|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **□ 수행평가 - 빅데이터를 활용한 IoT 시스템 개발**  **(feat.스마트시티프로젝트)** | | | | | | |
|  |  |  | |  | |  |
| **과정명** | | 빅데이터를 활용한 IoT시스템 개발(feat. 스마트시티 프로젝트) | | | | |
| **교과목명** | | 빅데이터저장및분석시스템구축기술 | | **훈련교사** | | 이진만 |
| **과정명** | | Linux OS 이해  빅데이터 수집시스템 개발  빅데이터 저장시스템 개발  빅데이터 분석시스템 개발 | | | | |
| **수행날짜** | | 2020.01.13 | 훈련생명 | | 유한솔 | |
| **과제개요** | | | | | | |
| 1. Linux OS 를 설치 하고 빅데이터 시스템 환경을 구축 하시오 2. 데이터 처리 및 분석을 위한 HIVE를 설치 하시오 3. 각 시스템의 시스템 구조도를 작성 하시오 4. Airline\_delay 데이터를 빅데이터 시스템에 저장 하시오. 5. Java Application을 통해 데이터를 분석 하시오 | | | | | | |
| **Linux**  **설치 및 환경설정**  1. vmware Workstation player 무료 버전  2. C:\Program Files (x86)\VMware\VMware Player  vmnetcfg 파일 설치 및 실행을 통해 IP 주소 변경 가능  3. 변경 주소는 192.168.111.1  **가상의 컴퓨터 생성**  1. create a new Virtual Machine  2. Linux//CentOS 7 64-bit  3. Edit virtual machine settings  memory 4GB  Processer(늘리면 빨라짐) 8  CD/DVD CentOS 설치 CD 삽입 후 후에 제거  4. 가상의 브라우저 환경설정  1. 키보드 : 한국어/영어(미국)  2. 소프트웨어 : 개발 및 창조를 위한 워크 스테이션  3. 네트워크 및 호스트 이름 : 활성화  4. 설치 대상 : 파티션을 설정합니다.  표준 파티션 선택  (추가) swap/2G  하드디스크의 2GB를 메모리화 하겠다  (추가) / / (메모리 자동 설정) : root 설정  5. 설치 중 Root와 사용자 설정  root/111111(관리자)  centos/111111  6. 라이센스 정보 동의  7. Kdump 비활성화  5. 업데이트 비활성화  1. 시스템 도구 -> 소프트웨어  최신 패키지만 **비활성화**  2. 시스템 도구 -> 소프트웨어 공급원  업데이트 확인 **하지 않기**  3. YUM 명령 중 업데이트 비활성화  *YUM*이란*?*  CentOS 소프트웨어를 설치할 때 사용하는 명령어  1. cd /etc/yum.repos.d/ (폴더 이동)  2. ls (파일 확인)  3. gedit CentOS-Base.repo  gedit CentOS-Source.repo (파일 편집)  4. # released updates 항목 지워주기  5. mv CentOS-Base.repo CentOS-Base.repo.bak (기존 저장소 백업)  6. wget http://download.hanbit.co.kr/centos/7/CentOS-Base.repo (새 저장소 다  운로드)  7. chmod 644 \*(권한 변경)  8. rm \*.repo~ (repo~가 딸린 필요없는 파일들 삭제)  9. yum clean all (저장소 초기화)  4. IP 주소 변경  1. cd /etc/sysconfig/network-scripts/  2. ls  3. gedit ifcfg-xxxxxxx (ls로 확인한 파일 편집)  4. 파일 편집으로 Server에 고정IP 할당  BOOTPROTO=none  IPADDR=192.168.111.100  NETMASK=255.255.255.0  GATEWAY=192.168.111.2  DNS1=192.168.111.2  5. 설정 내용 적용 명령어 실행  systemctl restart network  이후 ifconfig 로 변경된 IP 확인  6. 보안 설정 해제  gedit /etc/sysconfig/selinux  SELINUX='disabled'  7. host 이름 설정  hostname (해당 서버의 이름 확인)  hostnamectl set-hostname server1 (해당 서버의 이름을 server1로 바꾸겠다)  gedit /etc/hosts (해당 서버가 다른 IP 주소를 가진 서버를 인식할 수 있도록  IP주소와 서버이름 입력)  ping (서버이름) -> (서버이름)에게 응답 요청하는 것  **Server Clone 설정하기**  1. 이미 만든 server의 파일 복사, 붙여넣기  2. 폴더의 configuration 파일의 displayName을 원하는 서버 이름으로 변경  3. Open Virtual machine : 복사한 서버 열어주기  4. Virtual machine setting : Network Adapter->> macAddress(generate 하기)  5. 서버 실행(I moved it 선택)  6. IPADDRESS 변경  7. hostname 변경  **명령어 연습하기**  현업에서는 UI가 없는 화면에서 리눅스를 이용하기 때문에  gedit(편집 명령어) 대신 vi 명령어를 이용하자  UI 화면 설정  Ctrl + Alt + F5 : UI가 없는 검정 터미널 화면  Ctrl + Alt + F1 : UI가 설정된 화면으로 돌아가기  su : (Switch User)  su - : 사용자 뿐만 아니라 환경까지 이동(변경 후 변경된 사용자의 홈으로 이동)  history -c : 명령어 내역 삭제  vi 에디터 실행 => 명령 모드 => esc 키를 통해 **입력 모드**와 **라인 명령모드**를 번갈아가며 쓸 수 있다.  **키 설명**  i 입력모드, 현재 커서의 앞에 입력  a (append)현재 커서의 뒤에 입력  o 현재 커서의 다음 줄에 입력  s 현재 커서 위치의 한 글자를 지우고 입력  I 현재 커서의 줄 맨 앞에서 입력  A 현재 커서의 줄 맨 마지막에서 입력  O 현재 커서의 이전 줄에 입력  S 현재 커서의 한 줄을 지우고 입력  h,j,k,l 순서대로 ← ↓ ↑ →  gg 제일 첫 행으로 이동  G 제일 끝 행으로 이동  :(숫자) enter 해당 숫자의 행으로 이동  : set nu 각 행마다 숫자가 표시되도록 함  cw " " 속의 내용을 삭제(커서의 위치를 포함한 뒷부분의 내용이 삭제  됨)  u 되돌리기  Ctrl +R 다시하기  shift + ~ 대소문자 변경  x 커서가 위치한 글자 삭제  X 커서가 위치한 앞 글자 삭제  dd 현재 커서의 행 삭제  (숫자)dd 숫자만큼의 행 삭제  yy 현재 커서가 있는 행을 복사  (숫자)yy 숫자만큼의 행 복사  p 복사한 내용을 현재 행 이후에 붙여넣기  P 복사한 내용을 현재 행 이전에 붙여넣기  : q! 편집한 내용을 저장하지 않고 종료  :%s/기존문자열/새문자  열  문자열 치환하기  **CD 연결하기**  1. /media/cdrom 폴더를 만든다  2. 물리적인 cdrom을 위의 폴더로 mount한다.  3. cdrom에 있는 RPM-\* 파일들을 /에 복사한다.  4. 물리적인 cdrom을 umount 한다.  1. mkdir mycdrom  mycdrom 파일 생성  2. ls  파일 확인  3. mount /dev/cdrom /root/mycdrom  device/cdrom의 CD를 방금 만든 파일에 연결  4. cd mycdrom  mycdrom으로 이동  5. ls  연결된 CD를 확인해보자  6. cd Packages/  CD 구성품 중 Packages 폴더로 이동  7. ls mc\*  그중 mc~로 시작하는 파일을 목록 출력  8. cp mc-4\* ~  root 파일로 복사하자  9. cd  root 홈화면으로 이동  10. umount /dev/cdrom  복사가 완료된 것을 확인했다면 연결을 해제  **기타 명령어**  shutdown -P +3  이 프로그램을 3분 뒤 종료해랏!  shutdown -c  종료 명령어 취소요..  man ls  명령어 목록 출력  **서버에 자바, 이클립스, 톰캣 설치하기**  1. JAVA  1. tar xvf jdk-8u221-linux-x64.tar.gz  2. mv jdk1.8.0\_221 jdk1.8.0  3. cp -r jdk1.8.0 /etc  4. cd /usr/bin  5. ls -l java  6. rm java  7. ln -s /etc/jdk1.8.0/bin/java java  8. java -version  9. ls -l java  2. Eclipse  1. tar xvf eclipse-jee-oxygen-3a-linux-gtk-x86\_64.tar.gz  2. cp -r ./eclipse /etc  3. cd /etc/eclipse  4. ls  5. cd /usr/bin  6. ln -s /etc/eclipse/eclipse eclipse  7. ls -l eclipse  8. eclipse -version  3. Tomcat  1. tar xvf apache-tomcat-9.0.22.tar.gz  2. ls  3. cd apache-tomcat-9.0.22/  4. ls  5. cd conf  6. ls  7. vi server.xml  8. Port 번호 80으로 바꿔주기  9. cd /bin  10. ls  11. cd 다운로드  12. cp -r apache-tomcat-9.0.22 /etc  13. cd /etc/apache-tomcat-9.0.22/  14. ls  15. cd /bin  16. firewall-config  17. cd  18. cd /usr/bin  19. ln -s /etc/apache-tomcat-9.0.22/bin/startup.sh starttomcat  20. ls -l starttomcat  21. ln -s /etc/apache-tomcat-9.0.22/bin/shutdown.sh stoptomcat  22. ls -l stoptomcat  yum -y install system-config-date  ++날짜 설정 명령어  find /root/temp -name "\*.txt" -exec rm {} \;  ++ /root/temp/ 경로에 있는 .txt 확장자를 가진 모든 파일들을 찾아서 삭제하라  MySQL 연습용 테이블 만들기  create database shop;  use shop;  creat table USERS(  id varchar2(10) primary key,  pwd varchar2(10),  name nvarchar2(10),  age number(3)  );  insert into users values('id01','pwd01','귤',10);  insert into users values('id02','pwd02','사과',10);  insert into users values('id03','pwd03','복숭아',10);  select \* from users;  oracle 환경변수 설정  . /u01/app/oracle/product/11.2.0/xe/bin/oracle\_env.sh  **Maria DB와 MySQL 설치하기**  1. yum -y remove mariadb-libs  리눅스에 이미 설치되어 있는 마리아 디비 라이브러리를 삭제해준다.  2. yum -y localinstall Maria\*  yum 명령어로 Maria DB 설치에 필요한 세가지 파일을 설치해준다.  3. systemctl restart mysql  4. systemctl status mysql  5. firewall-config  6. mysql  7. mysqladmin -u root password '111111'  8. mysql -u root -p  9. mysql -h 192.168.112.102 -u root -p  10. mysql -h 192.168.112.102 -u muser -p  **오라클 설치하기**  1. unzip oracle-xe-11.2.0-1.0.x86\_64.rpm.zip  2. cd Disk1  3. swapon -s  4. dd if=/dev/zero of=/swapfile bs=1024 count=4194304  오라클은 설치 및 실행에 필요한 하드 디스크 용량이 4GB 이기 때문에,  처음 설정해주었던 여분의 하드 디스크 메모리 용량을 추가해주어야 한다.  5. mkswap /swapfile  swapfile을 만든다  6. swapon /swapfile  7. cd /etc/rc.d  8. chmod 755 rc.local  해당 파일의 권한 변경  9. vi rc.local  10. reboot  11.  621  622 systemctl status mysql  623 chkconfig mysql on  624 firewall-config  625 mysql  626 cd  627 mysqladmin -u root password '111111'  628 mysql -u root -p  629 history  630 mysql -u root -p  631 mysql -h 192.168.112.102 -u root -p  632 mysql -h 192.168.112.102 -u muser -p  633 history  634 mysql -h 192.168.112.102 -u muser -p  635 mysql  636 mysql -u root -p  637 use mysql  638 use sql\  639 mysql -u muser -p  640 cd  641 cd 다운로드  642 ls  643 unzip oracle-xe-11.2.0-1.0.x86\_64.rpm.zip  644 ls  645 cd Disk1  646 ls  647 dd  648 df  649 swapon -s  650 dd if=/dev/zero of=/ swapfile bs=1024 count=4194304  651 dd if=/dev/zero of=/swapfile bs=1024 count=4194304  652 mkswap /swapfile  653 swapon /swapfile  654 swapon -s  655 cd /etc/rc.d  656 ls  657 chmod 755 rc.local  658 ls  659 vi rc.local  660 reboot  661 swapon -s  662 cd 다운로드  663 ls  664 cd Disk1  665 ls  666 yum -y localinstall ora\*  667 ls  668 service oracle-xe configure  669 /etc/init.d/oracle-xe status  670 . /u01/app/oracle/product/11.2.0/xe/bin/oracle\_env.sh  671 cd /etc/bashrc  672 vi /etc/bashrc  673 reboot  674 history  shop 이름으로 workspace create  서버 네트워크 에딧 설정 들어가서  네트워크 Bridged 설정  vi /etc/sysconfig/network-scripts/ifcfg-eno16777736  ipaddres~dns  ip : 70.12.113.XXX  net:255.255.255.0  gw:70.12.113.1  dns:168.126.63.1  systemctl restart network  ping으로 확인해보기  ls /dev/sd\*  00:50:56:3F:72:CF  00:50:56:26:C1:36  **하둡과 빅데이터**  1. 개요  **하둡이란?**  기존 데이터베이스 관리도구  데이터를 수집, 저장, 관리, 분석할 수 있는 역량을 넘어서는  대량의 정형 또는 비정형 데이터 집합으로부터  가치를 추출하고 결과를 분석하는 기술  2. 정의  정보 자산을 효율적으로 이용하기 위한 기술들을  포괄하여 아우르는 말  비정형 데이터를 분석하여 활용가능한 정보를 추출하는 것  3. 비정형 데이터란?  페이스북의 다양한 정보들  **항목 의미**  Volume(양) 대용량의 데이터  Velocity(입출력  속도)  큰 용량의 데이터를 빨리 처리해야 함  Variety(다양성) 계량화 및 수치화가 어려운 비정형적 데이터를 포함함  Veracity(정확  성)  분석에서 목적에 맞는 데이터를 선별하고 수집하는 것이 분석 결과의 정  확성에 영향을 미침  Value(가치) 빅데이터를 통해 어떤 문제를 해결할 수 있는가?  빅데이터  3V 모델이다!(Volume, Velocity, Variety)  다양한 종류의 수많은 데이터를 빠르게 처리할 수 있다.  필요성  다변화된 현대 사회를 더욱 정확하게 예측하여 효율적으로 작동케 하고 개인화된 현대  사회 구성원마다 맞춤형 정보를 제공, 관리, 분석하여 과거에는 불가능했던 기술을 실  현  문제점  사생활 침해 및 정보 유출 측면, 개인정보 보호와 활용을 절충한 입법적 장치 필요  해결방안? 개인을 식별할 수 없는 정보를 활용하여 서비스를 제공하는 방법  **항**  **목**  **의미**  기  획  어떤 데이터를 수집 및 분석할 것인지 계획을 수립하고 추진하는 분야  처  리  데이터 수집,처리,시각화를 위해 사용되는 기술//IOT Seonsor,Haddop,D3  분  석  데이터간의 상관관계를 통해 의미있는 결론을 도출하는 기술(텍스트 마이닝, 오피니언  마이닝, 패턴 분석 등... )R, Python, 엑셀  **빅데이터 처리 기법**  기술구성  처리 기술  대규모의 정형/비정형 데이터를 처리하는 데 있어 가장 기본적인 분석 인프라  하둡, NoSQL, SPARK  분석 기술  기존 기술  데이터 마이닝, 기계학습, 자연 언어 처리, 패턴 인식 ...  **신규 기술**  텍스트 마이닝  비/반정형 텍스트 데이터에서 자연 언어 처리 기술에 기반하여 유용한 정보를  추출, 가공  오피니언 마이닝  소셜미디어 등의 정형/비정형 텍스트의 긍정, 부정, 중립의 선호도를 판별  소셜 네트워크 분석  소셜 네트워크의 연결 구조 및 강도 등을 바탕으로 사용자의 명성 및 영향력을  측정  군집 분석  비슷한 특성을 가진 개체를 합쳐가면서 최종적으로 유사 특성의 군집을 발굴  00:50:56:3F:2C:B3  **하둡 서버 셋팅**  1. Network Setting  2. hostname, etc/hosts  3. JDK& profile Setting  vi /etc/profile  JAVA\_HOME=/etc/jdk1.8.0  CLASSPATH=/etc/jdk1.8.0/lib  PATH=.:$JAVA\_HOME/bin:$PATH  export JAVA\_HOME CLASSPATH  . /etc/profil  4. Hadoop download  tar xvfz hadoop-1.2.1.tar.gz  cd /usr/local  cp -r /root/다운로드/hadoop-1.2.1 .  5. Hadoop profile Setting  cd /usr/local  ls  mv /etc/jdk1.8.0/ .  mv /etc/eclipse/ .  vi /etc/profile  cd /usr/bin  rm java  rm eclipse  cd  . /etc/profile(변경된 profile 환경 설정을 적용)  6. SSH  ssh hadoopserver1  ssh-keygen -t dsa -P '' -f ~/.ssh/id\_dsa  hadoopserver에 대한 Private/Public Key 생성  ls -a  만들어진 ssh 파일을 확인한다.  cd .ssh:  ssh 파일로 이동  ssh-copy-id -i id\_dsa.pub root@hadoopserver1  id\_dsa 에 대해 컴퓨터 자체 로그인을 할 때 key 값이 자동으로 매칭 되도록 authorized  keys 가 만들어지도록 하는 명령문  =>이후 ssh hadoopserver1으로 로그인 시 비밀번호를 묻지 않음  7. hadoop-env.sh  export JAVA\_HOME=/usr/local/jdk1.8.0  export HADOOP\_HOME\_WARN\_SUPPRESS="TRUE"  8. hadoop-1.2.1/conf/\*.xmlexi  (core-site, hdfs-site,mapred-site)  JAVA\_HOME=/usr/local/jdk1.8.0  CLASSPATH=$JAVA\_HOME/lib  HADOOP\_HOME=/usr/local/hadoop-1.2.1  PATH=.:$JAVA\_HOME/bin:$HADOOP\_HOME/bin:$PATH  export JAVA\_HOME CLASSPATH HADOOP\_HOME  1. core-site.xml  2. hdfs-site  3. mapred-site  9. hadoop namenode -format  10. start-all.sh //stop-all.sh  name, data, tmp 파일이 생기지 않을 경우 폴더를 모두 삭제 후 7, 8번 점검, 9번부터 다시 만들기  1월 7일  하둡 파일 관리 서버 들어가는 주소  http://hadoopserver1(NameNode 서버 이름):50070  <configuration>  <property>  <name>fs.default.name</name>  #how many file will you copy  <value>hdfs://hadoopserver1:9000</value>  </property>  <property>  <name>hadoop.tmp.dir</name>  #where to place name directory  <value>/usr/local/hadoop-1.2.1/tmp</value>  </property>  <configuration>  <property>  <name>dfs.replication</name>  #how many files will you copy  <value>1</value>  </property>  <property>  <name>dfs.name.dir</name>  #where to place name directory  <value>/usr/local/hadoop-1.2.1/name</value>  </property>  <property>  <name>dfs.data.dir</name>  <value>/usr/local/hadoop-1.2.1/data</value>  </property>  </configuration>  <configuration>  <property>  <name>mapred.job.tracker</name>  <value>hadoopserver1:9001</value>  </property>  </configuration>  하둡에 파일 넣었다가 빼기  1. hadoop fs -put CHANGES.txt mydata/CHANGES.txt  하둡에 해당 파일을 어느 폴더의 무슨 이름으로 넣겠다  2. hadoop jar hadoop-examples-1.2.1.jar wordcount mydata/CHANGES.txt wordcount\_output  3. hadoop fs -mkdir mydata2  4. cp anaconda-ks.cfg a.txt  5. zip a.txt.zip a.txt  6. hadoop fs -put a.txt.zip /mydata2  7. hadoop fs -get mydata2/a.txt.zip aa.zip  가상 분산 모드 예제 연습하기  hadoopserver1 -namenode -201  second -secondary namenode -202  data1 -datanode -203  data2 -datanode -204  각 서버에게 할당했던 IP 주소  192.168.111.202 name  192.168.111.203 hadoopserver2  192.168.111.204 data1  192.168.111.205 data2  **HDFS 구축**  1. name server setting  앞서 만든 hadoopserver1의 복사본.  2. 각 서버(hadoopserver2, data1, data2)에 SSH 연결(책 51pg)  3. name server : hadoop 및 기타 셋팅  ㅇㅇ  4. name 서버에서 /usr/local/hadoop-1.2.1/conf  master,slaves 파일 편집  ex) master => hadoopserver2  slaves => hadoopserver2, data1, data2  hdfs-site.xml 에서  name .dir설정은 (메타 데이터)  data 폴더는 물리적으로 실제 데이터가 들어가는 폴더  core-site, hdfs-site, mapred-site 재설정  5. JDK 설정  6. hadoop과 jdk 압축ls  7.  tar cvfz hadoop-1.2.1.tar.gz hadoop-1.2.1/  8. /etc/profile, hadoop, jdk 를 각 시스템에 전송  scp /etc/profile root@hadoopserver2:/etc  scp hadoop-1.2.1.tar.gz root@data1:/usr/local  scp jdk1.8.0.tar.gz root@data2:/usr/local  9. hadoop, jdk의 압축 해제  ssh root@data1 "cd /usr/local; tar xvf hadoop-1.2.1.tar.gz; rm -rf hadoop-1.2.1.tar.gz"  ; 으로 명령을 덧붙여서 여러개 사용 가능  " " 따옴표 안에 실행할 명령을 차례로 입력  10. hadoop 1.2.1 파일로 이동,  hadoop namenode -format  11. start-all.sh/stop-all.sh  12. jps 실행 시 3개 뜸  Jpsssh  JobTracker  NameNode  13. hadoopserver2, data1, data2 에서 jps 실행시에 잘 작동되려면  systemctl stop firewalld  systemctl disable firewalld  방화벽 설정 해주기  00:50:56:3F:2C:B3  예제 풀어보기  1. wordcount를 실행해보자(pg63)  hadoop fs -put conf/hadoop-env.sh conf/hadoop-env.sh  hadoop jar hadoop-examples-\*.jar wordcount conf/hadoop-env.sh  wordcount\_output  hadoop fs -cat wordcount\_output/part-r-00000  2. /boot 폴더에 있는 파일을 hadoop 시스템에 put, get  3. 모니터링 시스템을 통해 각 시스템의 상황을 모니터링 하기  server1을 이용하여 각 컴퓨터를 셋팅 한다.  1. 브릿지를 이용하여 IP 셋팅  2. namenode, second namenode(datanode), datanode1, datanode2  3. 대용량 파일 입력과 wordcloud를 실행해본다.  **HIVE 설치하기**  hive의 메타 데이터를 저장해줄 공간 마련을 위해 MariaDB(MySQL)을 설치해준다.  1. yum -y remove mariadb-libs  2. yum -y localinstall Maria\*  3. systemctl restart mysql  4. systemctl status mysql  5. chkconfig mysql on  6. firewall-config  7. mysqladmin -u root password '111111'  8. mysql -u root -p(mysql 접속)  grant all privileges on *.* to 'hive'@'localhost' identified by '111111';  create database hive\_db;  grant all privileges on hive\_db.\* to 'hive'@'localhost' identified by '111111';  grant all privileges on hive\_db.\* to 'hive'@'%' identified by '111111' with grant option;  9. tar xvf apache-hive-1.0.1-bin.tar.gz  10. cp -r apache-hive-1.0.1-bin /usr/local/hive  11. cp -r mariadb-java-client-1.3.5.jar /usr/local/hive/lib  12. vi /etc/profile  13. reboot  http://apache.tt.co.kr/hive/hive-1.2.2/apache-hive-1.2.2-bin.tar.gz  hive 파일 다운받을 ㄷ수 있는 곳  hive config  vi hive-site.xml  없어서 파일 새로 만든것  밑에는 이 파일에 덧붙일 내용  JAVA\_HOME=/usr/local/jdk1.8.0  CLASSPATH=$JAVA\_HOME/lib  HADOOP\_HOME=/usr/local/hadoop-1.2.1  HIVE\_HOME=/usr/local/hive  PATH=.:$JAVA\_HOME/bin:$HADOOP\_HOME/bin:$HIVE\_HOME/bin:$PATH  export JAVA\_HOME CLASSPATH HADOOP\_HOME HIVE\_HOME  export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE HISTCONTROL  <?xml version="1.0"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>hive.metastore.local</name>  <value>false</value>  <description>controls whether to connect to remove metastore server or  open a new metastore server in Hive Client JVM</description>  </property>  <property>  xml 파일 수정 후 하둡에 hive가 쓸 공간을 마련해준다.  1. cd /usr/local/hive/conf  2. hadoop dfs -mkdir /tmp  3. hadoop dfs -mkdir /tmp/hive  4. hadoop dfs -chmod 777 /tmp  5. hadoop dfs -mkdir /user/hive/warehouse  6. hadoop dfs -chmod 777 /user/hive/warehouse  start-all.sh로 하둡을 실행해 준 후 hive로 로그인  CREATE TABLE HDI(id INT, country STRING, hdi FLOAT, lifeex INT, mysch INT,  eysch INT, gni INT) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' STORED AS  TEXTFILE;  MariaDB에서 hive\_db 이용  select \* from TBLS;  테이블이 만들어진 것을 확인할 수 있음.  hdi 예제 데이터 파일 이용  엑셀 파일 형식 => CSV  mv hdi.csv hdi.txt  이름 바꿔줌  hive 로 로그인한 sql 에서  load data local inpath '/root/다운로드/hdi.txt' into table HDI;  (mysql은 구조만,, 실제 데이터는 하둡에 들어가는 것)  <name>javax.jdo.option.ConnectionURL</name>  <value>jdbc:mariadb://localhost:3306/hive\_db?  createDatabaseIfNotExist=true</value>  <description>JDBC connect string for a JDBC metastore</description>  </property>  <property>  <name>javax.jdo.option.ConnectionDriverName</name>  <value>org.mariadb.jdbc.Driver</value>  <description>Driver class name for a JDBC metastore</description>  </property>  <property>  <name>javax.jdo.option.ConnectionUserName</name>  <value>hive</value>  <description>username to use against metastore database</description>  </property>  <property>  <name>javax.jdo.option.ConnectionPassword</name>  <value>111111</value>  <description>password to use against metastore database</description>  </property>  </configuration>  select \* from hdi limit 5;(5번 미만의 데이터 출력)  맵리듀스 형식으로 출력되는 것.  describe hdi;  각 테이블 변수의 종류를 보여줌  +++ 예문  select sum(hdi) from hdi;  hive --service hiveserver2  hive가 대기하도록 리눅스 서버를 두고,  이클립스에서 java project, java class 생성 후  실행을 통해 데이터를 요청한다.  **hive 연습하기**  hadoophive  192.168.112.220  1. JDK 설치  2. Hadoop 설치  3. mariaDB 설치  4. hive 설치  hdi 데이터를 Java Application으로 분석 조회하기  00:50:56:2D:3C:5D  nayoung,2020-01-10,p001,10000  chocolate,2020-01-10,p002,20000  honye,2020-01-10,p003,30000  macarong,2020-01-10,p004,40000  cookie,2020-01-10,p005,50000  public static void main(String[] args) throws Exception {  Class.forName("org.apache.hive.jdbc.HiveDriver");  Connection conn = DriverManager.getConnection  ("jdbc:hive2://192.168.112.201:10000/default","root","111111");  Statement stmt = conn.createStatement();  ResultSet rs = stmt.executeQuery("SELECT sum(hdi) FROM HDI");  while(rs.next()) {  System.out.println(rs.getString(1));  }  conn.close();  System.out.println("Success....");  }  cereal,2020-01-10,p006,60000  apple,2020-01-10,p007,70000  CREATE TABLE LOGS(name STRING, date STRING, password STRING, value INT) ROW FORMAT  DELIMITED FIELDS TERMINATED BY ',' STORED AS TEXTFILE;  load data local inpath '/root/log.txt' into table LOGS;  SELECT year, month, deptime, arrtime, flightnum  FROM airline\_delay  WHERE delayYear = 2006;  2006년 월별 출발지연, 도착 지연의 평균을 구하시오.  SELECT year, month, AVG(arrdelay), AVG(depdelay)  FROM airline\_delay  WHERE delayYear = 2006  AND arrdelay > 0  GROUP BY Year, Month;  SELECT year, month, AVG(arrdelay), AVG(depdelay)  FROM airline\_delay  GROUP BY Year, Month  ORDER BY 1,2;  SELECT Year, Month, AVG(ArrDelay) AS avg\_arrive\_delay\_time, AVG(DepDelay) AS  avg\_departure\_delay\_time  FROM airline\_delay  WHERE delayYear = 2006  AND ArrDelay>0  GROUP BY Year, Month  ORDER BY Month;  1. 각 pc에 완전 분산모드의 Hadoop을 실행한다.  2. Namenode에 hive를 설치한다.  3. Hive 실행  4. airline\_delay 데이터를 입력한다.(2006~2008)  5. Java를 이용하여 데이터를 조회한다.\_\_    ㄴ | | | | | | |