《信息系统集成与管理》
实验一 halibut 描述文档

学　　院: 遥感信息工程学院

班　　级: 20F10

学　　号: 2020302131201

姓　　名: 常耀文

指导教师: 王华敏

2022 年 12 月 7 日

目录

1. 实习概述

1.1 实习目的

本次实验旨在通过 Kettle 开源工具，结合课程使用的 FTP 服务器以及 MySQL 关系数据库，实现对个人 FTP 访问日志的抽取、转换、装载（ETL）流程，通过对于理论课程的知识与内容进行实践来加深对 ETL 知识的理解。

1.2 实习任务

本次实习利用 Kettle 开源工具与 MySQL 数据库，进行具有以下实验目的实验操作：

1. 创建数据库表，用于输出
2. 对 ftp server 中的 halibut.log 发起 ftp 请求，作为数据输入源
3. 定义转换规则
4. 定义输出
5. 执行转换过程
6. 检查数据库表，验证是否成功
7. 导出数据库表为 sql 文件

1.3 实习所用软件梗概

1.3.1 Kettle

Kettle 是一款国外开源的 ETL 工具，对商业用户也没有限制，纯 Java 编写，可以在 Window、Linux、Unix 上运行，绿色无需安装，数据抽取高效稳定。Kettle 中文名称叫水壶，它允许管理来自不同数据库的数据，把各种**数据**放到一个壶里，然后以一种**指定的格式**流出。Kettle 中有两种脚本文件，**Transformation** 和 **Job**， Transformation 完成针对数据的基础转换，Job 则完成整个工作流的控制。通过图形界面设计实现做什么业务，并在 Job 下的 start 模块，有一个定时功能，可以每日，每周等方式进行定时。

本次使用的是 Kettle 中的 spoon 工具。

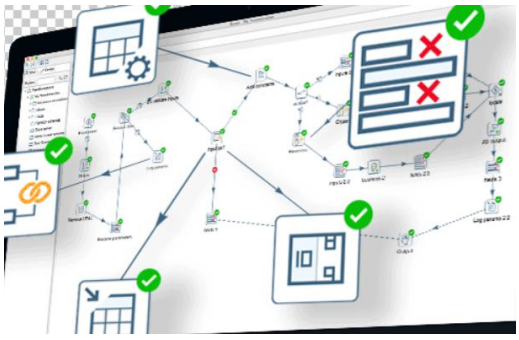


Fig1.Kettle 工具

1.3.2 MySQL

MySQL 是关系数据库管理系统，MySQL 所使用的 SQL 语言是用于访问数据库的最常用标准化语言。MySQL 软件采用了双授权政策，是许多公司都采用的数据库之一/

Fig2.MySQL 数据库

1.3.3 Navicat for MySQL

“Navicat”是一套可创建多个连接的数据库管理工具，用以方便管理 MySQL、Oracle、PostgreSQL、SQLite、SQL Server、MariaDB 和 MongoDB 等不同类型的数据库，它与阿里云、腾讯云、华为云、Amazon RDS、Amazon Aurora、Amazon Redshift、Microsoft Azure、Oracle Cloud 和 MongoDB Atlas 等云数据库兼容。你可以创建、管理和维护数据库。Navicat 的功能足以满足专业开发人员的所有需求，但是对数据库服务器初学者来说又简单易操作。Navicat 的用户界面 (GUI) 设计良好，让你以安全且简单的方法创建、组织、访问和共享信息。

在本次实验中 Navicat 主要用于创建数据库表以及可视化操作结果。

Fig3.Navicat 数据库

2. 实习环境概述

实验环境是实验操作的必不可少的步骤，本部分本节将从实验数据概述、软件环境，电脑硬件设别三方面对实验环境进行讨。

2.1 实验数据

实验使用的数据是朱老师提供的 FTP 访问日志，日志文件存放在 FTP 服务器的根目录下，名为 halibut.log，本文使用的日志文件大小为 8.05KB，**文件共计 81 行**。文件中日志记录是非定长字符串，通过分析文件的形式，可以得出文件的格式如下表所示。

内容	**字段长度（单位：UTF-8 字符）**
记录的时间	24（内部包含空格）
传输耗时	不定，长度为整型
客户端 IP	不定，长度也为整型
文件大小	不定，但是为整型
文件名	不定
传输类型、传输方式	2
特殊标记、访客类型	2
用户名	不定
服务名	3
授权方式	1
IP 已认证标志	1
传输状态	1

Table1.日志数据表

2.2 软件环境

本次实验采用个人计算机实现，所选取的软件也是从 Ftp 服务器下获取，具体的软件环境如下表所示

Table2.软件版本表

软件名称	安装版本
MYSQL	8.0.31
NAVICAT FOR MYSQL	Navicat 16
JDK	1.8.0_351
JAR	mysql-connector-java-5.1.48
KETTLE	7.1.0.0-12

2.3 设备环境

本次使用电脑设备与系统环境如下图所示：

项目	值
操作系统名称	Microsoft Windows 11 家庭中文版
版本	10.0.22621 版本 22621
其他操作系统描述	没有资料
操作系统制造商	Microsoft Corporation
系统名称	LAPTOP-QRUO7UUI
系统制造商	LENOVO
系统型号	82AV
系统类型	基于 x64 的电脑
系统 SKU	LENOVO_MT_82AV_BU_idea_FM_Legion Y7000 2020
处理器	Intel(R) Core(TM) i5-10300H CPU @ 2.50GHz，2496 Mhz，4 个内核，...
BIOS 版本/日期	LENOVO EFCN31WW, 2020/4/27
SMBIOS 版本	3.2
嵌入式控制器版本	1.31
BIOS 模式	UEFI
主板制造商	LENOVO
主板产品	LNVNB161216
主板版本	SDK0L77769 WIN
平台角色	移动
安全启动状态	启用
PCR7 配置	需要提升才能查看
Windows 目录	C:\WINDOWS
系统目录	C:\WINDOWS\system32
启动设备	\Device\HarddiskVolume1
区域设置	中国
硬件抽象层	版本 = "10.0.22621.819"
用户名	LAPTOP-QRUO7UUI\lenovo
时区	中国标准时间
已安装的物理内存(RAM)	16.0 GB
总的物理内存	15.9 GB
可用物理内存	7.88 GB
总的虚拟内存	34.8 GB

Fig4.电脑系统与硬件环境

3. 实验设计思路与细节设计

本部分主要对于本次实验进行思路与总体架构进行设计，并且提出总体的技术目标与实现流程。

3.1 总体设计思路

本次实习的任务均需要采用 Kettle 工具中的 Spoon 组件完成。于是构建起任务的总体思路：

1. 首先连接 FTP 服务器，并从服务器根目录中获取个人的日志文件；
2. 使用转换工具，转换从 FTP 服务器中抽取到的日志文件记录为与数据库中表属性对应的范式；
3. 连接数据库，将转换后的数据加载更新至数据库中；最后生成可导入数据库的.sql 文件同时查验作业结果的正确性。

本次实习严格遵循 ETL 的抽取，转换，装载的操作顺序流程，设计的流程图如下所示：

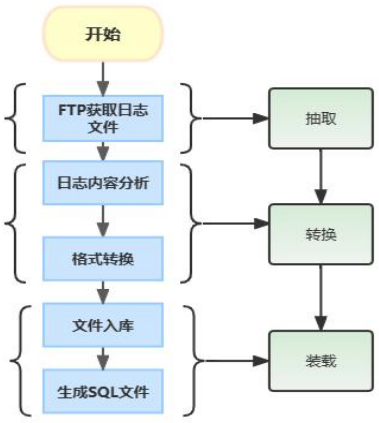


Fig5.实验设计流程图

3.2 实验细节设计

在本次实验设计的细节层面上，转换步骤中使用了先拆分后合并的数据流处理方式，将日志依空格拆成字段；然后合并日期字段；接着整理、命名字段；最终装载入数据库，利用 Navicat for MySQL 生成.sql 文件,具体实验步骤设计如下图所示：

Fig6.实验细节设计流程图

4. 实习内容

4.1 实习软件安装与环境配置

在开始实习前需要进行实习软件的下载与配置,本次实习需要特别注意各个软件之间存在版本的依赖关系，如果版本之间并不是相互对应的话，可能会导致在后续的操作流程中出现错误，因此需要安装正确且对应的版本。本次实习主要配置 5 个环境，JDK 环境，安装 Kettle，配置 Jar 包，配置 MySQL 数据库，安装 Navicat 16 for MySQL

4.1.1 安装 JDK

通过查阅网上资料，了解到安装 Kettle 之前需要安装 JDK，FTP 服务器下的 JDK 版本在向导指引下安装，下图是 JDK 版本。

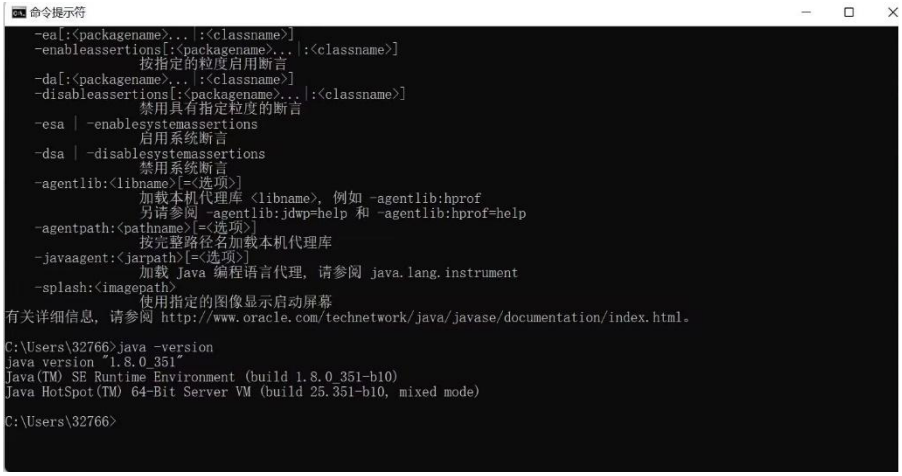


Fig7.JDK 安装结果

4.1.2 安装 Kettle Spoon

通过安装服务器下的 Kettle 进行安装，出现了版本不匹配的错误，因此从官网上又下载了最新版本的 Kettle 工具，然后解压压缩包，在解压后的文件夹中找到 Spoon.bat 后点击 Spoon.bat 就可以直接使用工具。

4.1.3 安装 MySQL

MySQL 的安装通过官网下载 MSI，然后根据安装向导选择自己想要的工具，并依照向导指引即可完成安装。安装成功后启动服务即可。安装成功后，打开命令行，在命令行中输入 mysql -u root -p 即可验证数据库是否成功启动。

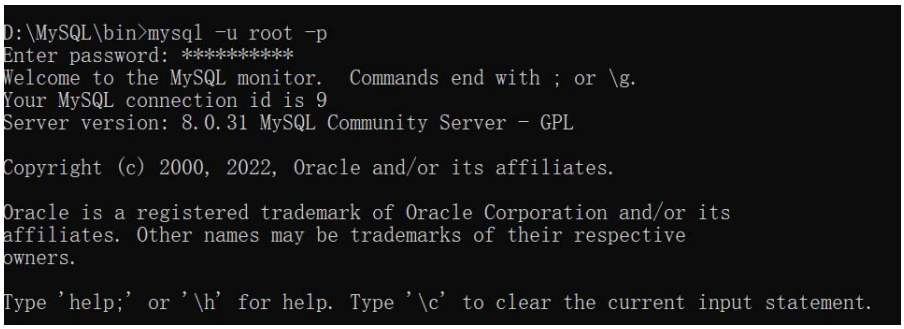


Fig8.MySQL 安装启动结果

4.1.4 配置 Jar 包

若要访问数据库，需要下载对应的 Jar 包提供相关支持。本文使用的是 MySQL 数据库，因此需要在官网（MySQL :: Download MySQL Connector/J (Archived Versions)）找寻并下载相应的数据库连接包，本文使用的包名称和版本详见 1.2 节中的表 2。下载后解压 Jar 包，然后将整个文件夹放在 Kettle 根目录中 lib 下，并将 jar 包中的相应的 jar 文件也放在 lib 目录下（如，本文将文件 mysql-connector-java-5.1.48-bin.jar 放置在 lib 目录中）。

如果未配置成功，则 Kettle 工具不能连接数据库，反之则可作为配置成功的标志。

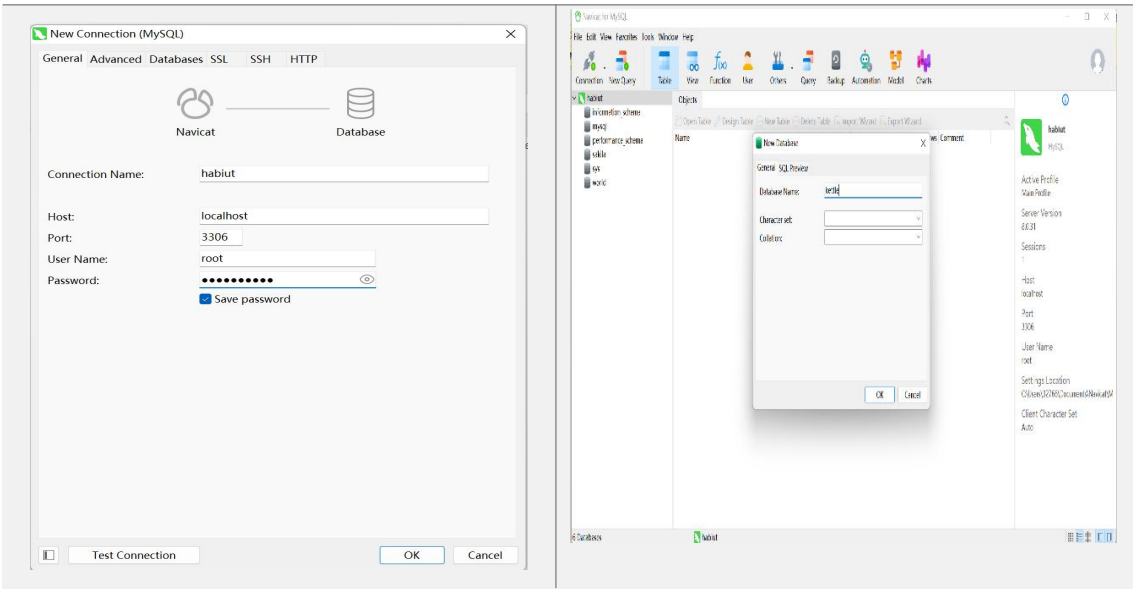
4.1.5 安装 Navicat for MySQL

Navicat for MySQL 作为一款可视化操作数据库软件，下载安装在官网（Navicat for MySQL | MySQL 数据库管理和开发工具），注意如果没有会员，只能限免使用 15 天，由于本次实习没有较多使用到该软件，仅仅利用游客模式访问了软件。下载安装的过程依据向导即可。

4.2 数据库、表的建立

在安装好 MySQL 数据库后，建立实验所需的数据库和存放日志记录的数据表。可以在命令行中连接数据库后使用 SQL 语句完成，或者使用 Navicat 建立数据库，本次建立数据库采用 Navicat，建立过程如图所示：

FIG9.NAVICAT 连接 MYSQL　　FIG10.创建数据库 KETTLE



然后根据 2.1 节中的 Table1 所示的字段名、字段类型和字段长度建立数据表，建立数据表通过命令行的方式建立数据表，具体操作如下图所示：

```
mysql> show tables;
Empty set (0.00 sec)

mysql> CREATE TABLE `haibut` (
    -> `record_time` varchar(30) ,
    -> `Transmission_time` int,
    -> `connection_ip` varchar(20) ,
    -> `file_size` int ,
    -> `file_name` varchar(200),
    -> `transmission_type` varchar(5),
    -> `action_mark` varchar(5) ,
    -> `transmission_direction` varchar(5),
    -> `access_mode` char(1),
    -> `user_name` varchar(200) ,
    -> `service_name` varchar(5),
    -> `authorization_method` int,
    -> `ip_authorization` varchar(5) ,
    -> `completion_status` varchar(5)
    -> ) ;
Query OK, 0 rows affected (0.03 sec)

mysql> show tables;
+------------------+
| Tables_in_kettle |
+------------------+
| haibut           |
+------------------+
1 row in set (0.00 sec)

mysql>
```

Fig11.建立数据表 haibut

各个字段与数据表之间的关系对应如下表所示：

字段名	*对应日志内容*	*类型*
record_time	记录的时间	varchar(24)

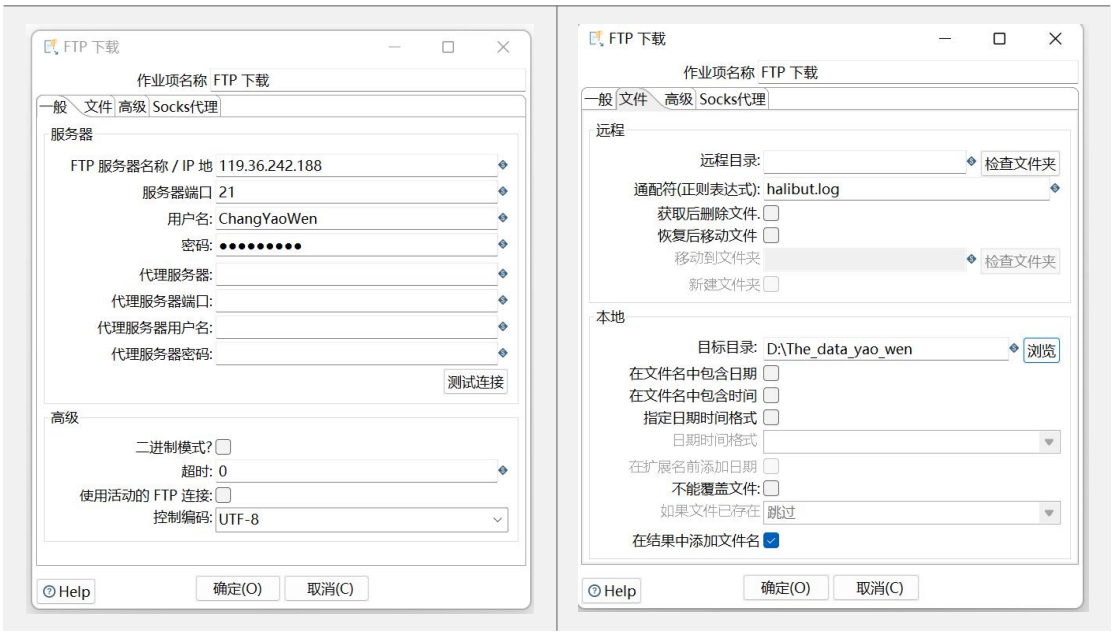
Transmission_time	传输耗时	int
connection_ip	客户端 IP	varchar(15)
file_size	文件大小	Int
file_name	文件名	varchar(1024)
transmission_type	传输类型	char(1)
action_mark	特殊标记	char(1)
transmission_direction	传输方式	char(1)
access_mode	访客类型	char(1)
user_name	用户名	varchar(1024)
Service_name	服务名	char(3)
authorization_method	授权方式	int
ip_authorization	IP 已认证标志	char(1)
completion_status	传输状态	char(1)

Table3. 数据表与数据库字段对应关系

4.2 从 FTP 下载日志数据

抽取日志数据需要连接 FTP 服务器，首先在 Spoon 工具中新建一个作业，然后在“核心对象”菜单中选择“FTP 下载”模块，双击该模块后配置模块如图 7 所示。值得注意，匹配日志文件使用了正则表达式，如果不使用正则表达式，在后续的下载日志文件中会出现错误，文件的本地保存路径为 D://The_data_yao_wen。配置如图所示：

FIG12.配置 FTP 下载　　FIG13.配置 FTP 下载



定义完成后，建立整个作业过程，如下图所示：

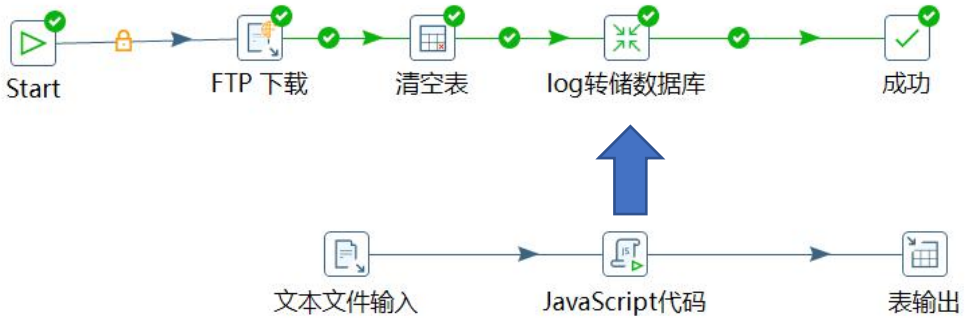


Fig14.作业流程设置

4.3 定义日志数据的转换规则

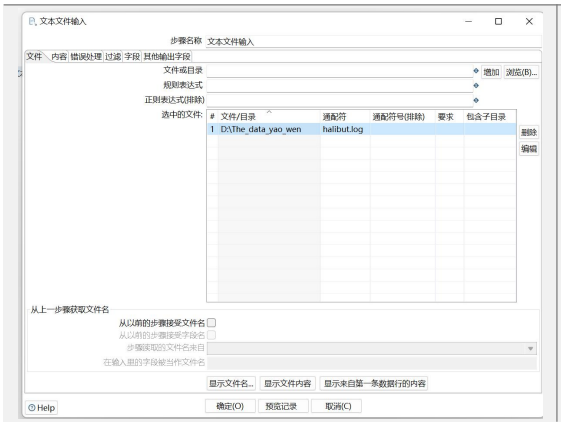
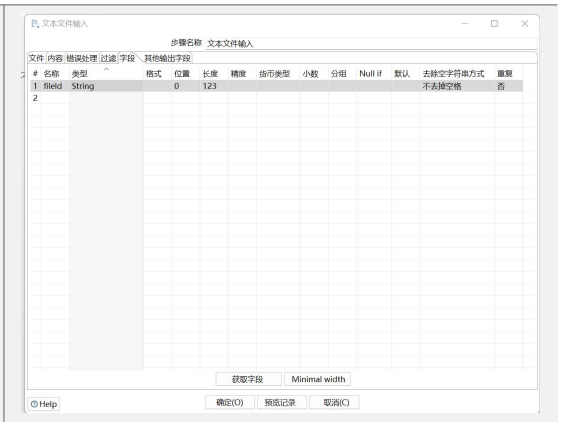
本次实验提取日志数据的转换通过先拆分然后再进行合并的方法进行日志数据的转换。该方法在读入日志文件后，以空格为标志，将日志文件中的每一条记录都拆分成 18 个小字段。由于日期字段中存在空格字符，所以在数据拆分时，日期字段被拆分成了 5 份。因此，拆分后需要使用 JavaScript 脚本对字段进行合并操作，将被拆分的日期字段复原。然后，删除多余字段并对所有字段重命名，增加可读性。然后存储到数据库中。

转换日志数据需要连接数据库，首先在 Spoon 工具中新建一个转换，命名为 log 转储数据库，并拉取文本文件输入，JavaScript 代码和数据库插入/更新三个流程并加以连线。如下图所示：

Fig15.log 转储数据库

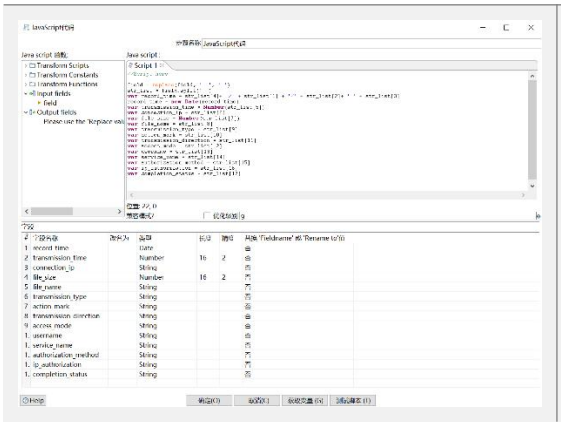
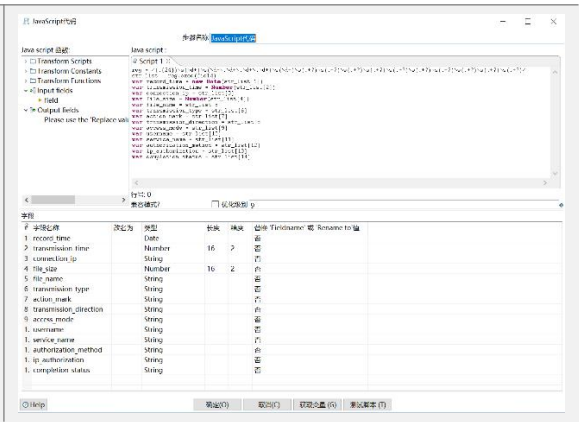
随后打开文本文件输入项，设置文本输入和字段，具体如下图所示：

FIG16.配置文本输入　　FIG17.配置 FTP 下载

打开 JavaScript 代码项，输入字段提取的 JavaScript 代码并设置输出字段。在此测试了两种提取方法，一种基于空格进行拆分提取，一种基于正则表达式提取。

FIG16.基于空格提取　　FIG17.基于正则表达式提取

或者在提取的过程中可以建立字段之间的映射关系，对于提取输入的 18 个 Field 使用“JavaScript 脚本组件”对“日志文件读取”模块拆分后的日期字段（Field 1 至 Field 5）进行合并，合并后的字段命名为 date，脚本模块内容如图 18 所示。

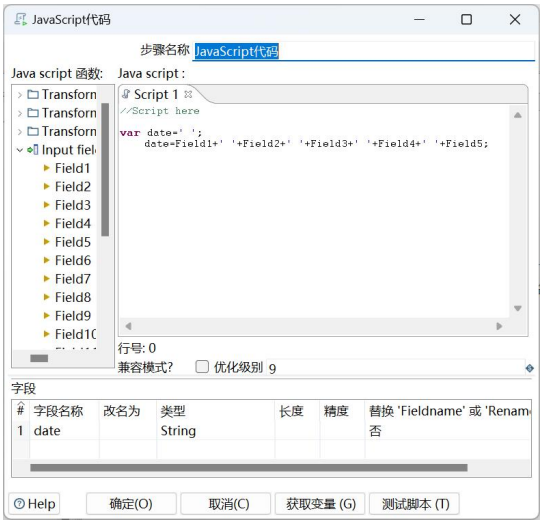


Fig18.合并 date

鉴于当前字段命名的语义信息弱，不便于后续与数据库表属性映射，且数据流中存在无

用字段，故采用“字段选择”模块进行重命名与字段删除，操作完成后，Field 1 至 Field 5 已被合并成了 date，所以在此步骤中删除，随后建立映射关系如下图所示：

Fig19.建立映射关系

至此，通过上述步骤，建立了数据的转换规则，下面开始进行数据的入库操作。

4.4 定义输出

先设置 mysql 数据库。点击主对象树中的 DB 连接，新建一个名为 kettle 的 mysql 连接，如下图所示：

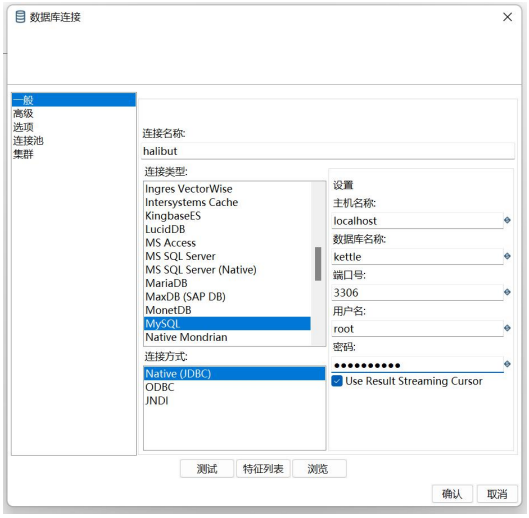


Fig20.数据库连接

打开数据库插入/更新项，设置数据库连接、查询关键字和更新字段，具体如下图所示：

Fig21.插入/更新

4.5 作业运行

整体的作业流程如图 22 所示，作业顺序遵循 ETL 步骤，即先从 FTP 服务器中获取日志、然后进行转换与装载，阅读并查看生成日志文件，发现程序成功运行，如下图所示：

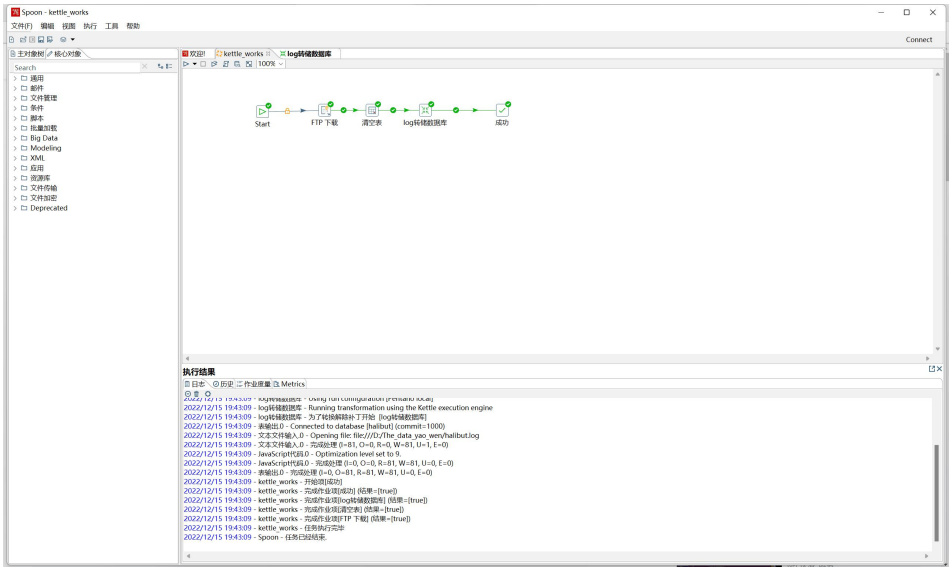


Fig22.运行成功

4.6 连接 Navicat，检查表单是否修改

在作业流程成功运行后，打开 Navicat,观察 haibut 数据表单是否成功完成修改，修改成功如下图所示：

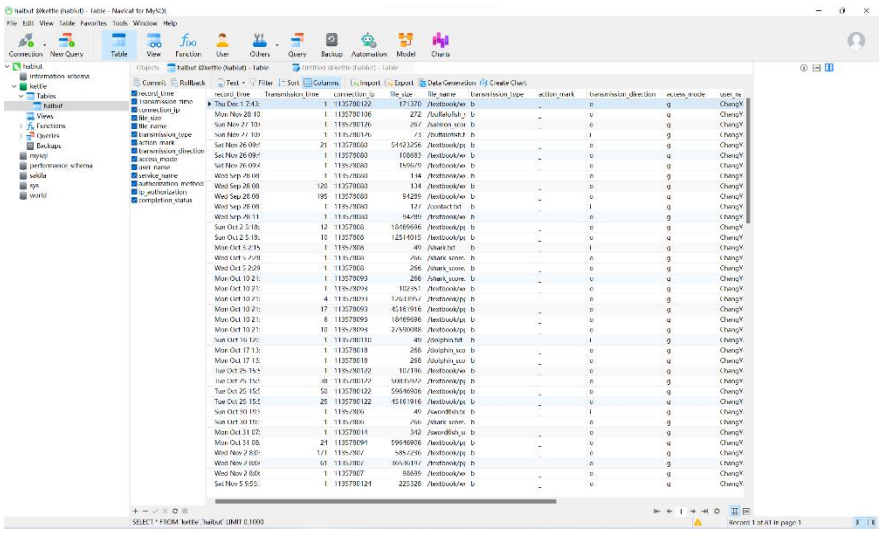


Fig23.表单修改完成

4.7 生成 SQL 文件

通过 Navicat 生成 SQL 文件，转换结果如下图：

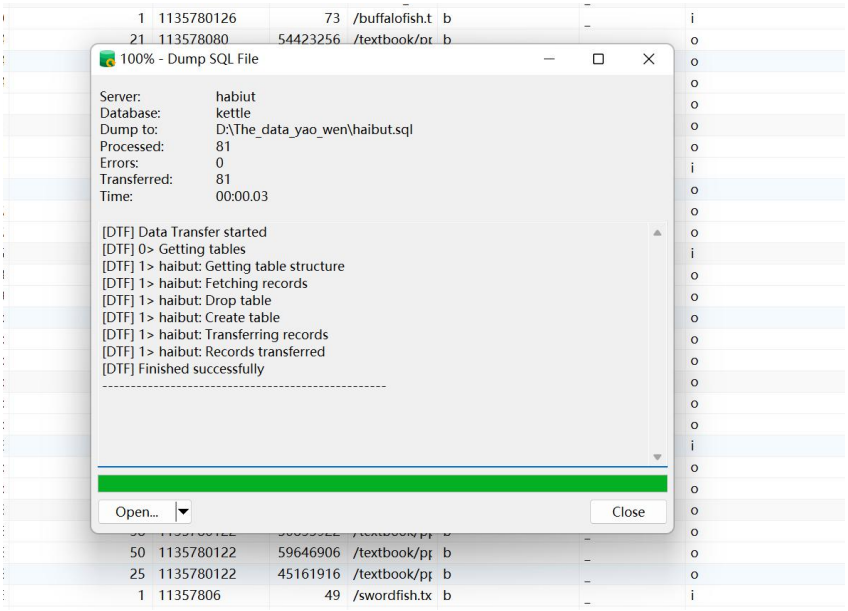


Fig24.SQL 文件生成

将生成的 SQL 文件使用 VS CODE 打开，发现一切结果正常：

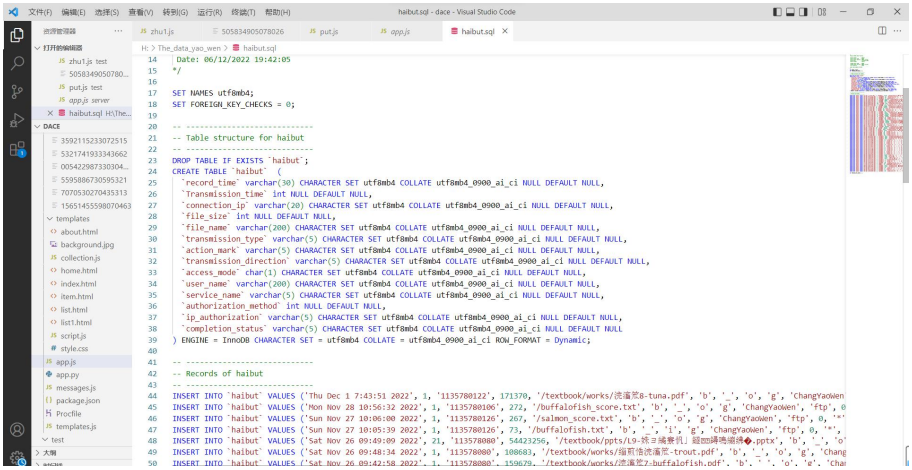


Fig25.SQL 文件正常打开

至此实习全部操作结束！

5. 实习问题

5.1 Java Script 代码报错

需要检查脚本中的语法，尤其检查是否正确的空行或者缩进。其次，脚本模块必须指定输出字段，输出字段是新定义的变量名称，如果没有指定输出字段直接运行该脚本，或者将输出字段定义为输入变量都会报错。

5.2 FTP 服务器连接错误

需要在配置 FTP 下载中，在设置文件访问路径时使用正则表达式，严格按照软件中设置的提示实现任务有助于减少错误的发生。

5.3 Spoon 工具无法连接数据库，连接信息正确

首先需要确认数据库已经安装并配置成功。其次需要使用访问数据库的 Jar 包，配置后还需要重启 Spoon 工具，再次连接数据库。经过本实验的数次尝试以及资料的查阅后，发现最好使用与版本相匹配的 Jar 包，否则还是会无法访问数据库。

6. 实习心得

本次实习，目的是使用 ETL 工具完成个人日志数据的抽取、转换、装载工作。实现对于信息系统集成与管理知识的巩固，本次实习，让我了解了 ETL 步骤以及 ETL 工具的用法，

尽管在实习的操作过程中遇到不少的困难，但我通过查阅资料与和同学交流沟通的方法，及时解决了在实习过程中遇到的困难，掌握了实习的任务，完成了实习的要求，全部实现了老师布置的任务，收获了知识。

感谢老师对于我们的教导，希望之后能够更好地使用 ETL 相关工具进行系统架构与建立系统。