ipynb	Ch03_01.ipynb	Ch03_02.ipynb	Ch03_03.ipynb
Ch04_01.ipynb	Ch04_02.ipynb	Ch04_03.ipynb	Ch04_04.ipynb
Ch04_05.ipynb	Ch04_06.ipynb	Ch05_01.ipynb	Ch05_02.ipynb
Ch05_03.ipynb	Ch06_02.ipynb	Ch06_03.ipynb	Ch06_04.ipynb
Ch06_05.ipynb	Ch07_01.ipynb	Ch07_02.ipynb	Ch07_03.ipynb
Ch08_01.ipynb	Ch08_02.ipynb	Ch08_03.ipynb	Ch08_04.ipynb
Ch09_01.ipynb	Ch09_02.ipynb	Ch09_03.ipynb	Ch09_04.ipynb
Ch09_05.ipynb	Ch09_06.ipynb	Ch09_07.ipynb	Ch14_01.ipynb
Ch14_02.ipynb	Ch14_03.ipynb	Ch14_04.ipynb	Ch15_01.ipynb
Ch15_02.ipynb	Ch19_01.ipynb	Ch19_02.ipynb	Ch19_03.ipynb
Ch20_01.ipynb	Ch20_02.ipynb	Ch20_03.ipynb	Ch20_04.ipynb
Ch20_05.ipynb	Ch20_06.ipynb	CPBA_Position.ipynb	CPBA_Position_CSR.ipynb
CPBA_Position_GIRR.ipynb	FRA.ipynb	FRN_Position.ipynb	FRN_Position_CSR.ipynb
FRN_Position_GIRR.ipynb	FWD_Position.ipynb	FXFwd.ipynb	FXFwd.py
FXOpt_Position.ipynb	hello.py	IRS_Position.ipynb	IRS_Position_GIRR.ipynb
MMS_Position.ipynb	NCD_Position_1.ipynb	NCD_Position_CSR.ipynb	NCD_Position_GIRR.ipynb
Pst_CCS.xlsx	Pst_CPBA.xlsx	Pst_CPBA_0.xlsx	Pst_FRN.xlsx
Pst_FRN_0.xlsx	Pst_FWD.xlsx	Pst_FXOption.xlsx	Pst_IRS.xlsx
Pst_IRS_0.xlsx	Pst_MMS.xlsx	Pst_NCD.xlsx	Pst_NCD_0.xlsx
Pst_TBond.xlsx	Pst_TBond_0.xlsx	Rate_CSR.xlsx	Rate_GIRR.xlsx
Rate_GIRR.xlsx	TBond_Position.ipynb	TBond_Position_CSR.ipynb	TBond_Position_GIRR.ipynb
Test1.xlsx	Test2.xlsx	Test3.xlsx	

88 個檔案 900,240 位元組
3 個目錄 1,690,526,212,096 位元組可用

(base) D:\Code>jupyter notebook

```
Anaconda3 - jupyter notebook

(base) D:\Code>jupyter notebook
Fail to get yarn configuration. C:\Users\andyd\Anaconda3\lib\site-packages\jupyterlab\staging\yarn.js:357
let buildActionsForCopy = (() => {
^^^

SyntaxError: Unexpected strict mode reserved word
    at exports.runInThisContext (vm.js:73:16)
    at Module._compile (module.js:443:25)
    at Object.Module._extensions..js (module.js:478:10)
    at Module.load (module.js:355:32)
    at Function.Module._load (module.js:310:12)
    at Function.Module.runMain (module.js:501:10)
    at startup (node.js:129:16)
    at node.js:814:3

[I 22:33:29.983 NotebookApp] JupyterLab extension loaded from C:\Users\andyd\Anaconda3\lib\site-packages\jupyterlab
[I 22:33:29.988 NotebookApp] JupyterLab application directory is C:\Users\andyd\Anaconda3\share\jupyter\lab
[I 22:33:29.994 NotebookApp] Serving notebooks from local directory: D:\Code
[I 22:33:29.995 NotebookApp] The Jupyter Notebook is running at:
[I 22:33:29.995 NotebookApp] http://localhost:8888/?token=5ffc0e2cc5bfeac583f74b84368bd5e182f541948e3cce7
[I 22:33:29.996 NotebookApp] or http://127.0.0.1:8888/?token=5ffc0e2cc5bfeac583f74b84368bd5e182f541948e3cce7
[I 22:33:29.997 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 22:33:30.196 NotebookApp]

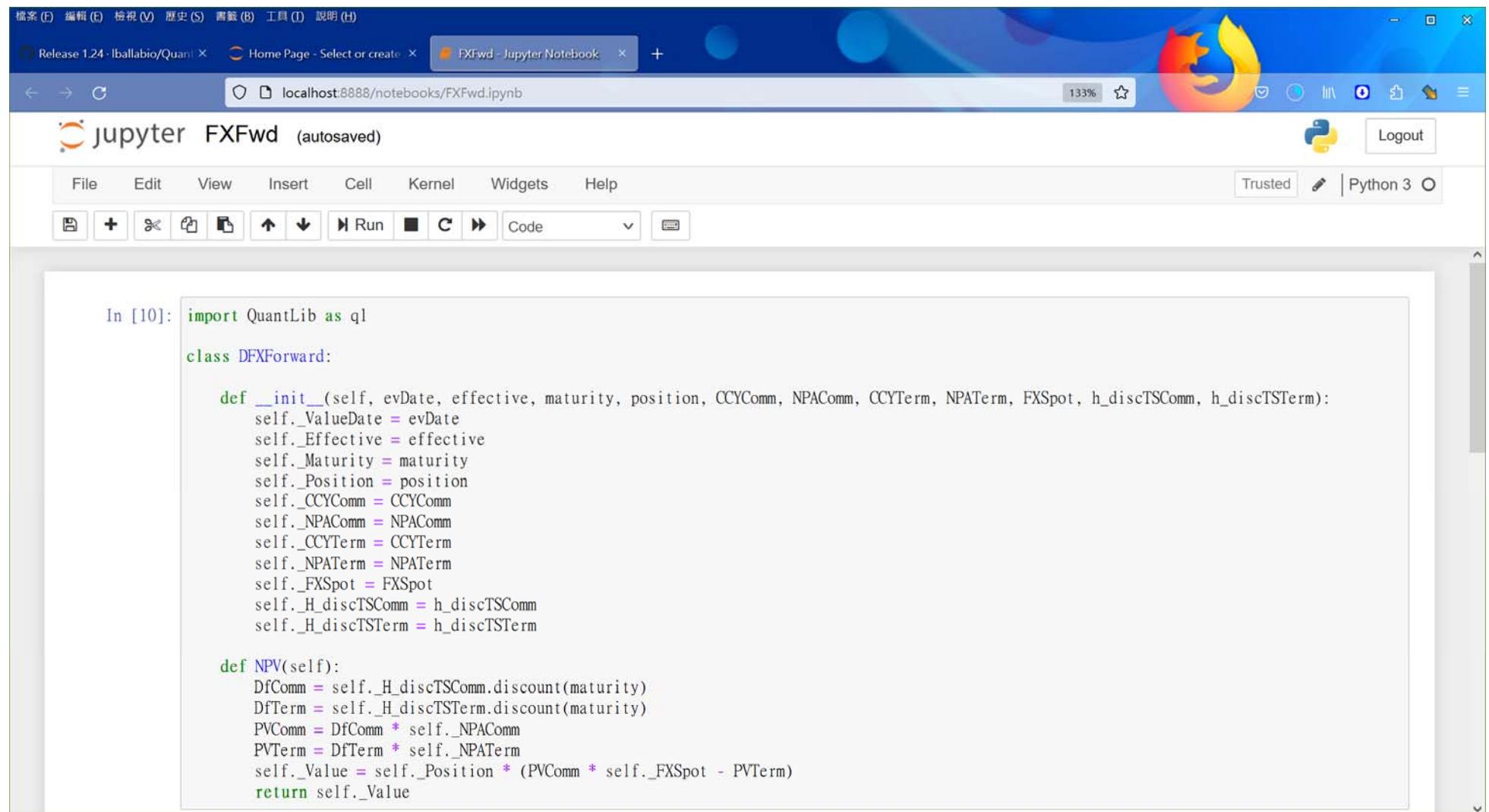
To access the notebook, open this file in a browser:
    file:///C:/Users/andyd/AppData/Roaming/jupyter/runtime/nbserver-8036-open.html
Or copy and paste one of these URLs:
    http://localhost:8888/?token=5ffc0e2cc5bfeac583f74b84368bd5e182f541948e3cce7
    or http://127.0.0.1:8888/?token=5ffc0e2cc5bfeac583f74b84368bd5e182f541948e3cce7
```

➤ 出現 Brower Notebook 畫面

The screenshot shows a Jupyter Notebook interface running in a web browser. The title bar indicates the URL is `localhost:8888/tree`. The main area displays a list of Jupyter Notebooks (ipynb files) in a table format. The columns are labeled 'Name', 'Last Modified', and 'File size'. The files listed are:

	Name	Last Modified	File size
<input type="checkbox"/>	CCS_Position.ipynb	1 個月前	15.9 kB
<input type="checkbox"/>	Ch03_01.ipynb	1 個月前	1.56 kB
<input type="checkbox"/>	Ch03_02.ipynb	1 個月前	4.5 kB
<input type="checkbox"/>	Ch03_03.ipynb	1 個月前	1.8 kB
<input type="checkbox"/>	Ch04_01.ipynb	1 個月前	8.08 kB
<input type="checkbox"/>	Ch04_02.ipynb	1 個月前	5.93 kB
<input type="checkbox"/>	Ch04_03.ipynb	1 個月前	2.11 kB
<input type="checkbox"/>	Ch04_04.ipynb	1 個月前	2.94 kB
<input type="checkbox"/>	Ch04_05.ipynb	1 個月前	1.07 kB
<input type="checkbox"/>	Ch04_06.ipynb	1 個月前	3.21 kB
<input type="checkbox"/>	Ch05_01.ipynb	1 個月前	1.49 kB
<input type="checkbox"/>	Ch05_02.ipynb	1 個月前	1.49 kB
<input type="checkbox"/>	Ch05_03.ipynb	1 個月前	3.05 kB
<input type="checkbox"/>	Ch06_02.ipynb	1 個月前	4.21 kB
<input type="checkbox"/>	Ch06_03.ipynb	1 個月前	1.95 kB
<input type="checkbox"/>	Ch06_04.ipynb	1 個月前	2.96 kB
<input type="checkbox"/>	Ch06_05.ipynb	7 個月前	2.13 kB
<input type="checkbox"/>	Ch07_01.ipynb	1 個月前	3.12 kB

◆ 點選 FXFwd.ipynb 檔案，可以開啟。



In [10]:

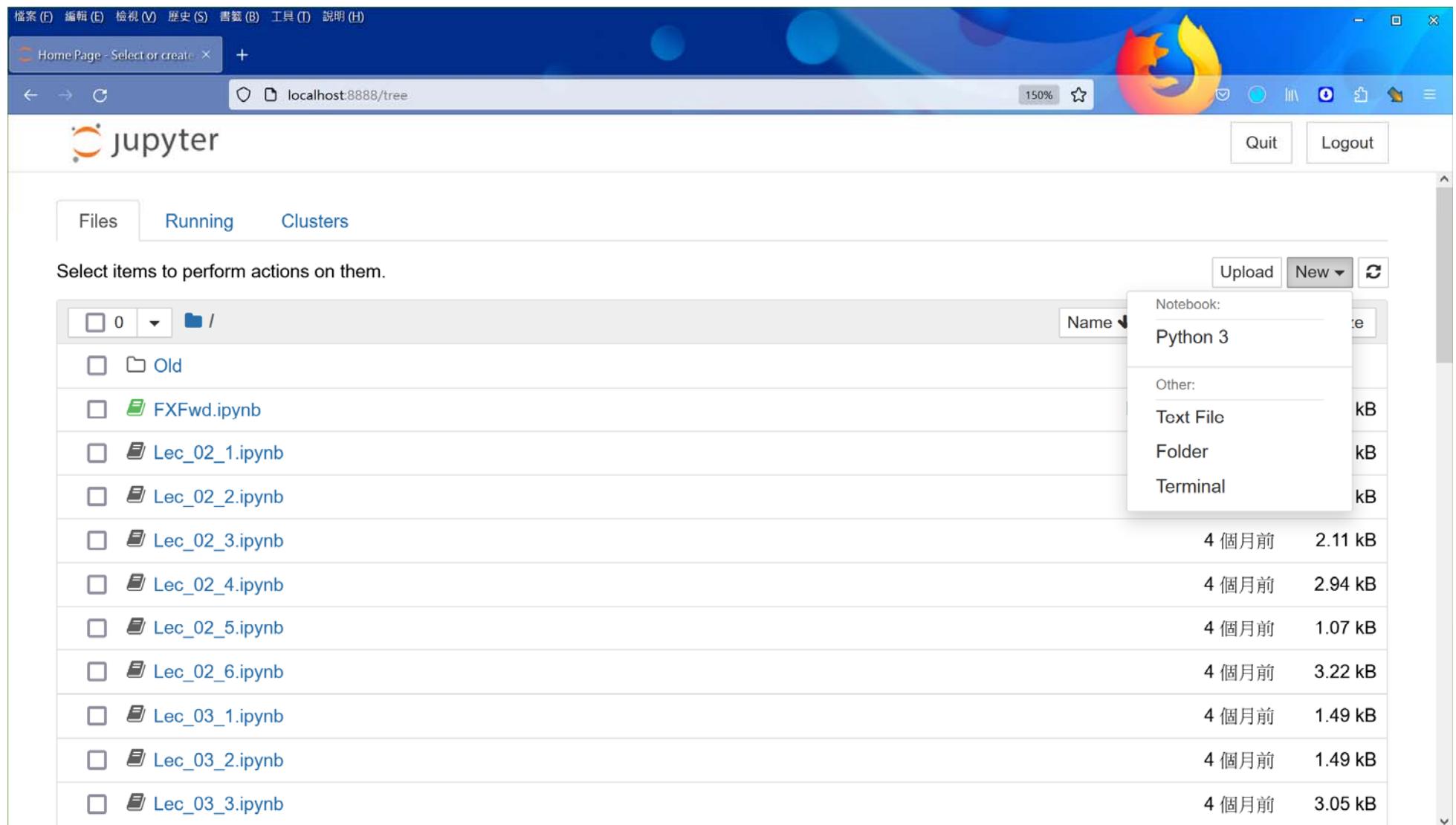
```
import QuantLib as ql

class DFXForward:

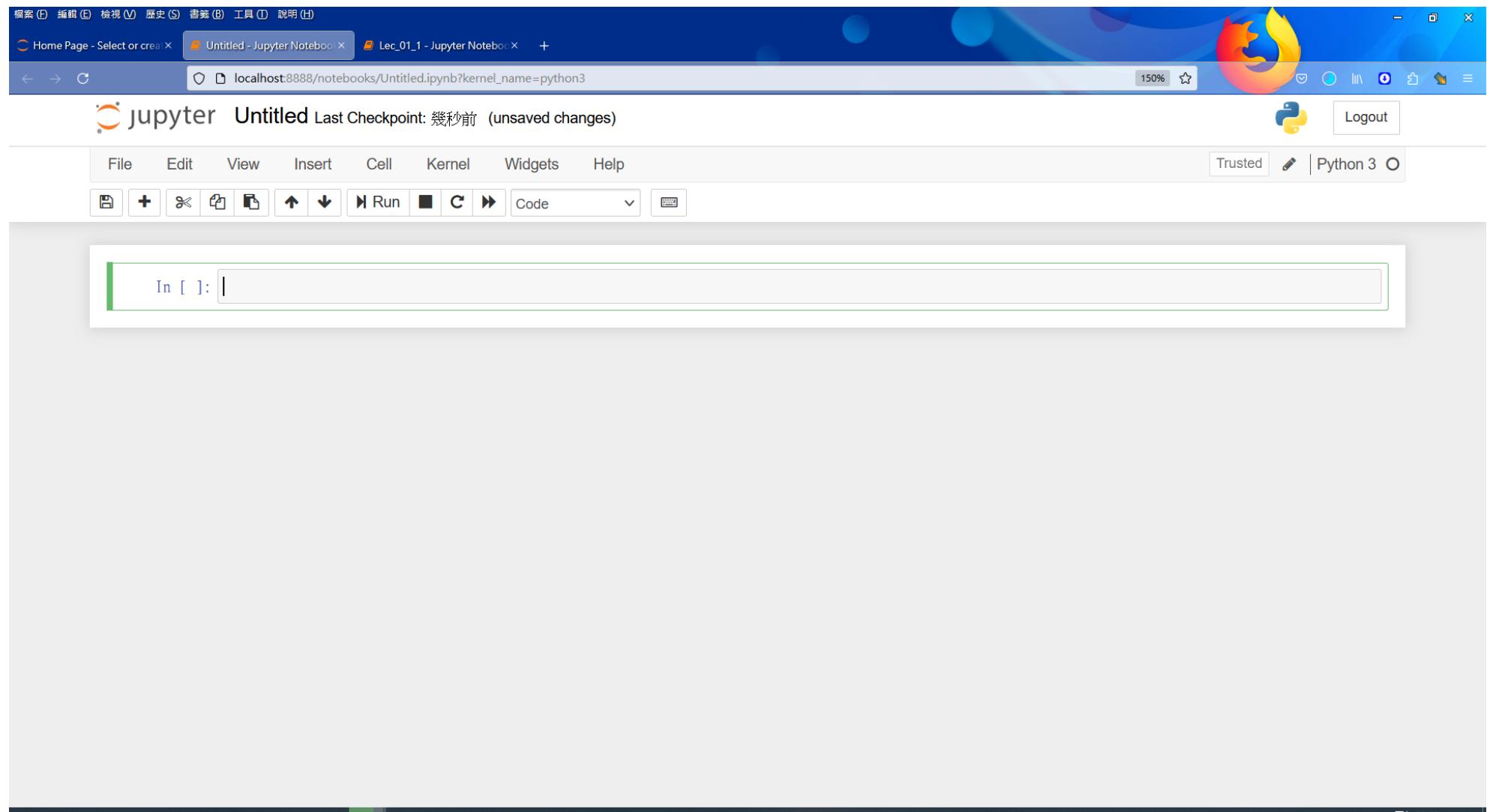
    def __init__(self, evDate, effective, maturity, position, CCYComm, NPAComm, CCYTerm, NPATerm, FXSpot, h_discTSComm, h_discTSTerm):
        self._ValueDate = evDate
        self._Effective = effective
        self._Maturity = maturity
        self._Position = position
        self._CCYComm = CCYComm
        self._NPAComm = NPAComm
        self._CCYTerm = CCYTerm
        self._NPATerm = NPATerm
        self._FXSpot = FXSpot
        self._H_discTSComm = h_discTSComm
        self._H_discTSTerm = h_discTSTerm

    def NPV(self):
        DfComm = self._H_discTSComm.discount(maturity)
        DfTerm = self._H_discTSTerm.discount(maturity)
        PVComm = DfComm * self._NPAComm
        PVTerm = DfTerm * self._NPATerm
        self._Value = self._Position * (PVComm * self._FXSpot - PVTerm)
        return self._Value
```

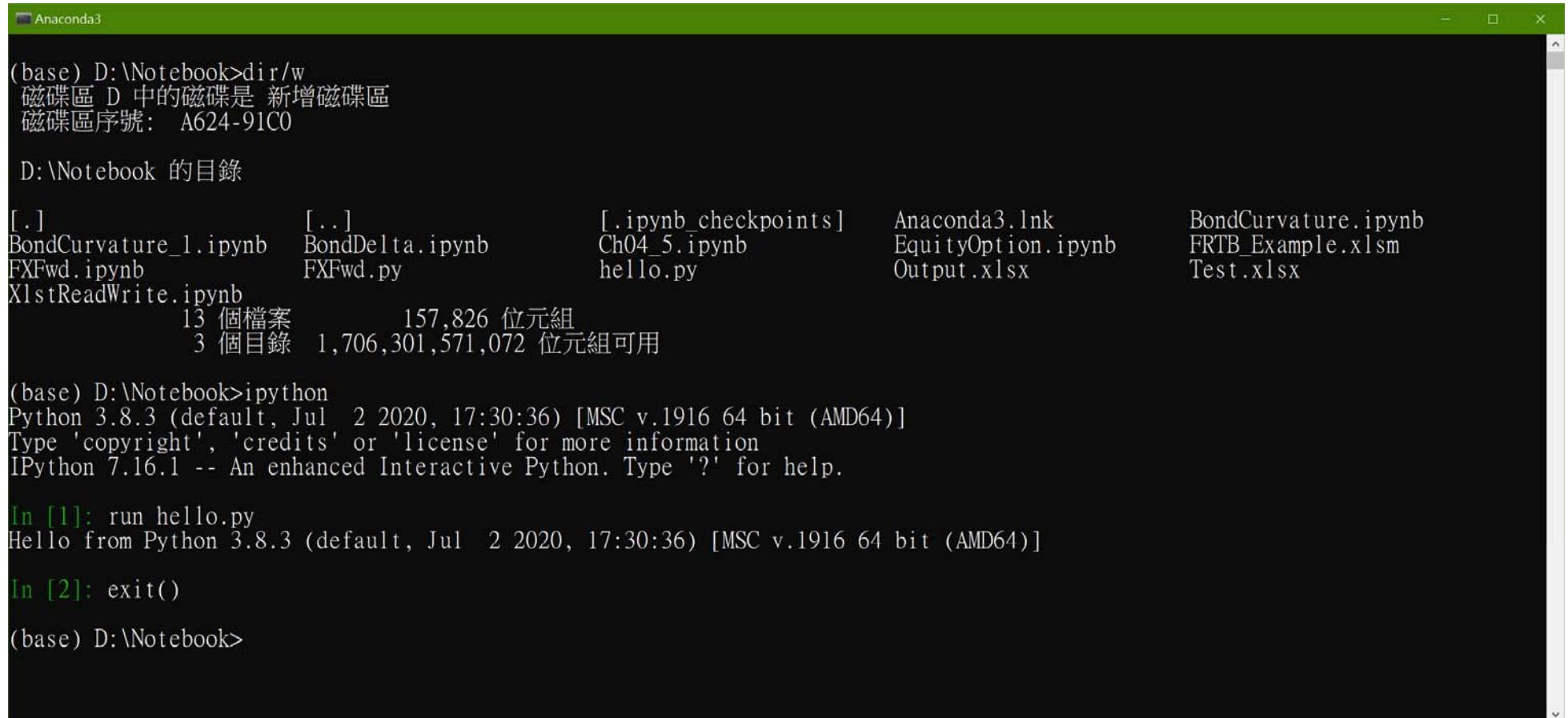
◆ 點選右上方，New → Python 3，空白 Notebook，開始輸入程式碼。



◆ Shift + Enter , 執行程式。File → Rename , 重新命名。時常存檔。



◆ 使用 ipython ,



(base) D:\Notebook>dir/w
磁碟區 D 中的磁碟是 新增磁碟區
磁碟區序號: A624-91C0

D:\Notebook 的目錄

[.]	[..]	[.ipynb_checkpoints]	Anaconda3.lnk	BondCurvature.ipynb
BondCurvature_1.ipynb	BondDelta.ipynb	Ch04_5.ipynb	EquityOption.ipynb	FRTB_Example.xlsm
FXFwd.ipynb	FXFwd.py	hello.py	Output.xlsx	Test.xlsx
XlstReadWrite.ipynb				

13 個檔案 157,826 位元組
3 個目錄 1,706,301,571,072 位元組可用

(base) D:\Notebook>ipython
Python 3.8.3 (default, Jul 2 2020, 17:30:36) [MSC v.1916 64 bit (AMD64)]
Type 'copyright', 'credits' or 'license' for more information
IPython 7.16.1 -- An enhanced Interactive Python. Type '?' for help.

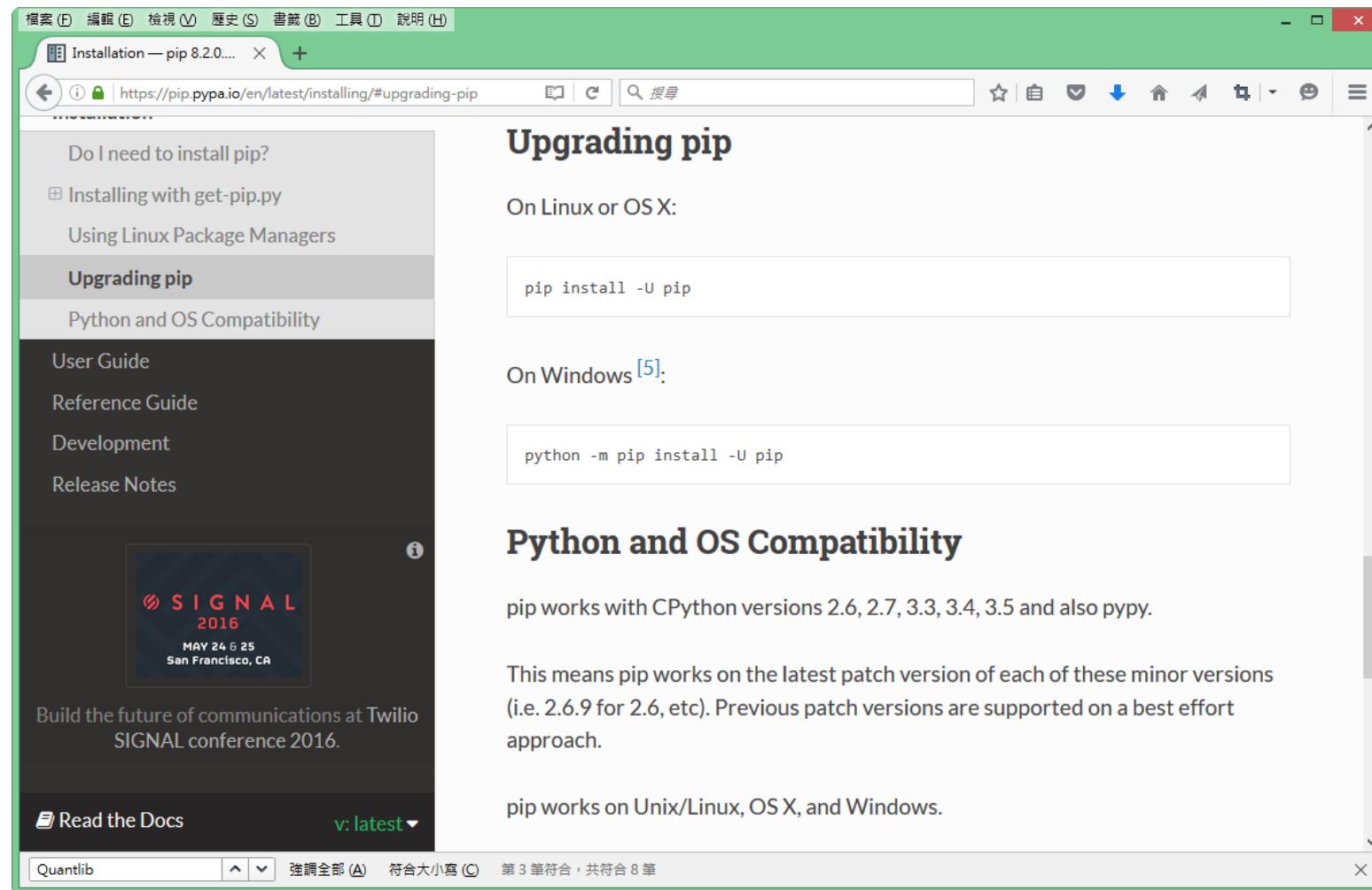
In [1]: run hello.py
Hello from Python 3.8.3 (default, Jul 2 2020, 17:30:36) [MSC v.1916 64 bit (AMD64)]

In [2]: exit()

(base) D:\Notebook>

◆ 升級 pip

C:\>python -m pip install -U pip

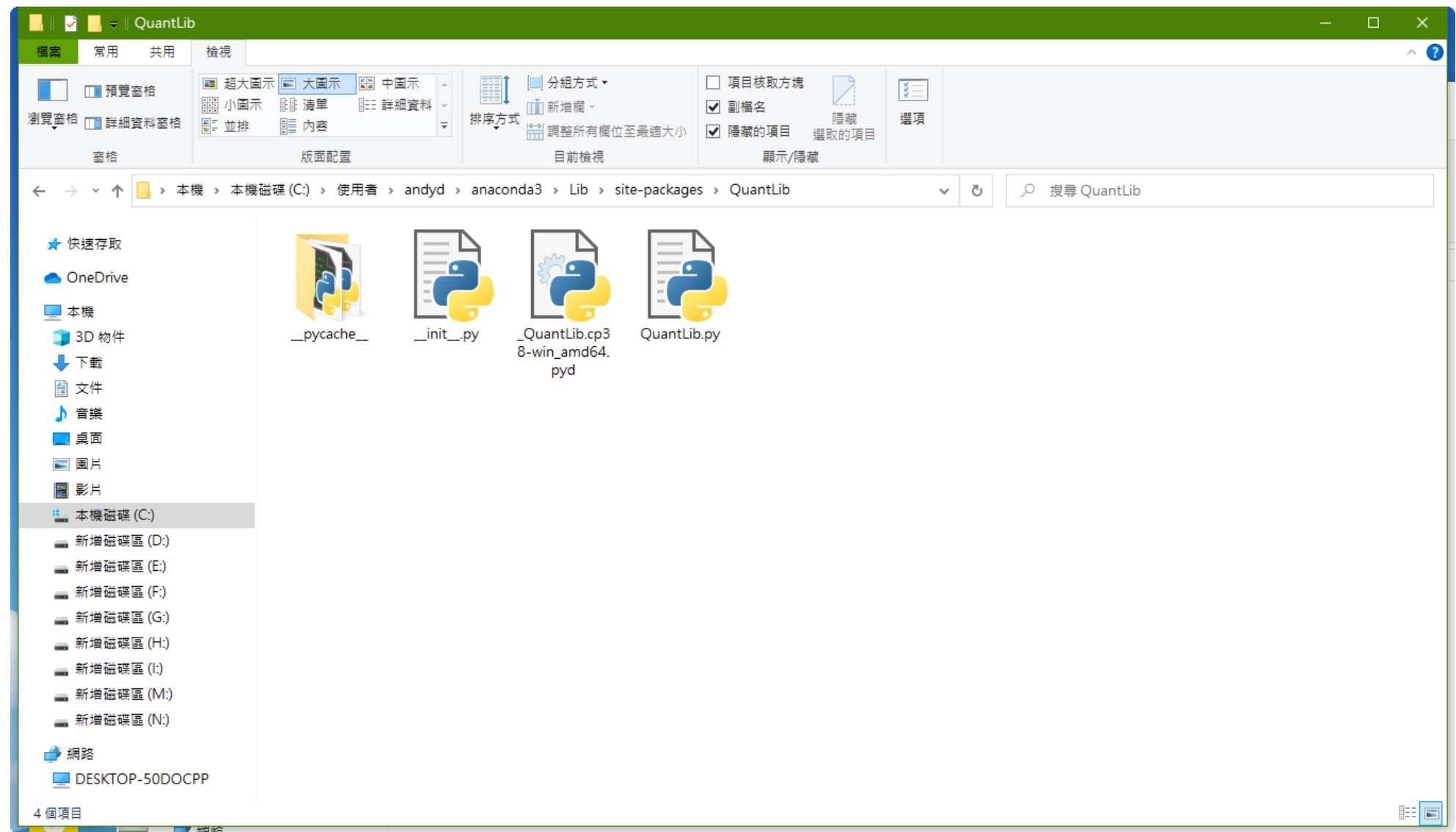


- ◆ 新版最簡單的安裝方式。

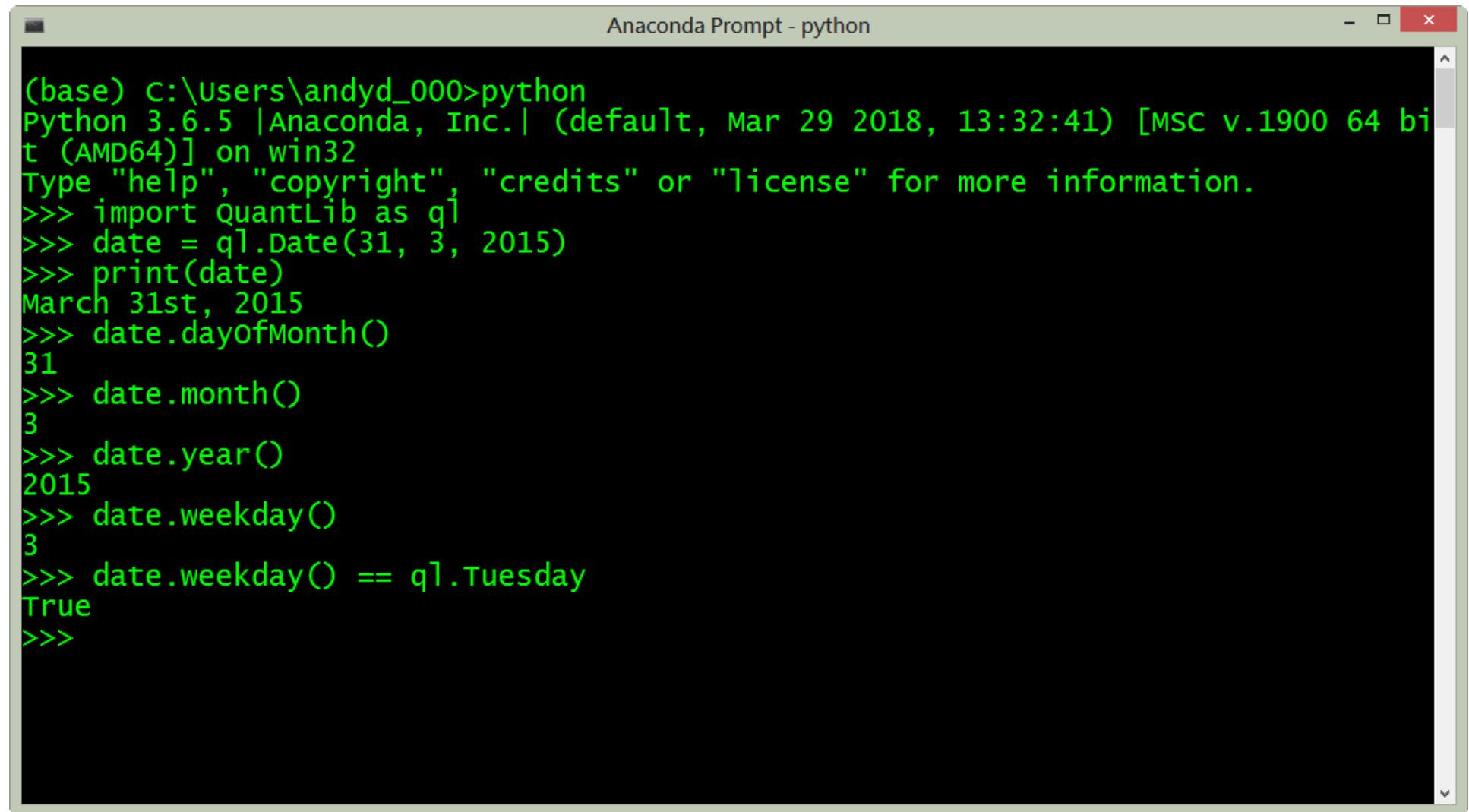
```
C:\pip install QuantLib
```

- ◆ 安裝舊版 QuantLib_Python Package，whl 格式。

```
C:\>pip install QuantLib_Python-1.11-cp36-cp36m-win_amd64.whl
```



四、QuantLib Package 使用與學習



Anaconda Prompt - python

```
(base) C:\users\andyd_000>python
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:32:41) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import QuantLib as ql
>>> date = ql.Date(31, 3, 2015)
>>> print(date)
March 31st, 2015
>>> date.dayOfMonth()
31
>>> date.month()
3
>>> date.year()
2015
>>> date.weekday()
3
>>> date.weekday() == ql.Tuesday
True
>>>
```

◆ 主要參考網站

The screenshot shows a web browser window with the following details:

- Header:** 檔案 (F) 編輯 (E) 檢視 (V) 歷史 (S) 書籤 (B) 工具 (I) 說明 (H)
- Title Bar:** Introduction to QuantLib Python
- Address Bar:** gouthamanbalaraman.com/blog/quantlib-basics.html
- Toolbar:** Back, Forward, Stop, Home, Refresh, Zoom (120%), More, Favorites, Help.
- Content Area:**
 - # Introduction to QuantLib Python
 - Published on March 24, 2015, by Goutham Balaraman.
 - Share options: Diaspora*, Twitter, Facebook, Google+, Email, Bloglovin.
 - A paragraph: "This post will walk through some of the basics of QuantLib Python library."
 - A link: "Visit here for other QuantLib Python examples. If you found these posts useful, please take a minute by providing some feedback."
 - A paragraph: "I installed the latest version of QuantLib (V1.5) and the python wrapper to QuantLib. My experiments lately have been to get a feel for the QuantLib API. The library itself is so extensive, that it is rather hard for a new comer to get going. In this post we will look into some of the basic classes and functionality in QuantLib."
 - A code snippet: "Let us import QuantLib as:

```
import QuantLib as ql
```
- Right Sidebar:**
 - About
 - A circular profile picture of Goutham Balaraman.
 - A bio: "I am Goutham Balaraman, and I explore topics in quantitative finance, programming, and data science. You can follow me @gsbalaraman."
 - A "Checkout my book" section with a thumbnail image of a book cover.

檔案 (F) 編輯 (E) 檢視 (V) 歷史 (S) 書籤 (B) 工具 (I) 說明 (H)

QuantLib Python Tutorials With Examples +

gouthamanbalaraman.com/blog/quantlib-python-tutorials-with-examples.html 120% About

G B

QuantLib Python Tutorials With Examples

October 30, 2015 by Gouthaman Balaraman

Share on: Diaspora* / Twitter / Facebook / Google+ / Email / Bloglovin

This post is a collection of links to all my quantlib python tutorial

Visit here for other [QuantLib Python examples](#). If you found these posts useful, please take a minute by providing some [feedback](#).

I have written a lot of little tutorials on using QuantLib python bindings. In these posts I explain some of the QuantLib concepts using minimal examples. Following are the links to these posts:

- [Introduction to QuantLib Python](#): This post will walk through some of the basics of QuantLib Python library.
- [Modeling Fixed Rate Bonds in QuantLib Python](#): This post will walk through an example of modeling fixed rate bonds using QuantLib Python.
- [An Introduction to Interest Rate Term Structure in QuantLib Python](#): This post will walk through the basics of bootstrapping yield curve in QuantLib Python.
- [Hull White Term Structure Simulations with QuantLib Python](#): Discusses simulation of the Hull



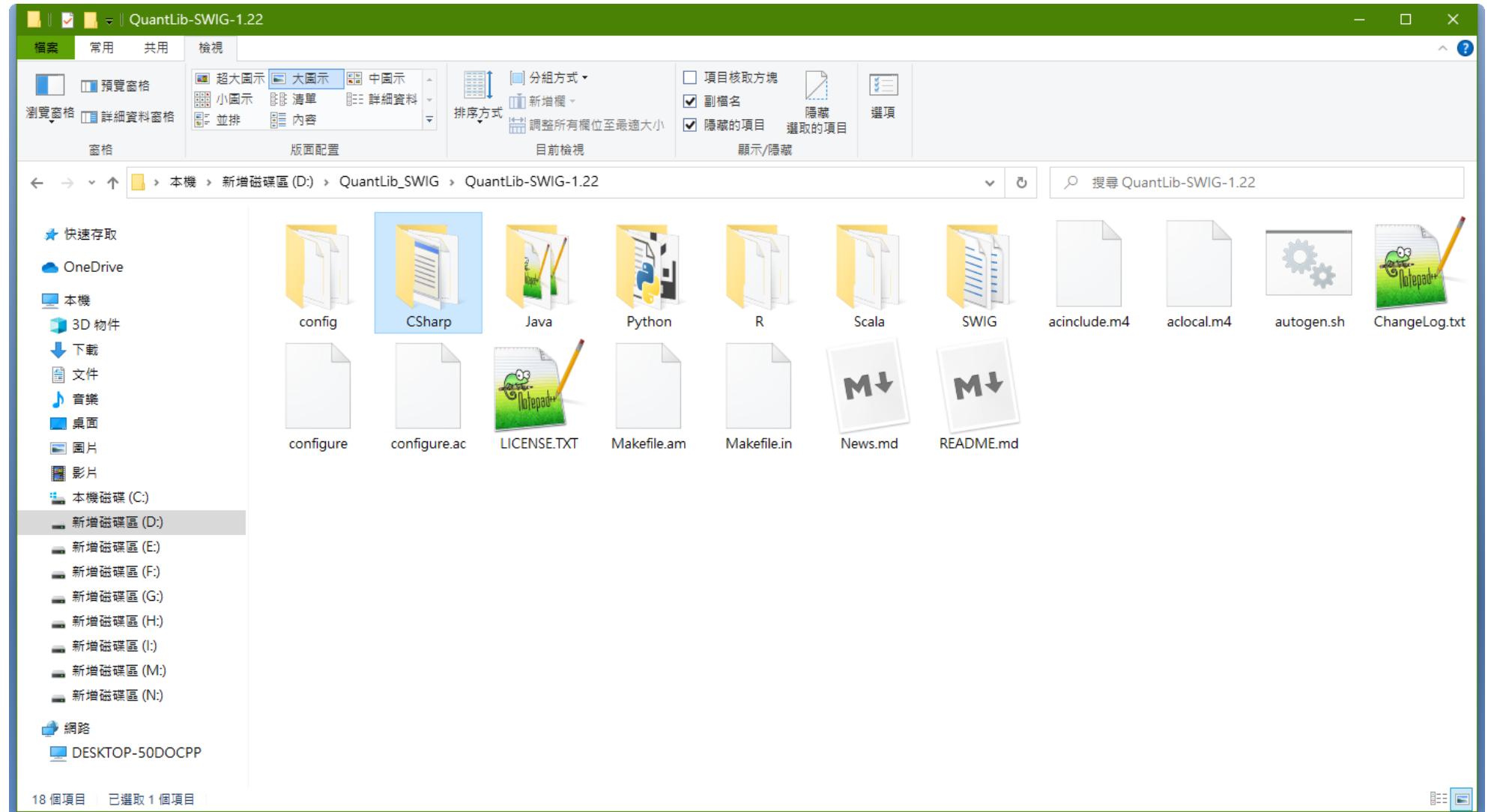
I am Goutham Balaraman, and I explore topics in quantitative finance, programming, and data science. You can follow me [@gsbalaraman](#).

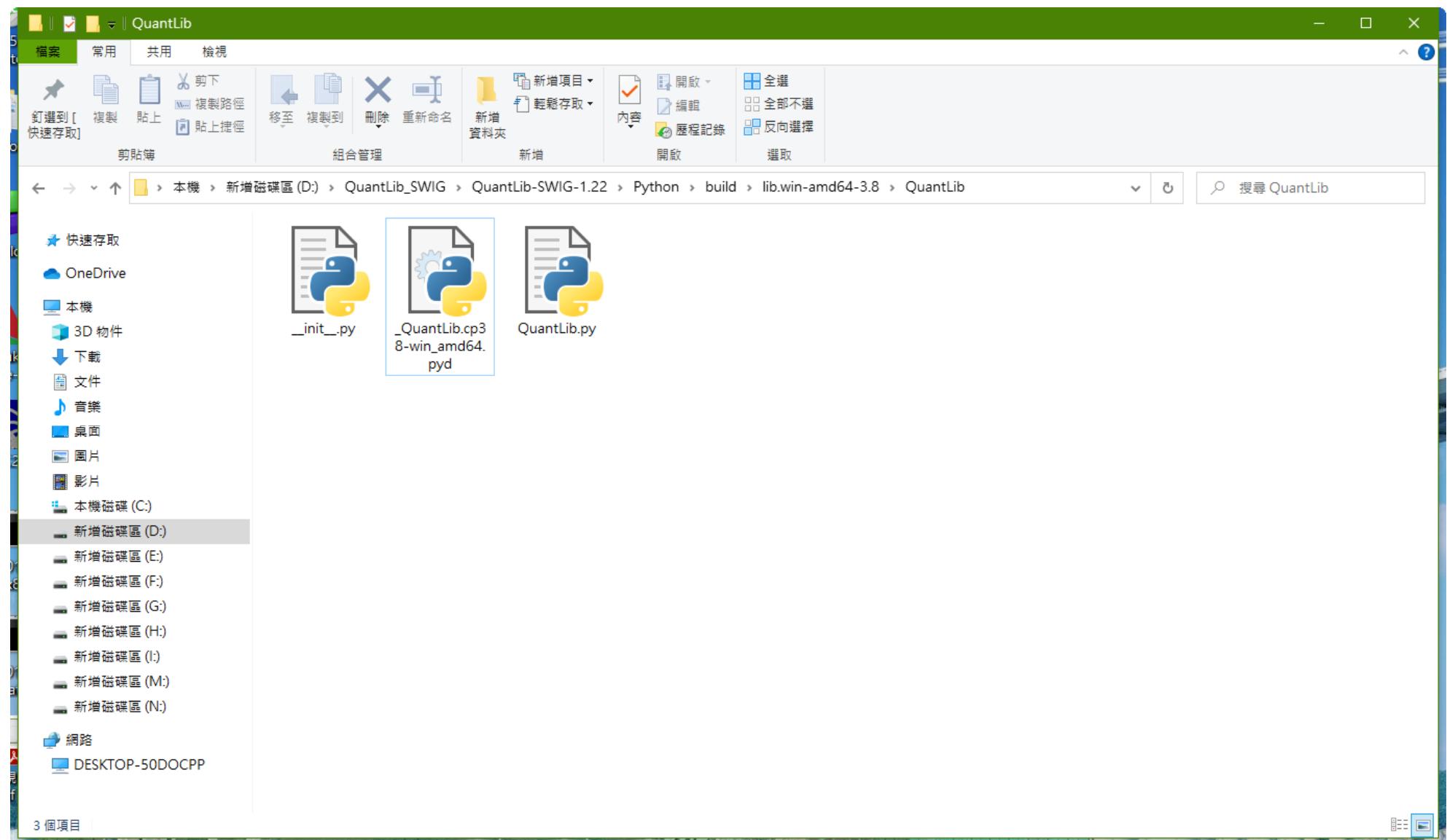
Checkout my book



(一)QuantLib Package 的學習方式

- ◆ Python Package 是使用 SWIG 工具，將 QuantLib C++ 程式庫加以包覆，成為 Python 可以使用的套件。
 - 此方法通用於 C#、Java、Python、R、Scala 等語言。
- ◆ 作者有自行編譯這些語言的呼叫程式庫。
 - C#、Java、Python 都沒問題。
 - 32 與 64 位元版本都有。
- ◆ 另外，QuantLibXL 是類似專案，作者也有自行編譯成功。
 - QuantLibXL 的文件寫得還好，可以參考。





◆ 基本上，這些使用 SWIG 產生的包覆程式庫，其物件與函數呼叫格式與功能相通。

- QuantLib 的 Python 套件，文件與功能查詢不方便
- 文件的不足是通病，必須以 QuantLib C++的文件來補充。
 - ✓ C++的程式碼是最好的說明文件。
 - ✓ C++的 HTML 文件也還可以。
- 早期的 C++文件有物件繼承圖，現在的文件沒有了。

◆ 1.22 版文件

QuantLib: Class List

https://www.quantlib.org/reference/annotated.html

QuantLib

A free/open-source library for quantitative finance

Reference manual - version 1.22

Main Page Related Pages Modules Classes Examples Search

Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

[detail level 1 2 3]

Category	Description
QuantLib	
detail	
initializers	
CashFlow	Base class for cash flows
AverageBMACoupon	Average BMA coupon
AverageBMALeg	Helper class building a sequence of average BMA coupons
CappedFlooredCoupon	Capped and/or floored floating-rate coupon
CappedFlooredYoYInflationCoupon	Capped or floored inflation coupon
CashFlows	cashflow-analysis functions
CmsCoupon	CMS coupon class

◆ 1.0 版文件

檔案(F) 編輯(E) 檢視(V) 歷史(S) 書籤(B) 工具(I) 說明(H)

QuantLib in other language X QuantLib download | Source X QuantLib: GeneralizedBlackScholesProcess X +

file:///D:/QuantLibCPP/docs_1_html/QuantLib-docs-1.0.html/class_quant_lib_1_1_generalized_black_scholes_process.html 133% ☆

QuantLib

A free/open-source library for quantitative finance

QuantLib::GeneralizedBlackScholesProcess

GeneralizedBlackScholesProcess Class Reference

[[Stochastic processes](#)]

Generalized Black-Scholes stochastic process. [More...](#)

```
#include <ql/processes/blackscholesprocess.hpp>
```

Inheritance diagram for GeneralizedBlackScholesProcess:

```
graph TD; StochasticProcess1D --> GeneralizedBlackScholesProcess; BlackScholesProcess --> GeneralizedBlackScholesProcess; ExtendedBlackScholesMertonProcess --> GeneralizedBlackScholesProcess; BlackProcess --> GeneralizedBlackScholesProcess; BlackScholesMertonProcess --> GeneralizedBlackScholesProcess;
```

[\[legend\]](#)

[List of all members.](#)

Version 1.0

Getting started

- [Introduction](#)
- [Where to get QuantLib](#)
- [Installation](#)
- [Configuration](#)
- [Usage](#)
- [Version history](#)
- [Additional resources](#)
- [The QuantLib group](#)
- [Copyright and license](#)

Reference manual

- [Modules](#)
- [Class Hierarchy](#)
- [Compound List](#)
- [File List](#)
- [Compound Members](#)
- [File Members](#)
- [Todo List](#)

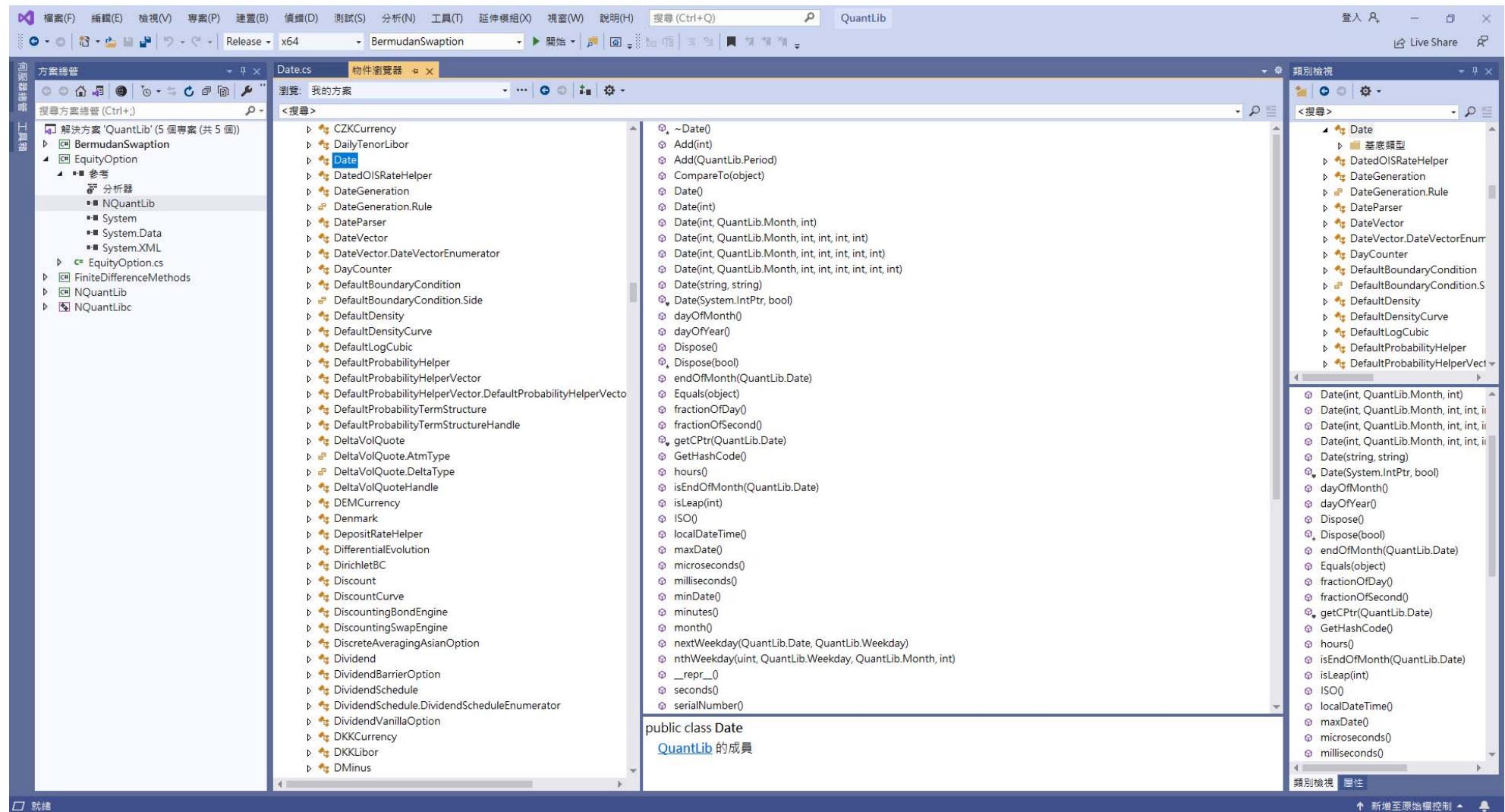
◆ 至於，Python 函數功能的查詢上，建議使用 C# 的 Visual Studio IDE 來補充。

➤ 以 Date 物件為例，IDE 中的 C# 專案可以透過類別檢視、或物件總管看到所有的屬性與方法。

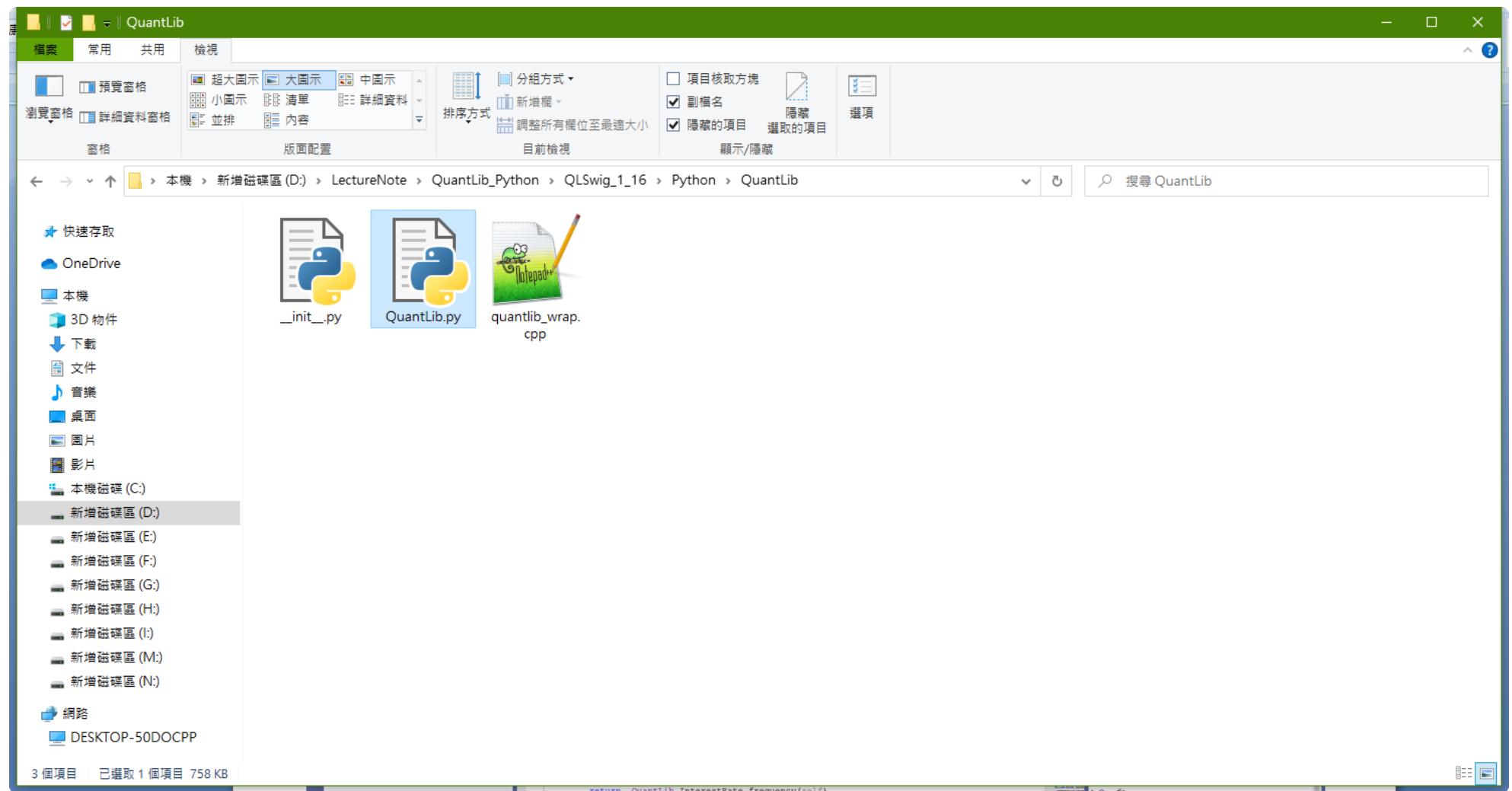
The screenshot shows the Visual Studio IDE interface with the following details:

- Solution Explorer:** Shows the project structure with a file named Date.cs under the Date folder.
- Date.cs Code Editor:** Displays the generated C# code for the Date class. The code includes constructor overloads, properties like swigCPtr, swigCMemOwn, and methods like Equals, GetHashCode, and Dispose.
- Output Window:** Shows build logs indicating successful compilation of the project.
- Class View (Category Explorer):** A floating window titled "類別檢視" (Category View) which lists all members of the Date class, such as Date(int), Date(string), dayOfMonth, dayOfYear, and various ToString and CompareTo implementations.

➤ 物件瀏覽器也可使用。



◆ 使用 Visual Studio 2019 打開 QuantLib.py 查詢，也很方便



➤ QuantLib.py 內容

The screenshot shows the Microsoft Visual Studio 2019 interface with the Python 3.8 (64-bit) development environment. The main window displays the `QuantLib.py` file, which contains Python code for interacting with the QuantLib C++ library via SWIG. The code defines a class `InterestRate` that wraps the `_QuantLib.InterestRate` C++ class. The `__init__` method calls `_QuantLib.InterestRate_swiginit`, and various properties and methods are defined to map to C++ functions like `rate`, `dayCounter`, `compounding`, etc. The `Object Browser` on the left shows the project structure and the current file. The `Category Explorer` on the right lists numerous C++ classes and functions from the QuantLib library, such as `SEKCurrency`, `SGDCurrency`, `Shibor`, `ShortRateModel`, `SimpleCashFlow`, `SimpleDayCounter`, `SimplePolynomialFitting`, `SimpleQuote`, `Simplex`, `SimpsonIntegral`, `Singapore`, and `SITCurrency`. The status bar at the bottom indicates the current line (行: 3395), character count (字元: 26), and file endings (SPC, CRLF).

```
def Actual365NoLeap():
    return _QuantLib.Actual365NoLeap()
Simple = _QuantLib.Simple
Compounded = _QuantLib.Compounded
Continuous = _QuantLib.Continuous
SimpleThenCompounded = _QuantLib.SimpleThenCompounded
CompoundedThenSimple = _QuantLib.CompoundedThenSimple
class InterestRate(object):
    __swig_setown = property(lambda x: x.this.setown(), lambda x, v: x.this.setown(v), doc='The membership function for interacting with the C++ object')
    __repr__ = __swig_repr__

    def __init__(self, *args):
        _QuantLib.InterestRate_swiginit(self, _QuantLib.new_InterestRate(*args))

    def rate(self):
        return _QuantLib.InterestRate_rate(self)

    def dayCounter(self):
        return _QuantLib.InterestRate_dayCounter(self)

    def compounding(self):
        return _QuantLib.InterestRate_compounding(self)

    def frequency(self):
        return _QuantLib.InterestRate_frequency(self)

    def discountFactor(self, *args):
        return _QuantLib.InterestRate_discountFactor(self, *args)

    def compoundFactor(self, *args):
        return _QuantLib.InterestRate_compoundFactor(self, *args)

    @staticmethod
    def impliedRate(*args):
        return _QuantLib.InterestRate_impliedRate(*args)

    def equivalentRate(self, *args):
        return _QuantLib.InterestRate_equivalentRate(self, *args)

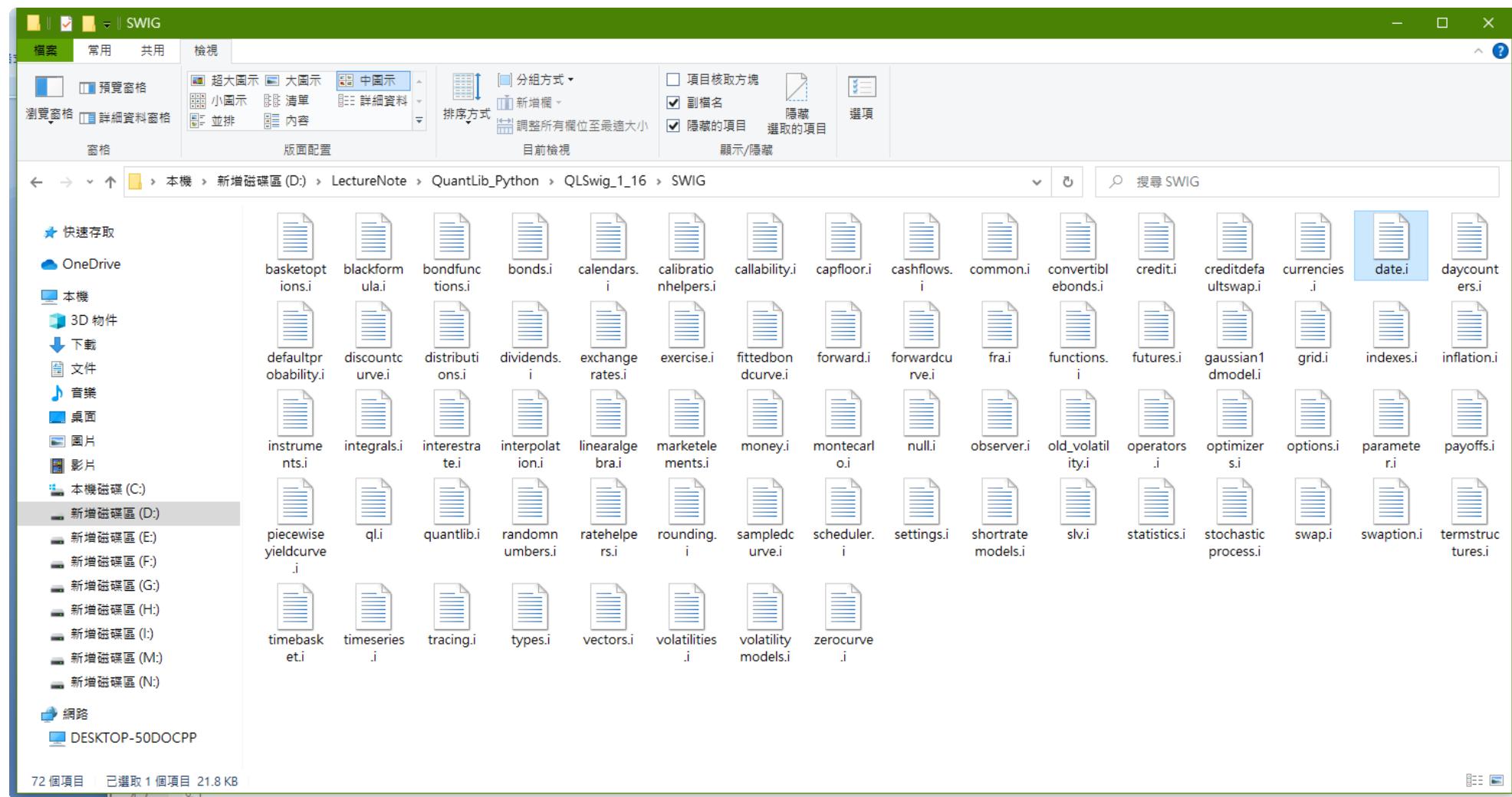
    def __str__(self):
        return _QuantLib.InterestRate__str__(self)
    __swig_destroy__ = _QuantLib.delete_InterestRate
```

◆ Java 版的 Eclipse IDE 介面也很方便使用查詢。

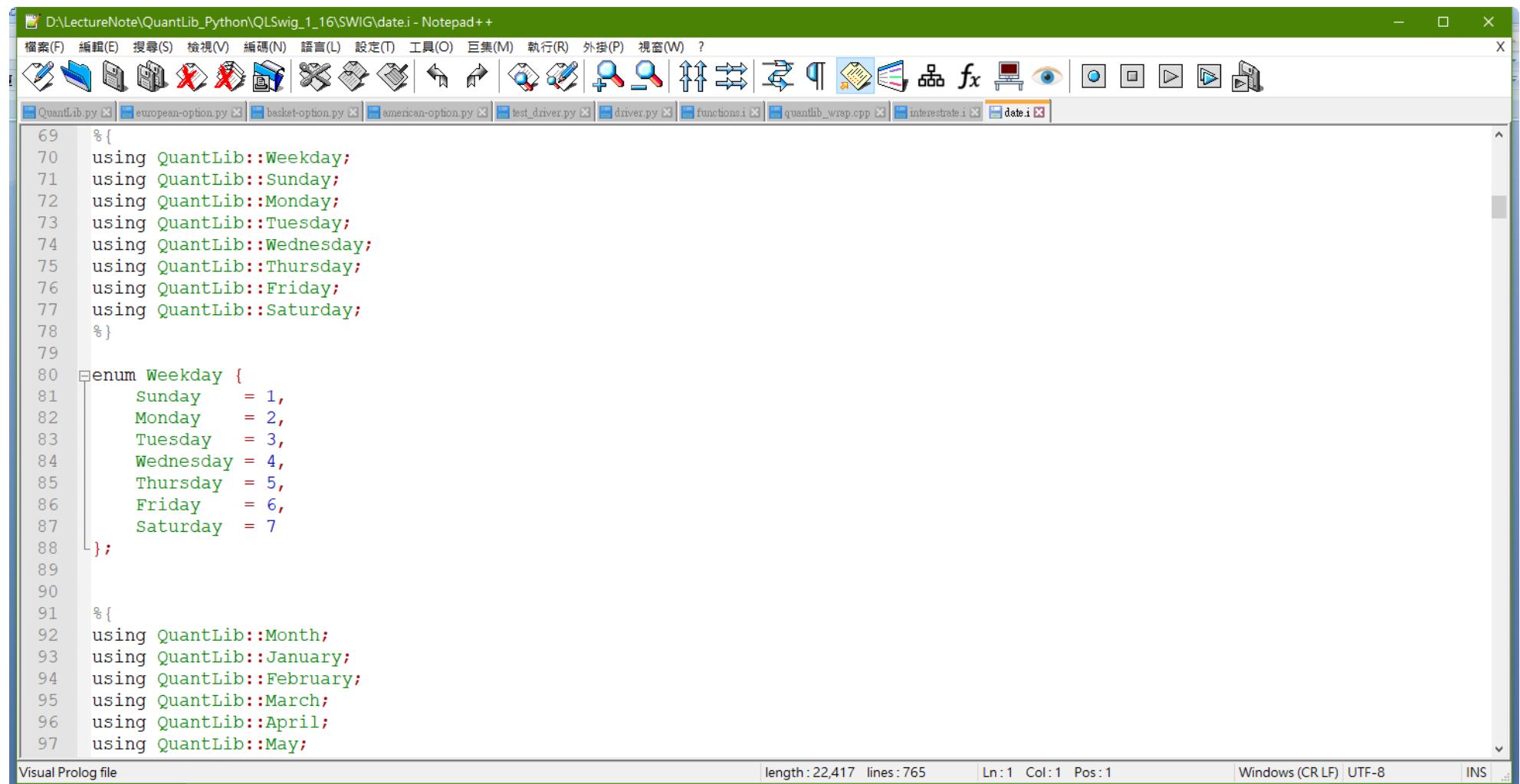
The screenshot shows the Eclipse IDE interface with the following components:

- Top Bar:** eclipse-workspace - QuantLib16/org/quantlib/Date.java - Eclipse IDE. Includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Left Sidebar:** Package Explorer view showing the project structure with Date.java selected.
- Center Editor:** Date.java code editor. The code is generated by SWIG and implements the Date class with various methods like getCPtr, isEndOfMonth, and Date constructors.
- Right Sidebar:** Outline view showing the class hierarchy and member list for Date.
- Bottom Status Bar:** Writable, Smart Insert, 2:1:81, Sync System PYTH...PATH: (100%), and other status indicators.

◆ SWIG 的檔案可以參考



➤ Date.i 文件檔



The screenshot shows the Notepad++ interface with the file "date.i" open. The window title is "D:\LectureNote\QuantLib_Python\QLSwig_1_16\SWIG\date.i - Notepad++". The menu bar includes 檔案(F), 編輯(E), 搜尋(S), 檢視(V), 編碼(N), 語言(L), 設定(T), 工具(O), 巨集(M), 執行(R), 外掛(P), 視窗(W), and ?.

The toolbar contains various icons for file operations like Open, Save, Find, Replace, and Print.

The tab bar shows multiple files: QuantLib.py, european-option.py, basket-option.py, american-option.py, test_driver.py, driver.py, functions.i, quantlib_wrap.cpp, interestrate.i, and date.i (the current file).

The code in the editor is:

```
69 %{
70 using QuantLib::Weekday;
71 using QuantLib::Sunday;
72 using QuantLib::Monday;
73 using QuantLib::Tuesday;
74 using QuantLib::Wednesday;
75 using QuantLib::Thursday;
76 using QuantLib::Friday;
77 using QuantLib::Saturday;
78 %}
79
80 enum Weekday {
81     Sunday      = 1,
82     Monday      = 2,
83     Tuesday     = 3,
84     Wednesday   = 4,
85     Thursday    = 5,
86     Friday      = 6,
87     Saturday    = 7
88 };
89
90
91 %{
92 using QuantLib::Month;
93 using QuantLib::January;
94 using QuantLib::February;
95 using QuantLib::March;
96 using QuantLib::April;
97 using QuantLib::May;
```

The status bar at the bottom displays "Visual Prolog file", "length : 22,417 lines : 765", "Ln : 1 Col : 1 Pos : 1", "Windows (CR LF) | UTF-8", and "INS".

◆ 另外一個補充文件來源為 QuantLibXL 專案。

The screenshot shows a Firefox browser window with the URL <https://www.quantlib.org/quantlibxl/>. The page is titled "QuantLibXL" and features an "Overview" section. On the left, there is a sidebar with navigation links for "Version 1.22.0" and "Functional Documentation". A green button on the right says "QuantLib XL Download Version 1.22 - 16MB". Below the download button is a screenshot of an Excel spreadsheet titled "SWAPTION" with data for Notional, Option Tenor, European Exercise Date, Swap Tenor, and Settlement Type.

QuantLibXL exports the functionality of the QuantLib C++ analytics library to Microsoft Excel.

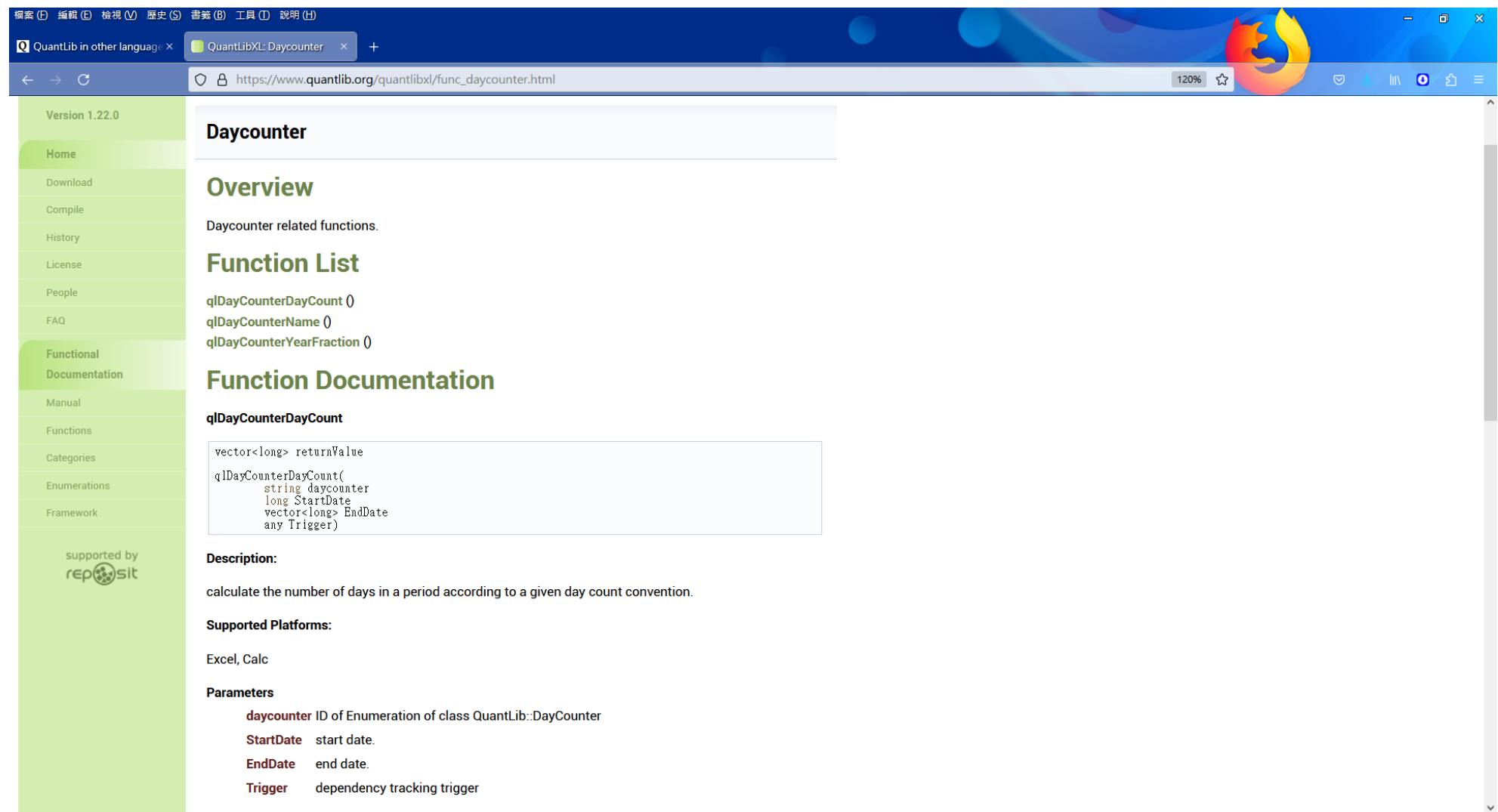
The project comprises

- A compiled Excel Addin (an XLL)
- End user documentation
- Example workbooks

The addin implements 1,114 functions in support of market data and pricing for a variety of instruments including bonds, options, swaps, swaptions, and caps/floors.

	A	B	D	G	I	J	K
1	SWAPTION						
2	Notional	1,000,000.00					
3	Option Tenor	2Y					
4	European Exercise Date	Mon, 21-Dec-2009					
5	Swap Tenor	5Y					
6	Settlement Type	Physical					

◆ Daycounter 物件的建立與功能



The screenshot shows a Firefox browser window with the following details:

- Address Bar:** https://www.quantlib.org/quantlibxl/func_daycounter.html
- Page Title:** Daycounter
- Content Area:**
 - Overview:** Daycounter related functions.
 - Function List:**
 - qlDayCounterDayCount()
 - qlDayCounterName()
 - qlDayCounterYearFraction()
 - Function Documentation:**
 - qlDayCounterDayCount**
 - Description:** calculate the number of days in a period according to a given day count convention.
 - Supported Platforms:** Excel, Calc
 - Parameters:**
 - daycounter** ID of Enumeration of class QuantLib::DayCounter
 - StartDate** start date.
 - EndDate** end date.
 - Trigger** dependency tracking trigger

◆ 作者曾經以 QuantLibXL 增益集，來介紹 QuantLib 的使用。

➤ 對於不是以 C++為開發工具的程序員，有其便利性

◆ 本課程的目的不是在教導 Python 的程式開發，而是教導學員利用 Python 去使用 QuantLib 程式庫。

➤ 財務功能的正確使用是重點，

✓ 強調財務知識勝於程式開發

➤ 由於財務功能是以財務理論為基礎，會提到必要的財務公式的使用，

✓ 但不以模型推導方式教導，重點在使用

➤ 很多的財務功能使用與市場交易慣例有關，

✓ 這些是 QuantLib 的強項，完全符合實務規範

✓ 實用的程式，對此要求非常高

➤ 當然，使用 QuantLib 所需的 Python 語法會說明的

✓ 主要是物件導向的程式語法

◆ 學習資訊：<https://www.facebook.com/groups/421949278879071>

The screenshot shows a Facebook group page titled "QuantLib-昀騰學習園地". The page has 426 members and is public. The main content is a guide titled "QuantLib-Python installation on Windows" by Luigi Bellabio. The guide discusses prerequisites and provides instructions for installing from PyPI or precompiled binary versions. To the right, there's a sidebar with sections for "Get QuantLib", "Documentation", and "Need Help?". The left sidebar shows navigation options like Home, Groups, and Notifications, along with links to other pages like "QuantLib download" and "andydong1209". The bottom of the screen shows the Facebook navigation bar with tabs for About, Discussion, Posts, Videos, and Members.

◆ 學習資訊：<https://github.com/andydong1209>

The screenshot shows the GitHub profile page for the user `andydong1209`. The profile picture is a circular photo of a man with glasses and a striped shirt standing outdoors with a scenic view in the background. The user's name, `andydong`, and GitHub handle, `andydong1209`, are displayed below the profile picture. A bio states: "A financial engineer, senior risk manager, familiar with QuantLib, C#/C++/C, Python, CUDA, OpenCL, SQL, VBA." Below the bio are buttons for "Edit profile" and "Contribution settings". The main navigation bar includes links for "Overview", "Repositories" (25), "Projects", and "Packages". The "Overview" tab is selected. The "Pinned" section contains four items: "TrainCourse" (Train Courses Provided by Andy Dong(董夢雲)), "Articles" (Articles to share), "NTU_FA2021" (Financial Algorithms 2021 in NTU), and "QuantLib_Learning_Center" (All information about QuantLib learning). The "Repositories" section shows 25 repositories. The "Explore" link is also visible in the top navigation bar.