binance

https://www.binance.com/en/lending#lending-activity

币安

https://www.binance.com/zh-CN/fee/schedule

这个是用来参加策略分享会和课程的qq, 上课要用这个，有时间也可以潜水看看有没有有用的策略。他们还有个论坛：

https://bbs.quantclass.cn/

qq: 2724517296 密码：shuzihuobi123!

binance合约

https://www.binance.com/zh-CN/futures/funding-history/4

u本位

BTCUSDT 永续

ETHUSDT 永续

BCHUSDT 永续

XRPUSDT 永续

EOSUSDT 永续

LTCUSDT 永续

TRXUSDT 永续

ETCUSDT 永续

LINKUSDT 永续

XLMUSDT 永续

ADAUSDT 永续

XMRUSDT 永续

DASHUSDT 永续

ZECUSDT 永续

XTZUSDT 永续

BNBUSDT 永续

ATOMUSDT 永续

ONTUSDT 永续

IOTAUSDT 永续

BATUSDT 永续

VETUSDT 永续

NEOUSDT 永续

QTUMUSDT 永续

IOSTUSDT 永续

THETAUSDT 永续

ALGOUSDT 永续

ZILUSDT 永续

KNCUSDT 永续

ZRXUSDT 永续

COMPUSDT 永续

OMGUSDT 永续

DOGEUSDT 永续

SXPUSDT 永续

KAVAUSDT 永续

BANDUSDT 永续

RLCUSDT 永续

WAVESUSDT 永续

MKRUSDT 永续

SNXUSDT 永续

DOTUSDT 永续

DEFIUSDT 永续

YFIUSDT 永续

BALUSDT 永续

CRVUSDT 永续

TRBUSDT 永续

YFIIUSDT 永续

RUNEUSDT 永续

SUSHIUSDT 永续

SRMUSDT 永续

BZRXUSDT 永续

EGLDUSDT 永续

SOLUSDT 永续

ICXUSDT 永续

STORJUSDT 永续

BLZUSDT 永续

UNIUSDT 永续

AVAXUSDT 永续

FTMUSDT 永续

HNTUSDT 永续

ENJUSDT 永续

FLMUSDT 永续

TOMOUSDT 永续

RENUSDT 永续

KSMUSDT 永续

NEARUSDT 永续

AAVEUSDT 永续

FILUSDT 永续

RSRUSDT 永续

LRCUSDT 永续

MATICUSDT 永续

OCEANUSDT 永续

CVCUSDT 永续

BELUSDT 永续

CTKUSDT 永续

AXSUSDT 永续

ALPHAUSDT 永续

ZENUSDT 永续

SKLUSDT 永续

GRTUSDT 永续

1INCHUSDT 永续

BTCBUSD 永续

AKROUSDT 永续

CHZUSDT 永续

SANDUSDT 永续

ANKRUSDT 永续

LUNAUSDT 永续

BTSUSDT 永续

LITUSDT 永续

UNFIUSDT 永续

DODOUSDT 永续

REEFUSDT 永续

RVNUSDT 永续

SFPUSDT 永续

XEMUSDT 永续

COTIUSDT 永续

CHRUSDT 永续

MANAUSDT 永续

ALICEUSDT 永续

HBARUSDT 永续

ONEUSDT 永续

LINAUSDT 永续

STMXUSDT 永续

DENTUSDT 永续

CELRUSDT 永续

HOTUSDT 永续

MTLUSDT 永续

OGNUSDT 永续

BTTUSDT 永续

NKNUSDT 永续

SCUSDT 永续

DGBUSDT 永续

1000SHIBUSDT 永续

ICPUSDT 永续

BAKEUSDT 永续

GTCUSDT 永续

ETHBUSD 永续

BTCUSDT 0924

ETHUSDT 0924

BTCDOMUSDT 永续

KEEPUSDT 永续

币本位

BTCUSD

ETHUSD

LINKUSD

BNBUSD

TRXUSD

DOTUSD

ADAUSD

EOSUSD

LTCUSD

BCHUSD

XRPUSD

ETCUSD

FILUSD

EGLDUSD

DOGEUSD

UNIUSD

THETAUSD

XLMUSD

数据网站

coinmarketcap

crypto watch https://cryptowat.ch/zh-cn/products/cryptocurrency-market-data-api

aicoin

coinapi https://docs.coinapi.io/

reference

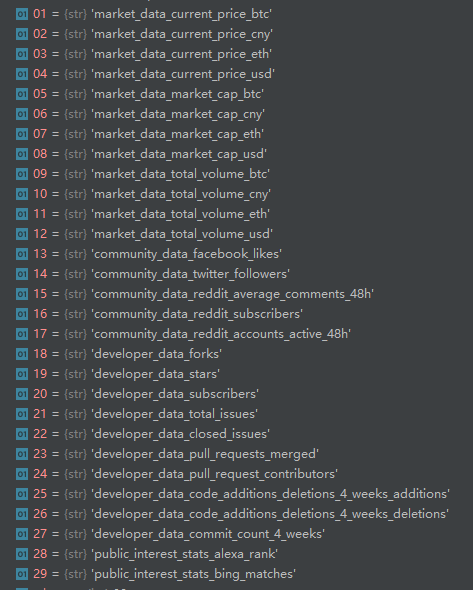
https://blog.rmotr.com/top-5-free-apis-to-access-historical-cryptocurrencies-data-2438adc8b62

Coingeko

1. coingeko中格式



1. pandas转换后列名称



3.Pandas columns 列表

['market\_data\_current\_price\_btc',

      'market\_data\_current\_price\_cny', 'market\_data\_current\_price\_eth',

      'market\_data\_current\_price\_usd', 'market\_data\_market\_cap\_btc',

      'market\_data\_market\_cap\_cny', 'market\_data\_market\_cap\_eth',

      'market\_data\_market\_cap\_usd', 'market\_data\_total\_volume\_btc',

      'market\_data\_total\_volume\_cny', 'market\_data\_total\_volume\_eth',

      'market\_data\_total\_volume\_usd', 'community\_data\_facebook\_likes',

      'community\_data\_twitter\_followers',

      'community\_data\_reddit\_average\_comments\_48h',

      'community\_data\_reddit\_subscribers',

      'community\_data\_reddit\_accounts\_active\_48h', 'developer\_data\_forks',

      'developer\_data\_stars', 'developer\_data\_subscribers',

      'developer\_data\_total\_issues', 'developer\_data\_closed\_issues',

      'developer\_data\_pull\_requests\_merged',

      'developer\_data\_pull\_request\_contributors',

      'developer\_data\_code\_additions\_deletions\_4\_weeks\_additions',

      'developer\_data\_code\_additions\_deletions\_4\_weeks\_deletions',

      'developer\_data\_commit\_count\_4\_weeks',

      'public\_interest\_stats\_alexa\_rank',

      'public\_interest\_stats\_bing\_matches']

算法交易

2019介绍

[了解算法交易，此篇足矣\_量化交易研究-CSDN博客\_算法交易](https://blog.csdn.net/weixin_42219751/article/details/97893208)

简介

https://blog.csdn.net/u012724887/article/details/99070574

一些例子

[算法交易简介 (renrendoc.com)](https://www.renrendoc.com/paper/135968067.html)

**算法交易的策略多达30多种**，目前已经发展到第四代算法技术。

**第一代算法交易比较简单，注重交易的执行效率和目的**，如TWAP、VWAP等；

**第二代算法交易注重提高交易质量，降低交易成本，如最大限度接近开盘价、收盘价、或心理价等某一特定价格为目标**，该算法有I/S（按照执行价格和期望价格差来拆分大单）、MOC(让成交价尽可能接近闭市价格)、MOO（让成交价尽可能接近开市价格）、Peg/Pegging（让成交价尽可能接近某一档价格）；

**第三代算法交易是**你在这样交易，对手也在这样交易，**为了蒙蔽对手**，引入Iceberg算法（按给定比例逐步暴露订单直至全部成交），之后又引入了反侦察算法比如用来发现潜在Iceberg算法的Sniffer算法等。



**第四代算法技术更趋向于智能交易**，例如复杂事件处理（CEP）、新闻交易（News Trading）等。这些策略根据不同的市场特点和交易需求进行设计。

Ccxt

ccxt文档

https://ccxt.readthedocs.io/en/latest/

ccxt github

https://github.com/ccxt/ccxt

服务器实盘

<https://cloud.tencent.com/document/product/213/35700>

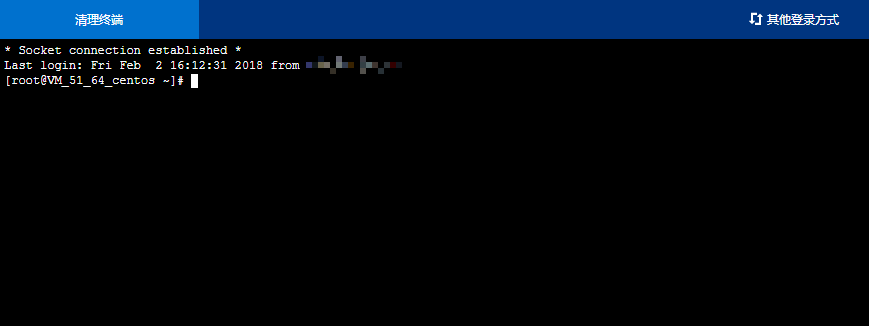
**适用本地操作系统**

Window，Linux 或者 Mac OS

**鉴权方式**

**密码**或**密钥**

**操作步骤**

1. 登录 [云服务器控制台](https://console.cloud.tencent.com/cvm/index)。
2. 在实例的管理页面，选择需要登录的 Linux 云服务器，单击【登录】。如下图所示：
3. 在弹出的【登录Linux实例】窗口，选择【标准登录方式】，单击【立即登录】。如下图所示：
4. 在打开的 WebShell 登录页面，根据实际需求，选择【密码登录】或者【密钥登录】方式进行登录。如下图所示：如果登录成功，WebShell 界面会出现 Socket connection established 提示。如下图所示：

以上是从控制台登陆，以下介绍使用SSH文件登陆

**操作场景**

本文以 PuTTY 软件为例，介绍如何在 Windows 系统的本地电脑中使用远程登录软件登录 Linux 实例。

**适用本地操作系统**

Windows

**鉴权方式**

**密码**或**密钥**

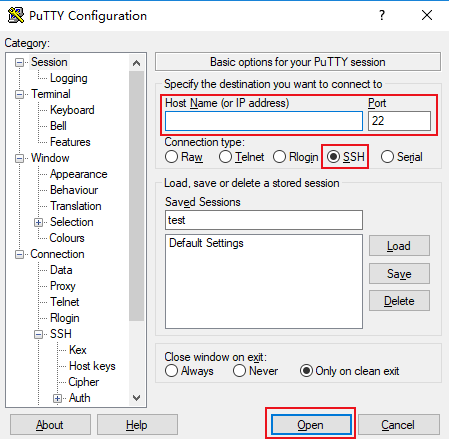
**操作步骤**

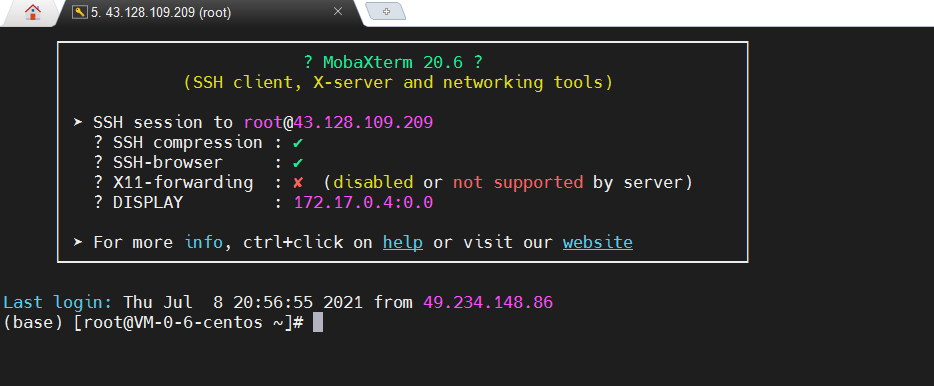
* 使用密钥登录

1. 下载 Windows 远程登录软件，即 PuTTY。

[点此获取 PuTTY](https://the.earth.li/~sgtatham/putty/latest/w64/putty.exe)

1. 双击【putty.exe】，打开 PuTTY 客户端。
2. 在 PuTTY Configuration 窗口中，输入以下内容。如下图所示：

参数举例说明如下：

* **Host Name（or IP address）**：云服务器的公网 IP（登录 [云服务器控制台](https://console.cloud.tencent.com/cvm/index)，可在列表页及详情页中获取公网 IP）。
* **Port**：云服务器的端口，必须设置为22。
* **Connect type**：选择 “SSH”。
* **Saved Sessions**：填写会话名称，例如 test。配置 “Host Name” 后，再配置 “Saved Sessions” 并保存，则后续使用时您可直接双击 “Saved Sessions” 下保存的会话名称即可登录服务器。
* 单击【Open】，进入 “PuTTY” 的运行界面，提示 “login as:”。
* 在 “login as” 后输入用户名，按 **Enter**。
* 在 “Password” 后输入密码，按 **Enter**。输入的密码默认不显示，如下图所示：

1. 登录完成后，命令提示符左侧将显示当前登录云服务器的信息。

**使用限制**

* 仅支持 Linux 实例。
* 每个腾讯云账号最多可具备100个 SSH 密钥对。
* 腾讯云不会保管您的私钥信息，您需要在创建 SSH 密钥后下载私钥，并妥善保管。
* 基于数据安全考虑，加载密钥需要在关机状态下进行。

代码例子

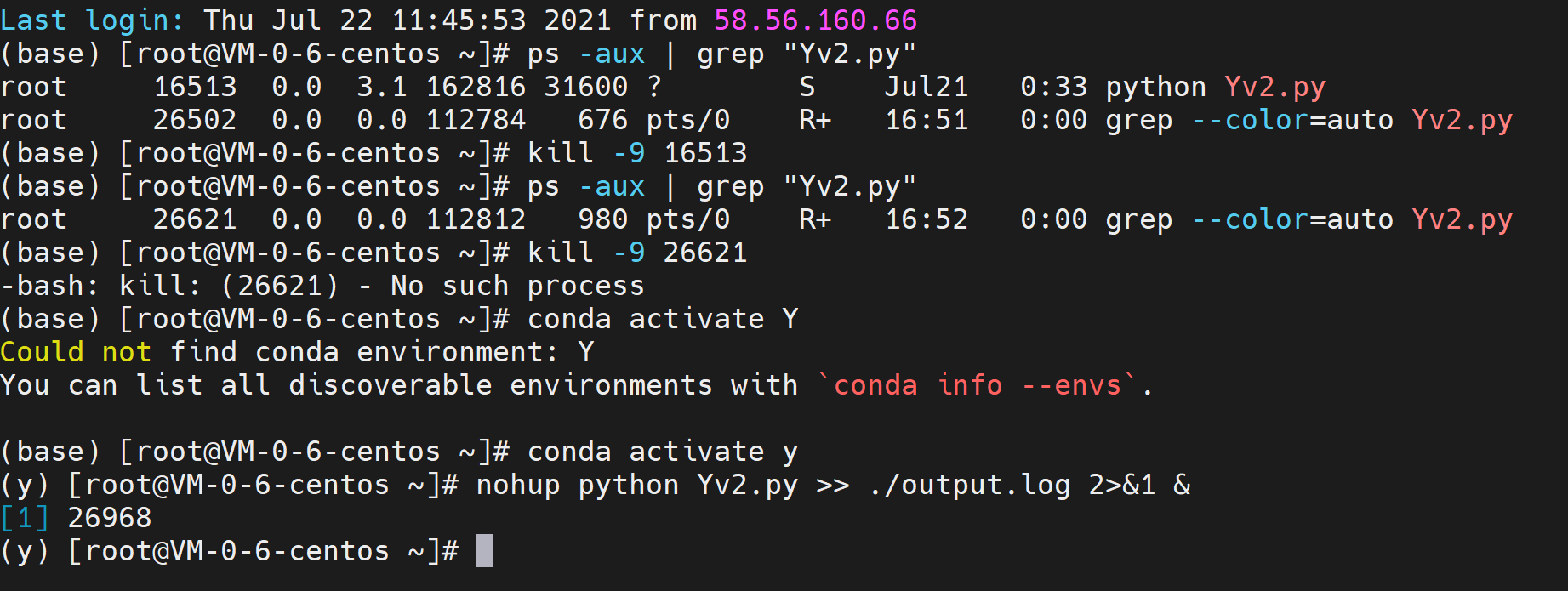
[root@VM-0-6-centos ~]# conda activate y(y)

[root@VM-0-6-centos ~]# cd /root/SunDX/(y)

[root@VM-0-6-centos SunDX]# nohup python Y\_v1.py >> ./output.log 2>&1 &

(y) [root@VM-0-6-centos SunDX]# ps -aux|grep python

root      1034  0.0  1.3 574288 13560 ?        Ssl  Jul08   0:37 /usr/bin/python2 -Es /usr/sbin/tuned -l -Proot     23948  0.3  3.2 161276 32768 pts/1    S    20:56   0:00 python Y\_v1.pyroot     24164  0.0  0.0 112812   972 pts/1    R+   20:57   0:00 grep --color=auto python



备注

不要ctrl+C

<https://blog.csdn.net/hfismyangel/article/details/80258126>

文章目录

安装

升级

卸载Anaconda软件

conda环境使用基本命令

查看指定包可安装版本信息命令

更新，卸载安装包：

删除虚拟环境

清理（conda瘦身）

复制/重命名/删除env环境

conda自动开启/关闭激活

Conda 安装本地包

解决conda/pip install 下载速度慢

conda数据源管理

pip数据源管理

pip安装包管理

pip和conda批量导出、安装组件(requirements.txt)

常用软件安装

问题

1:failed ERROR conda.core.link:\_execute(502):

2.anaconda或conda不是内部命令

安装

linux环境

bash Anaconda3-2019.07-Linux-x86\_64.sh

#yes+回车

#然后重启terminal

window环境：直接双击安装exe文件，然后根据安装向导进行安装

升级

升级Anaconda需要先升级conda

conda update conda          #基本升级

conda update anaconda       #大的升级

conda update anaconda-navigator    //update最新版本的anaconda-navigator

卸载Anaconda软件

由于Anaconda的安装文件都包含在一个目录中，所以直接将该目录删除即可。删除整个Anaconda目录：

计算机控制面板->程序与应用->卸载 //windows

或者

找到C:\ProgramData\Anaconda3\Uninstall-Anaconda3.exe执行卸载

rm -rf anaconda    //ubuntu

1

最后，建议清理下.bashrc中的Anaconda路径。

conda环境使用基本命令

conda update -n base conda        #update最新版本的conda

conda create -n xxxx python=3.5   #创建python3.5的xxxx虚拟环境

conda activate xxxx               #开启xxxx环境

conda deactivate                  #关闭环境

conda env list                    #显示所有的虚拟环境

conda info --envs                 #显示所有的虚拟环境

conda install --channel <https://conda.anaconda.org/anaconda> tensorflow=1.8.0

更新，卸载安装包：

conda list         #查看已经安装的文件包

conda list  -n xxx       #指定查看xxx虚拟环境下安装的package

conda update xxx   #更新xxx文件包

conda uninstall xxx   #卸载xxx文件包

删除虚拟环境

conda remove -n xxxx --all   //创建xxxx虚拟环境

清理（conda瘦身）

conda clean就可以轻松搞定！第一步：通过conda clean -p来删除一些没用的包，这个命令会检查哪些包没有在包缓存中被硬依赖到其他地方，并删除它们。第二步：通过conda clean -t可以将删除conda保存下来的tar包。

conda clean -p      //删除没有用的包

conda clean -t      //删除tar包

conda clean -y --all //删除所有的安装包及cache

conda activate   #默认激活base环境

conda activate xxx  #激活xxx环境

conda deactivate #关闭当前环境

conda config --set auto\_activate\_base false  #关闭自动激活状态

conda config --set auto\_activate\_base true  #关闭自动激活状态