**任务1 hadoop组件安装**

**一、hadoop全分布式安装**

1、 解压 JDK 安装包到/usr/local/src路径，并配置环境变量；截取环境变量配置文件截图。

2、创建ssh密钥，实现主节点与从节点的无密码登录，通过主节点登录其中一个从节点并截图；

3、修改core-site.xml、hdfs-site.xml、mapred-site.xml、yarn-site.xml，分别对这四个配置文件的主要配置参数进行截图。

4、配置slaves文件，对文件的主要配置参数进行截图。

5、启动hadoop集群，对master、slave1、slave2上的相关进程进行截图。

**二、flume安装**

1、在master节点上设置Flume环境变量，截取环境变量配置截图

2、修改flume-env.sh并截图

3、查询flume版本并截图

**三、zookeeper安装**

1、在master节点上修改配置文件zoo.cfg并截图

2、修改/etc/profile,设置zookeeper的配置参数并截图

3、启动zookeeper，查看其状态并截图

**任务2 数据采集**

**一、爬取book全站信息**

登录网站分析<http://books.toscrape.com/>分析网页结构，爬取书本的书名、价格、简介。

要求：

1、在mysql数据库中创建表book，用于记录书本的书名、价格、简介。

2、使用scrapy/requests框架爬取书本信息，并将这些信息保存到数据库的表book中，爬取的书本数量不少于50。



**二、爬取腾讯招聘信息**

登录网站<https://careers.tencent.com/search.html> 分析网页结构，爬取招聘信息的标题和内容。要求：

1、使用scrapy/requests爬取指定字段的信息，并将其分别保存到careers.csv和careers.json中，爬取招聘信息不少于100条。



**任务3 大数据清洗（spark + hdfs）**

**已知data.txt文件记录了tomcat的运行日志信息**

访问ip 访问时间 请求类型 url 协议名称 http返回码 响应时长

172.18.35.1 - - [2013-03-04 23:38:27] "POST /service/1.htm HTTP/1.0" 200 1

172.18.35.2 - - [2013-03-04 23:38:27] "GET service/1.htm HTTP/1.0" 200 1

172.18.35.2 - - [2013-03-05 23:38:27] "GET service/2.htm HTTP/1.0" 200 2

172.18.35.1 - - [2013-03-05 23:38:27] "POST / service/12.htm HTTP/1.0" 200 2

172.18.35.3 - - [2013-03-06 23:38:27] "GET / HTTP/1.0" 200 2

172.18.35.3 - - [2013-03-06 23:38:27] "GET / HTTP/1.0" 200 2

172.18.35.4 - - [2013-03-07 23:38:27] "GET / service/2.htm HTTP/1.0" 404 3

172.18.35.5 - - [2013-03-07 23:38:27] "POST / service/2.htm HTTP/1.0" 200 3

172.18.35.6 - - [2013-03-08 23:38:27] "POST / service/3.htm HTTP/1.0" 200 3

172.18.36.1 - - [2013-03-08 23:38:27] "GET / service/1.htm HTTP/1.0" 200 4

172.18.35.8 - - [2013-03-08 23:38:27] "GET /html/notes/1.html HTTP/1.0" 200 5

172.18.35.1 - - [2013-03-09 23:38:27] "POST html/notes/1.html HTTP/1.0" 404 2

172.18.35.1 - - [2013-04-04 23:38:27] "GET html/notes/2.html HTTP/1.0" 200 1

172.18.37.1 - - [2013-04-04 23:38:27] "GET html/notes/2.html HTTP/1.0" 200 1

172.18.38.1 - - [2013-04-05 23:38:27] "POST /html/notes/1.html HTTP/1.0" 404 1

1. 172.18.39.1 - - [2013-04-06 23:38:27] "GET/ HTTP/1.0" 200 1

**1）统计每个url的访问数量**

输出结果举例：

POST /service/1.htm 3 【本条数据表示：发送post请求访问/service/1.htm的次数为3（一个url可能会发送post请求，也可能会发送get请求）】

GET /service/1.htm 2 【本条数据表示：发送get请求访问/service/1.htm的次数为2，（一个url可能会发送post请求，也可能会发送get请求）】

GET / 4

**2）将每一天访问的url放到不同的目录下（目录名称为日期）**

输出结果举例：

目录名称：2013-03-04

数据内容：

172.18.35.1 - - [2013-03-04 23:38:27] "POST /service/1.htm HTTP/1.0" 200 1

172.18.35.2 - - [2013-03-04 23:38:27] "GET service/1.htm HTTP/1.0" 200 1

目录名称：2013-03-08

数据内容：

172.18.35.6 - - [2013-03-08 23:38:27] "POST / service/3.htm HTTP/1.0" 200 3

172.18.36.1 - - [2013-03-08 23:38:27] "GET / service/1.htm HTTP/1.0" 200 4

172.18.35.8 - - [2013-03-08 23:38:27] "GET /html/notes/1.html HTTP/1.0" 200 5

**3） 计算每一个url响应的平均时间，并按从大到小的顺序排列**

输出结果举例：

/service/1.htm 3.1

/ service/3.htm 2.9

/html/notes/1.html 3.5

**4） 找出所有404的url**

输出结果举例：

172.18.35.4 - - [2013-03-07 23:38:27] "GET / service/2.htm HTTP/1.0" 404 3

172.18.35.1 - - [2013-03-09 23:38:27] "POST html/notes/1.html HTTP/1.0" 404 2

**任务4 大数据可视化**

已知若干班级的学习成绩表，完成以下任务：

1）每个班级的总人数比例（饼状图）

2）每个班级在4个学期的公共课1的平均成绩（柱状图）

（横坐标为班级）

3）软件1801所有学生在第4学期的所有课程的成绩（多条折线图，一条折线图代表一门课的成绩）

