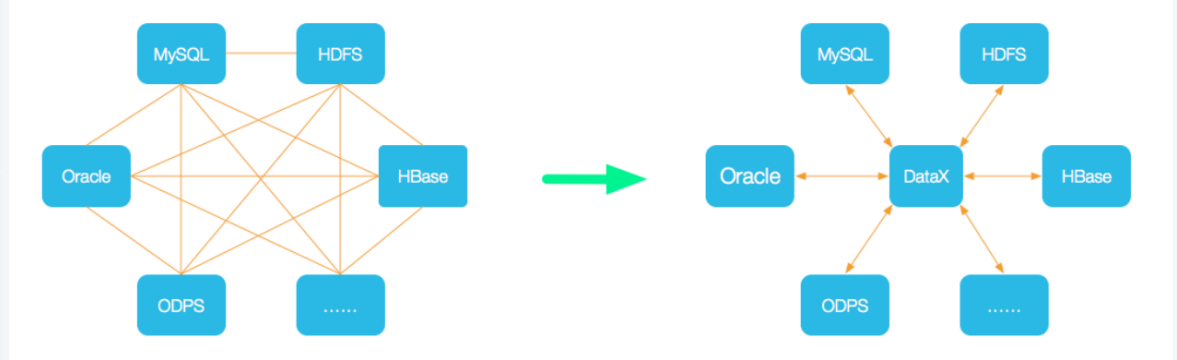
datax介绍与使用

源码地址

https://github.com/alibaba/DataX/

DataX

DataX 是阿里巴巴集团内被广泛使用的离线数据同步工具/平台，实现包括 MySQL、SQL Server、Oracle、PostgreSQL、HDFS、Hive、HBase、OTS、ODPS 等各种异构数据源之间高效的数据同步功能。



设计理念

为了解决**异构数据源同步问题**，DataX将复杂的网状的**同步链路**变成了星型数据链路，**DataX**作为**中间传输载体负责连接各种数据源**。当需要接入一个新的数据源的时候，只需要将此数据源对接到DataX，便能跟已有的数据源做到无缝数据同步。（这里我解释一下左边的网状结构是什么意思，譬如我们要做一个从MySql到Oracle的同步工具，我们需要写一个mysql-oracle，这时，如果我们需要MySql到SqlServer的同步工具，我们又需要写一个mysql-sqlserver，以此类推，会构成一个复杂网状结构。而DataX的理念就是，**所有的异源数据库只需将数据从元数据同步到DataX，再由DataX同步到目的数据库，以此架构即构成右边星形数据链路。**

datax解决的问题

Features

DataX本身作为数据同步框架，将不同数据源的同步抽象为从源头数据源读取数据的Reader插件，以及向目标端写入数据的Writer插件，理论上DataX框架可以支持任意数据源类型的数据同步工作。同时DataX插件体系作为一套生态系统, 每接入一套新数据源该新加入的数据源即可实现和现有的数据源互通。

System Requirements

- Linux
- JDK(1.8以上，推荐1.8)
- Python(推荐Python2.6.X)
- Apache Maven 3.x (Compile DataX)

Quick Start

- 工具部署
  - 方法一、直接下载DataX工具包：DataX下载地址

    下载后解压至本地某个目录，进入bin目录，即可运行同步作业：

    ```
    $ cd  {YOUR_DATAX_HOME}/bin
    $ python datax.py {YOUR_JOB.json}
    ```

    自检脚本：
    python {YOUR_DATAX_HOME}/bin/datax.py {YOUR_DATAX_HOME}/job/job.json
  - 方法二、下载DataX源码，自己编译：DataX源码

    (1)、下载DataX源码：

    ```
    $ git clone git@github.com:alibaba/DataX.git
    ```

    (2)、通过maven打包：

    ```
    $ cd  {DataX_source_code_home}
    $ mvn -U clean package assembly:assembly -Dmaven.test.skip=true
    ```

    打包成功，日志显示如下：

    ```
    [INFO] BUILD SUCCESS
    [INFO] ------------------------------------------------------------------
    -
    [INFO] Total time: 08:12 min
    [INFO] Finished at: 2015-12-13T16:26:48+08:00
    [INFO] Final Memory: 133M/960M
    [INFO] ------------------------------------------------------------------
    -
    ```

    打包成功后的DataX包位于 {DataX_source_code_home}/target/datax/datax/ ，结构如下：

    ```
    $ cd  {DataX_source_code_home}
    $ ls ./target/datax/datax/
    bin     conf        job     lib     log     log_perf    plugin
    script      tmp
    ```

- 配置示例：从stream读取数据并打印到控制台
  - 第一步、创建作业的配置文件（json格式）

    可以通过命令查看配置模板： python datax.py -r {YOUR_READER} -w {YOUR_WRITER}

    ```
    $ cd  {YOUR_DATAX_HOME}/bin
    $  python datax.py -r streamreader -w streamwriter
    DataX (UNKNOWN_DATAX_VERSION), From Alibaba !
    Copyright (C) 2010-2015, Alibaba Group. All Rights Reserved.
    Please refer to the streamreader document:
    ```

```
https://github.com/alibaba/DataX/blob/master/streamreader/doc/streamreader.md

Please refer to the streamwriter document:

https://github.com/alibaba/DataX/blob/master/streamwriter/doc/streamwriter.md

Please save the following configuration as a json file and  use
     python {DATAX_HOME}/bin/datax.py {JSON_FILE_NAME}.json
to run the job.

{
    "job": {
        "content": [
            {
                "reader": {
                    "name": "streamreader",
                    "parameter": {
                        "column": [],
                        "sliceRecordCount": ""
                    }
                },
                "writer": {
                    "name": "streamwriter",
                    "parameter": {
                        "encoding": "",
                        "print": true
                    }
                }
            }
        ],
        "setting": {
            "speed": {
                "channel": ""
            }
        }
    }
}
```

根据模板配置json如下：

```
#stream2stream.json
{
  "job": {
    "content": [
      {
        "reader": {
          "name": "streamreader",
          "parameter": {
            "sliceRecordCount": 10,
            "column": [
              {
                "type": "long",
                "value": "10"
              },
```

```
              {
                "type": "string",
                "value": "hello，你好，世界-DataX"
              }
            ]
          }
        },
        "writer": {
          "name": "streamwriter",
          "parameter": {
            "encoding": "UTF-8",
            "print": true
          }
        }
      }
    ],
    "setting": {
      "speed": {
        "channel": 5
       }
    }
  }
}
```

- 第二步：启动DataX

```
$ cd {YOUR_DATAX_DIR_BIN}
$ python datax.py ./stream2stream.json
```

同步结束，显示日志如下：

```
...
2015-12-17 11:20:25.263 [job-0] INFO  JobContainer -
任务启动时刻                    : 2015-12-17 11:20:15
任务结束时刻                    : 2015-12-17 11:20:25
任务总计耗时                    :                 10s
任务平均流量                    :              205B/s
记录写入速度                    :              5rec/s
读出记录总数                    :                 50
读写失败总数                    :                  0
```

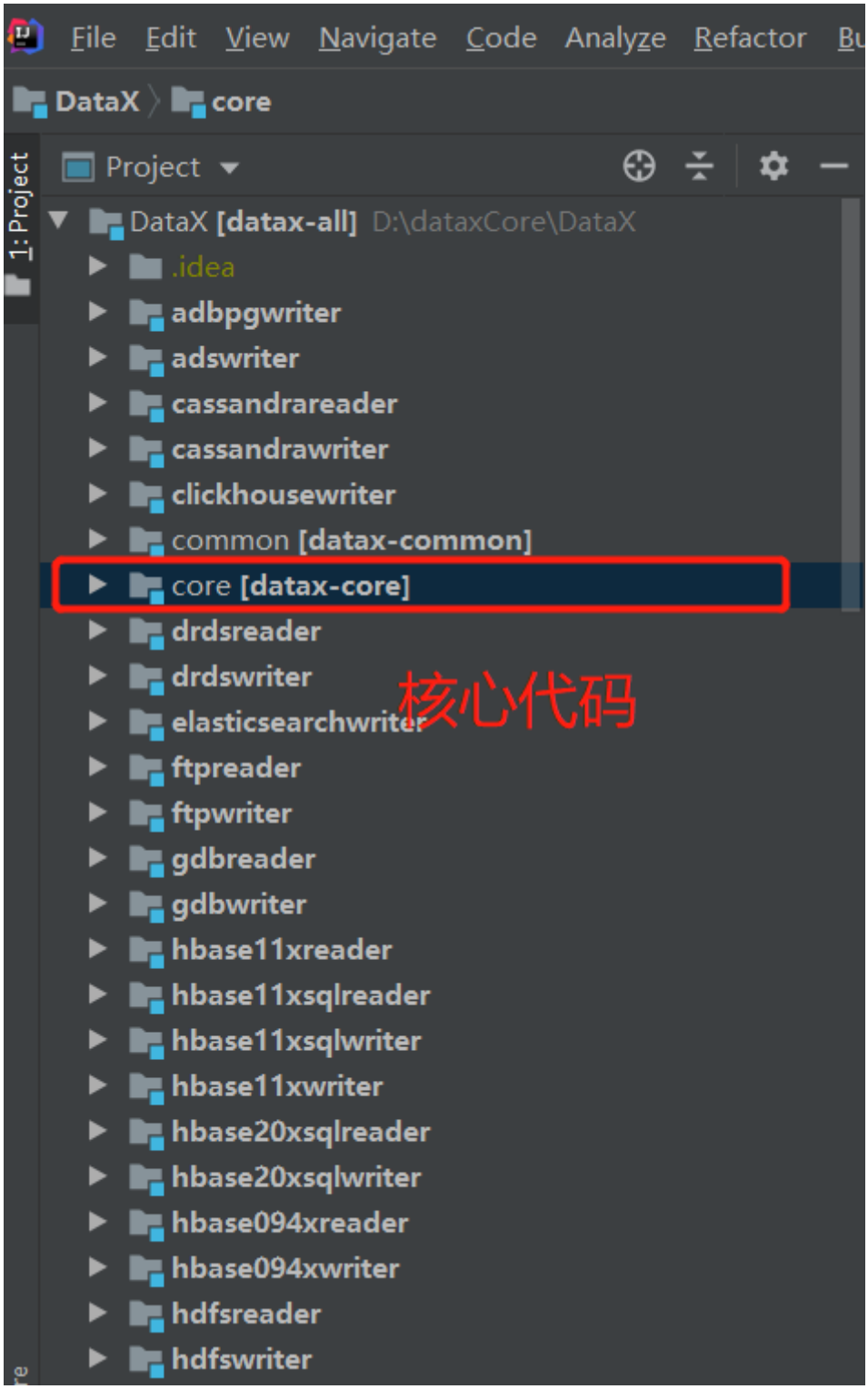
datax使用案例

1.首先要看过Quick Start内容

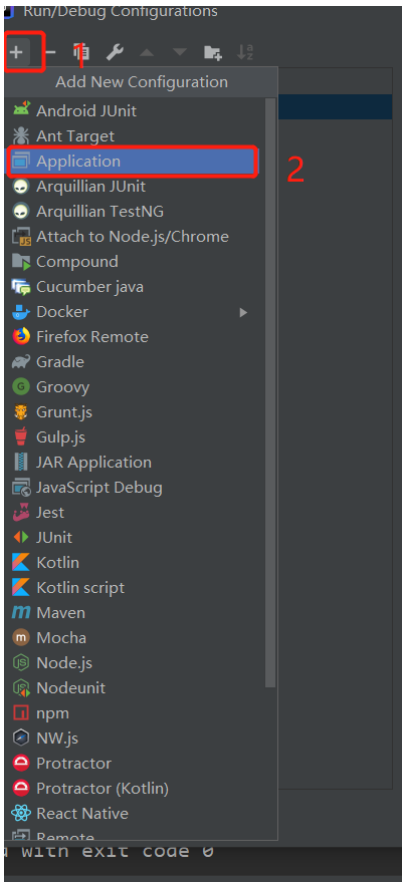
2.在本地idea运行datax项目，免下载python

3.首先下载好datax源码

4.用idea打开

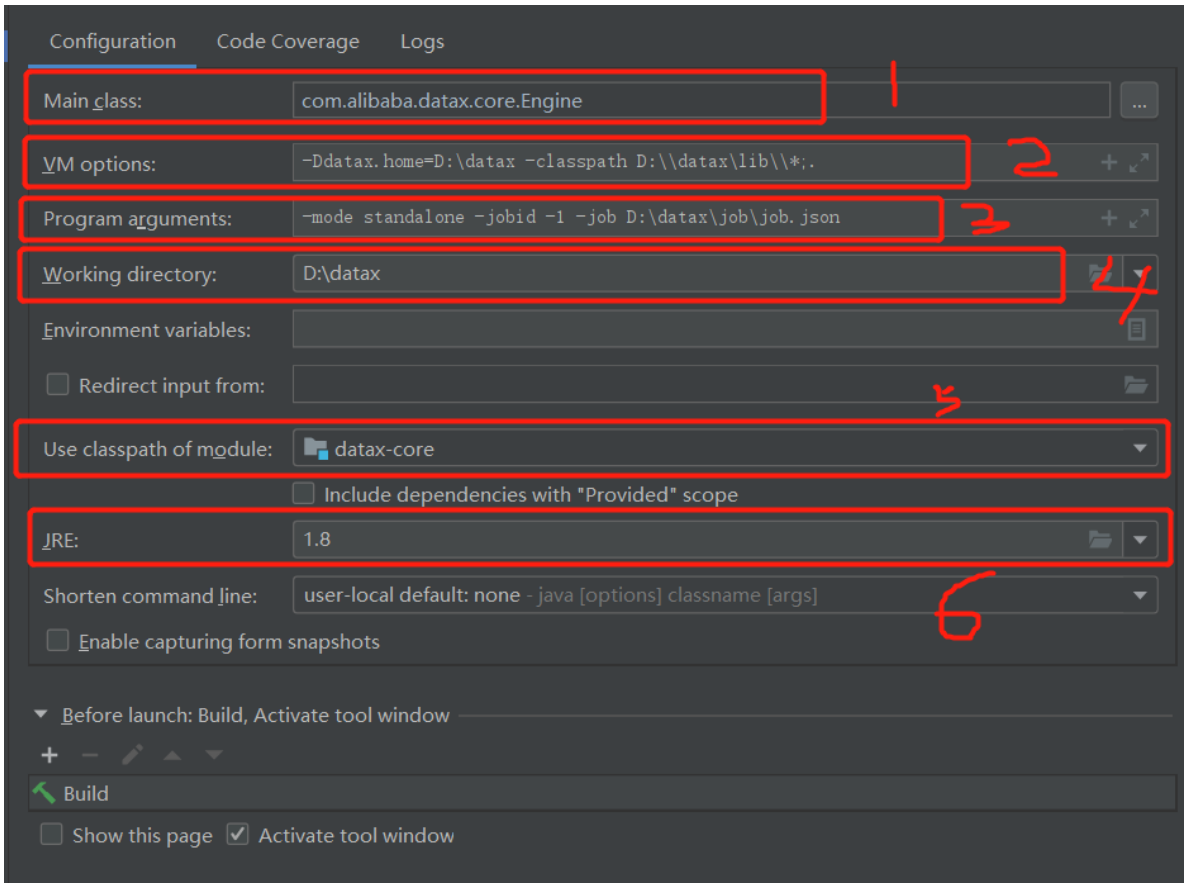


5.启动类Engine上配置参数



创建一个Application

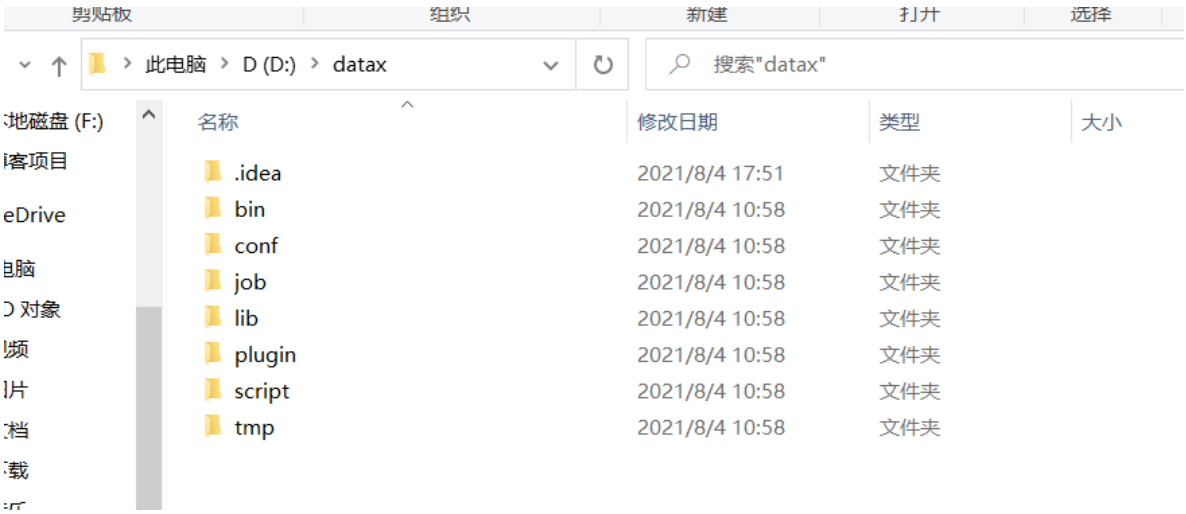
6.



图片上的1 统一填写为com.alibaba.datax.core.Engine

图片上的2

```
-Ddatax.home=D:\datax  (改成自己电脑上的datax（github的上工具包目录））
-classpath D:\\datax\lib\\*;. (改成datax的lib目录)
```



图片上的3

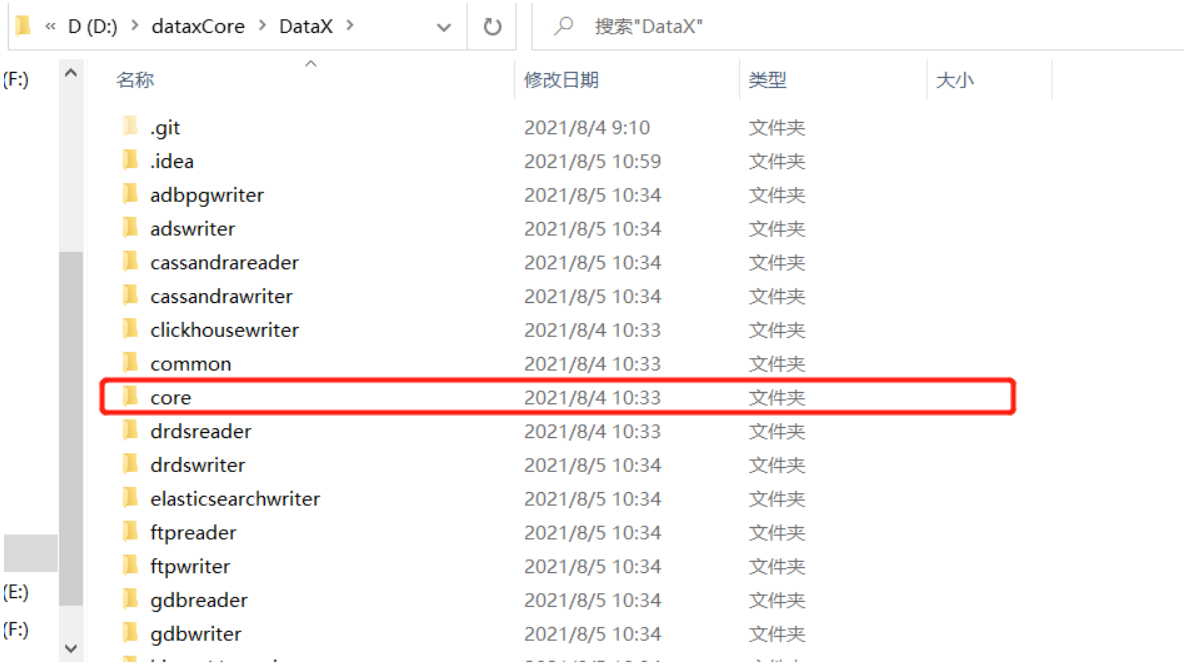
```
-mode standalone -jobid -1 -job D:\datax\job\job.json
```

改成本地的datax目录的job目录下的job.json（这是模板配置json）

图片上的4

改成自己的datax根目录

图片上的5 改成datax源码的core

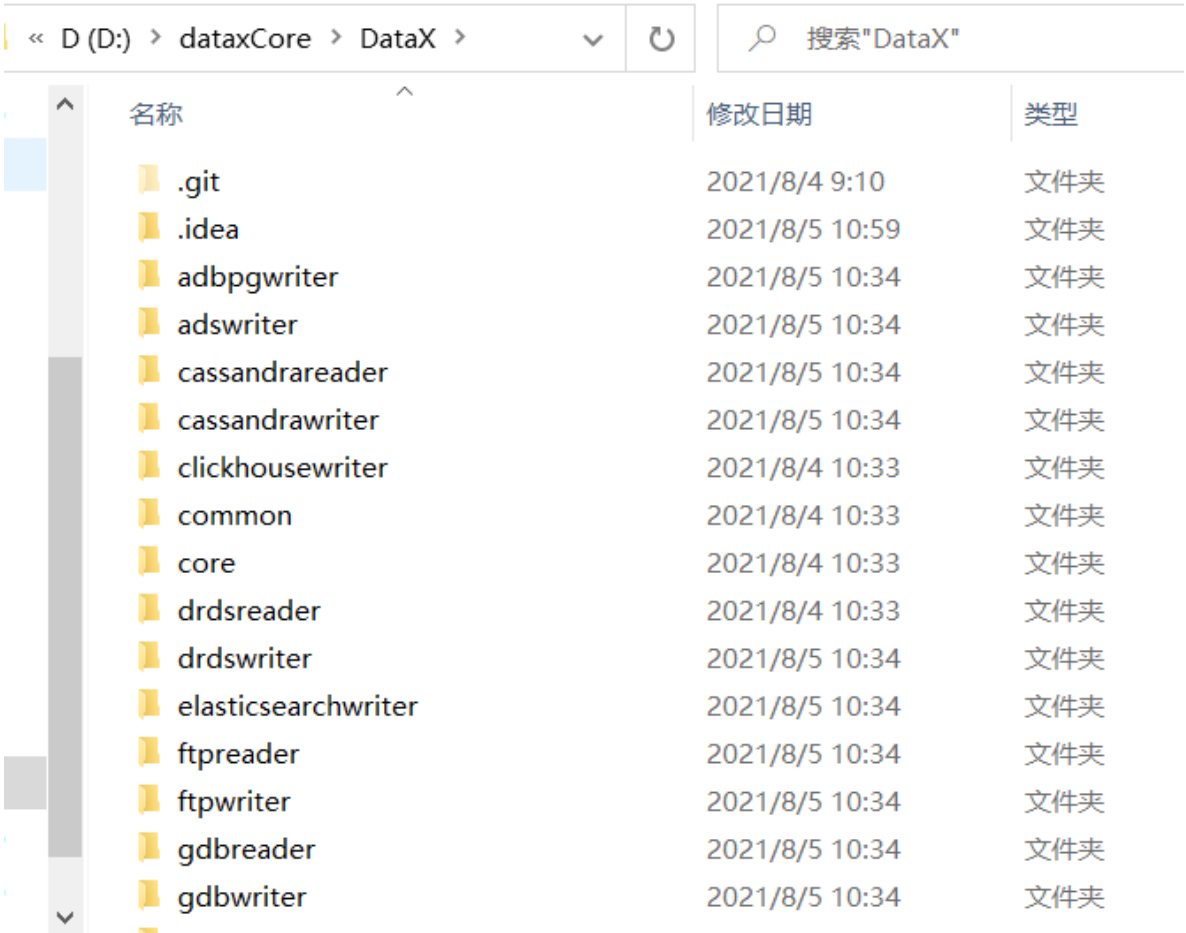


图片上的6 jre改成1.8

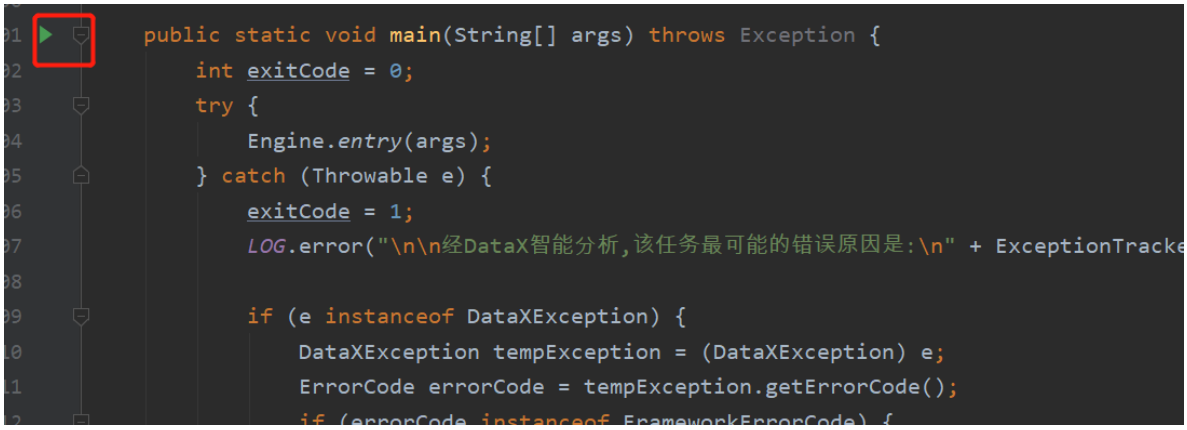
PS重要 datax的工具包和datax源码是不同的下载地址 请注意

datax工具包的目录是

datax源码的目录是



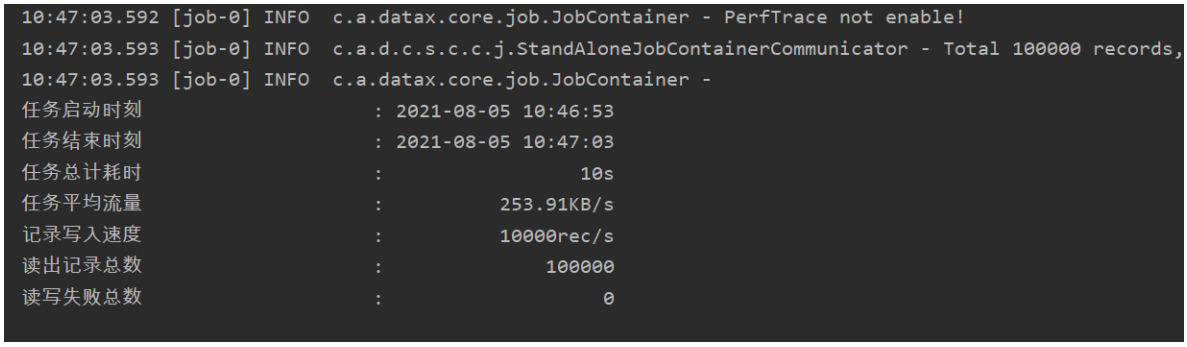
当我们配置好之后，运行Engine.java的main方法



```
public static void main(String[] args) throws Exception {
    int exitCode = 0;
    try {
        Engine.entry(args);
    } catch (Throwable e) {
        exitCode = 1;
        LOG.error("\n\n经DataX智能分析,该任务最可能的错误原因是:\n" + ExceptionTracke

        if (e instanceof DataXException) {
            DataXException tempException = (DataXException) e;
            ErrorCode errorCode = tempException.getErrorCode();
            if (errorCode instanceof FrameworkErrorCode) {
```

运行成功：



```
10:47:03.592 [job-0] INFO  c.a.datax.core.job.JobContainer - PerfTrace not enable!
10:47:03.593 [job-0] INFO  c.a.d.c.s.c.c.j.StandAloneJobContainerCommunicator - Total 100000 records,
10:47:03.593 [job-0] INFO  c.a.datax.core.job.JobContainer -
任务启动时刻                    : 2021-08-05 10:46:53
任务结束时刻                    : 2021-08-05 10:47:03
任务总计耗时                    :                 10s
任务平均流量                    :          253.91KB/s
记录写入速度                    :          10000rec/s
读出记录总数                    :              100000
读写失败总数                    :                   0
```

datax源码解读汇总

地址：

https://waterwang.blog.csdn.net/article/details/114630690?utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7Edefault-6.base&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7Edefault-6.base