

# 主观题 1：数字技术平台部署（30 分）

本任务需要在 Linux 下中完成完全分布式 Hadoop 集群搭建、Hive 安装及配置、完全分布式 Spark 集群搭建，并验证组件的可用性，请使用 root 用户完成相关配置，具体要求如下。

## 1.1 完全分布式 Hadoop 集群搭建：

（1）在 master 主节点将/data 目录下的 JAVA 安装包和 Hadoop 安装包解压到/opt/software 目录下（需自行创建/opt/software 目录），并将解压后的 JAVA 远程拷贝至 slave1、slave2 相同目录下，最后查看各节点的/opt/software 目录结构并将查看结果截图。

（2）在 3 个节点的/etc/profile 文件中配置 JDK 环境变量 JAVA\_HOME、Hadoop 环境变量 HADOOP\_HOME 和 PATH 的值，并让配置文件立即生效，之后在 master 节点使用“java -version”查看 JAVA 版本，检测 JAVA 是否安装成功，将查看 JAVA 版本结果截图。

（3）根据下表修改 core-site.xml、hdfs-site.xml、mapred-site.xml、yarn-site.xml、workers、hadoop-env.sh、yarn-env.sh 配置文件，以及 HDFS 与 YARN 的启动和关闭脚本。

表 1 Hadoop 集群部署规划

服务器	master	slave1	slave2
HDFS	NameNode		
HDFS		SecondaryNameNod	
HDFS		DataNode	DataNode
YARN	ResourceManager		
YARN		NodeManager	NodeManager
历史日志服务器	JobHistoryServer		

（4）在 master 节点上使用 scp 命令将配置完的 Hadoop 安装目录远程拷贝至 slave1 和 slave2 相同目录下，之后查看 slave1 和 slave2 的/opt/software 目录结构并将查看结果截图。

（5）在主节点格式化集群，成功格式化之后在主节点依次启动 HDFS、YARN 服务、JobHistoryServer 服务，并查看其节点进程，将查看结果截图。

（6）在 HDFS 文件系统中创建/etc 目录，之后将本地/etc/profile 文件上传至 HDFS 的/etc 目录下，并查看该目录下的文件和目录。

（7）使用 hadoop-mapreduce-examples-3.1.4.jar 包中的“wordcount”类对 HDFS 上的/etc/profile 文件内容进行单词计数，设置输出路径为“/output/”，查看最终单词计数结果中出现次数最多的 5 个单词。

## 1.2 Hive 安装及配置：

(1) 将/data 目录下的 Hive 安装包解压到/opt/software 目录下（需自行创建/opt/software 目录）。

(2) 进入 Hive 安装目录的 conf 目录，将 hive-env.sh.template 重命名为 hive-env.sh，之后查看 /etc/profile 文件中的 Hadoop 安装目录（HADOOP\_HOME），并将查看到的 HADOOP\_HOME 添加至 hive-env.sh 文件末尾。

(3) 在 Hive 安装目录的 conf 目录下新建 hive-site.xml 配置文件并添加内容。

表 2 hive-site.xml 部分参数

配置参数	描述	参数值
hive.metastore.warehouse.dir	元数据库位置	hdfs://master:8020/user/hive/warehouse
javax.jdo.option.ConnectionURL	元数据库的连接信息	jdbc:mysql://master:3306/hive?createDatabaseIfNotExist=true
javax.jdo.option.ConnectionDriverName	连接数据库驱动	com.mysql.cj.jdbc.Driver
javax.jdo.option.ConnectionUserName	连接数据库用户名称	root
javax.jdo.option.ConnectionPassword	连接数据库用户密码	123456

(4) 将/data 目录下的 MySQL 驱动 mysql-connector-java-8.0.30.jar 复制到 Hive 安装目录的 lib 目录，同时将该 lib 目录下的 jline-2.12.jar 复制到各节点的 Hadoop 安装目录的 /share/hadoop/yarn/lib/ 目录中。

(5) 删除 Hive 安装目录的 lib 目录下的 guava-19.0.0.jar 包，并将 Hadoop 安装目录的 /share/hadoop/common/lib/ 目录中的 guava-27.0-jre.jar 复制至 Hive 安装目录的 lib 目录下。

(6) 在 master 主节点的/etc/profile 文件中配置 Hive 环境变量 HIVE\_HOME 和 PATH 的值，并让配置文件立即生效。

(7) 初始化 Hive 元数据库，之后依次启动 Hadoop 集群、MySQL 服务和 Hive 元数据服务。

(8) 进入 Hive CLI 在 Hive 中创建一个名为 school 的数据库，并在该数据库下创建一个名为 student 的数据表，字段包括“id、name、gender、age”，数据类型分别为“int、string、string、int”。

(9) 先使用 insert 语句向表中插入三条测试数据，再使用 select 语句查看表数据。

### 1.3 完全分布式 Spark 集群搭建:

(1) 在 master 节点将/data 目录下的 spark-3.2.1-bin-hadoop3.2.tgz 安装包解压到/opt/software 目录下（需自行创建/opt/software 目录）。

(2) 把解压后的 spark-3.2.1-bin-hadoop3.2 文件夹更名为 spark-3.2.1。

(3) 在 master 节点修改/etc/profile 文件，设置 Spark 环境变量，并使环境变量生效。

(4) 在 master 节点上将/opt/software/spark-3.2.1/conf 目录下的 spark-env.sh.template 文件更名为 spark-env.sh。

(5) 在更名后的 spark-env.sh 文件中进行修改，需要指定 standalone 模式运行时的 Master 进程运行在 master 节点上，指定 Master 进程内部通信端口号为 7077，SPARK 的 WEB UI 端口为 8085。

(6) 在 master 节点上面修改/opt/software/spark-3.2.1/conf/workers.template 文件名为 workers 后在/opt/software/spark-3.2.1/conf/workers 文件中指定 standalone 模式运行时 Worker 进程需要分别在 master、slave1、slave2 节点上运行。

(7) 在 master 节点上面将配置的 Spark 环境变量文件及 Spark 解压包拷贝到 slave1、slave2 节点的/opt/software 路径下，并查看 slave1、slave2 的/opt/software 目录下的内容。

(8) 启动 Spark 集群，使用 jps 查看 master 节点、slave1 节点、slave2 节点的进程。

(9) Spark 集群启动后，通过“http://主机名称:8085”访问 Spark 的监控界面，查看集群各节点信息，并将查看结果截图。

#### 【说明】

(1) 进入环境后需先在 Linux 终端执行命令“initnetwork”，或者双击桌面上名称为“初始化网络”的图标，初始化实训平台网络。

(2) 提供的环境中的 master、slave1、slave2 节点已设置 SSH 免密登录，且各节点时间已同步。若要切换至 slave1 或 slave2 节点，可以打开新的 Linux 终端窗口，然后输入“ssh slave1”或“ssh slave2”即可切换到对应的节点。

(3) 安装包获取需要先在 Linux 终端使用 wget 命令获取：

```
“ wget -P /data/ http://house.tipdm.com/SZ-Competition/software/jdk-8u281-linux-x64.tar.gz ”
```

```
“wget -P /data/ http://house.tipdm.com/SZ-Competition/software/hadoop-3.1.4.tar.gz”
```

```
“                               wget                               -P                               /data/
http://house.tipdm.com/SZ-Competition/software/mysql-connector-java-8.0.30.jar ”
```

```
“wget -P /data/ http://house.tipdm.com/SZ-Competition/software/apache-hive-3.1.2-bin.tar.gz”
```

```
“wget -P /data http://house.tipdm.com/SZ-Competition/software/spark-3.2.1-bin-hadoop3.2.tgz”
```

## 主观题 2：大数据技术应用（40 分）

2、基于采集的二手房信息数据，根据题目要求运用 Python 实现二手房数据清洗与挖掘。

（1）读取文件 `SecondhandHouse.csv`。用 `null` 值替换数据中含有“暂无”两字的信息，再删除含有 `null` 的行数据。统计原数据集的行列数以及删除 `null` 值后数据集的行列数，对输出结果截图。

（2）统计行数据重复的数量，并将结果进行展示；对重复的行数据进行删除，统计删除重复数据后数据集的行列数，对输出结果截图。

（3）删除“建筑面积”列的面积单位“平米”，仅保留数值，并将其数据类型转为浮点型。删除“建筑年代”列的年份单位“年”，仅保留数值，并将其数据类型转为整数型。

（4）二手房的出售一般为旧楼房，建筑年代应小于 2022 年，保留“建筑年代”小于等于 2021 年的房屋数据。展示处理后数据集的行列数，对输出结果截图。

（5）提取“户型”列中的室、厅、卫的数量，如“3 室 1 厅 1 卫”则提取“3，1，1”，分别存入“室”、“厅”和“卫”列。

（6）使用当前年份（2022 年）减去建筑年份获取房龄，并将结果存入“房龄”列。

（7）完成上述步骤后，将处理后的二手房数据集 `DataFrame` 以 `csv` 格式导出至 `/data/result` 目录。查看 `csv` 文件并截图。

### 【文件读取路径】

下载题目附件中的数据，并上传到实训平台中。

### 【文件读取路径】

`“/data/SecondhandHouse.csv”`

3、基于二手房房价数据，根据题目要求运用数据挖掘与可视化知识对数据进行统计与基本图形绘制。

（1）绘制建筑面积与房价分布情况的散点图：设定 `x` 轴数据为建筑面积，`y` 轴数据为总价；`x` 轴与 `y` 轴标签（`xlabel` and `ylabel`）分别为“建筑面积（平米）”和“总价（万）”；图形标题（`title`）为“二手房建筑面积与房价的关系分析”。

（2）运用 `seaborn` 库绘制不同装修程度的二手房房价的分组箱线图：设定 `x` 轴数据为装修程度，`y` 轴数据为总价；`x` 轴与 `y` 轴标签（`xlabel` and `ylabel`）分别为“装修程度”和“二

手房房价（万）”；图形标题（title）为“不同装修程度的二手房房价分组箱线图”。

（3）统计“楼层”、“电梯”和“学校”列不同情况的二手房数量，并绘制二手房楼层、电梯和学校不同情况占比的饼图：将画布分成 1 行 3 列的 3 个子图，子图 1、2、3 分别绘制楼层、电梯和学校不同情况占比的饼图；对每个子图设定标题，分别为“楼层”、“电梯”和“学校”；令每个饼图展示各自的百分比（autopct）和标签（labels），其中百分比保留小数点后 1 位（如 12.3%）。

（4）将“总价”列的数值按照指定区间划分至不同等级，统计不同等级的二手房出售数量，根据统计结果，绘制不同等级的二手房出售数量分布的柱状图：设定 x 轴数值为房价等级，y 轴数值为二手房出售数量；房价等级沿 x 轴正方向进行排序（从小到大）；设定 x 轴刻度标签（xticks）为具体的数值区间；设定柱状图显示不同等级的二手房出售数量。

数值区间	≤50	(50,65]	(65,80]	(80,95]	(95,110]
等级	1	2	3	4	5
数值区间	(110,125]	(125,140]	(140,155]	(155,170]	>170
等级	6	7	8	9	10

【文件读取路径】

下载题目附件中的数据，并上传到实训平台中。

【文件读取路径】

“/data/SecondhandHouse view.csv”

### 主观题 3：人工智能技术应用（30 分）

4、基于兰州二手房房价数据，根据题目要求运用机器学习知识实现数据建模与评估。

（1）特征编码：将“朝向”，“楼层”，“装修”，“电梯”，“产权性质”，“建筑结构”，“建筑类别”和“区域”8 列数据的类型由字符型转化为数值型，如“电梯”列，原{'有','无'}转化为{1,0}。特征编码后的数据保存为 CSV 文件，保存路径设定为“/data/result5.csv”。

（2）数据集划分：提取“总价”列作为标签，其他列作为特征；按 8:2 的比例划分训练集与测试集，并展示划分后训练集与测试集的数据性质（shape）。

（3）模型训练与评估：使用 sklearn 估计器构建回归模型，放入训练集进行训练；放入测试集特征实现模型预测；结合预测结果与测试集标签，运用 sklearn.metrics 模块的 r2\_score 函数评估模型优劣，并展示评估结果。

【文件读取路径】

下载题目附件中的数据，上传到实训平台中

【文件读取路径】

“/data/SecondhandHouse\_view.csv”

河北省职业院校技能大赛数字技术应用