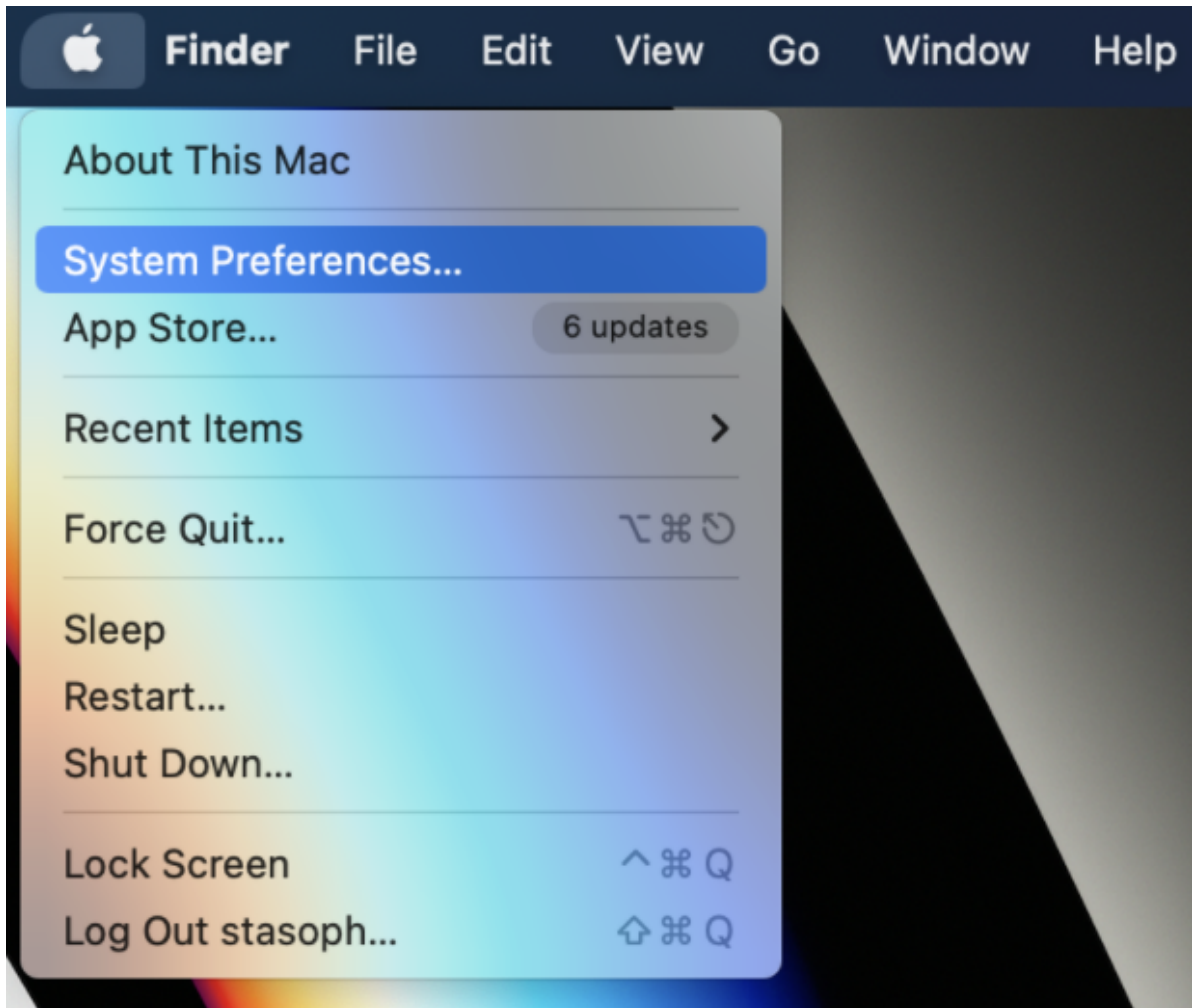


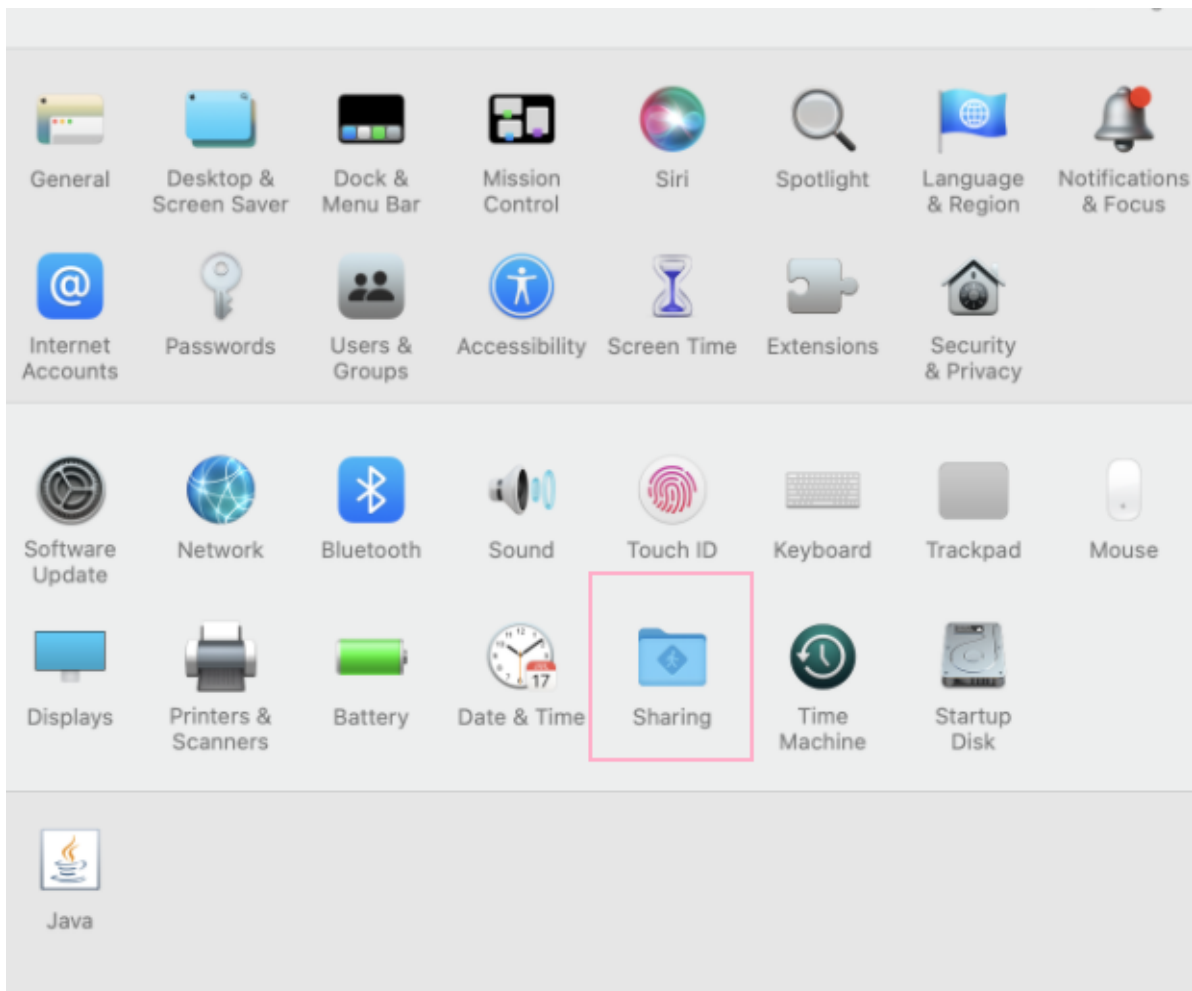
Install and configure hadoop on Mac

Configure ssh

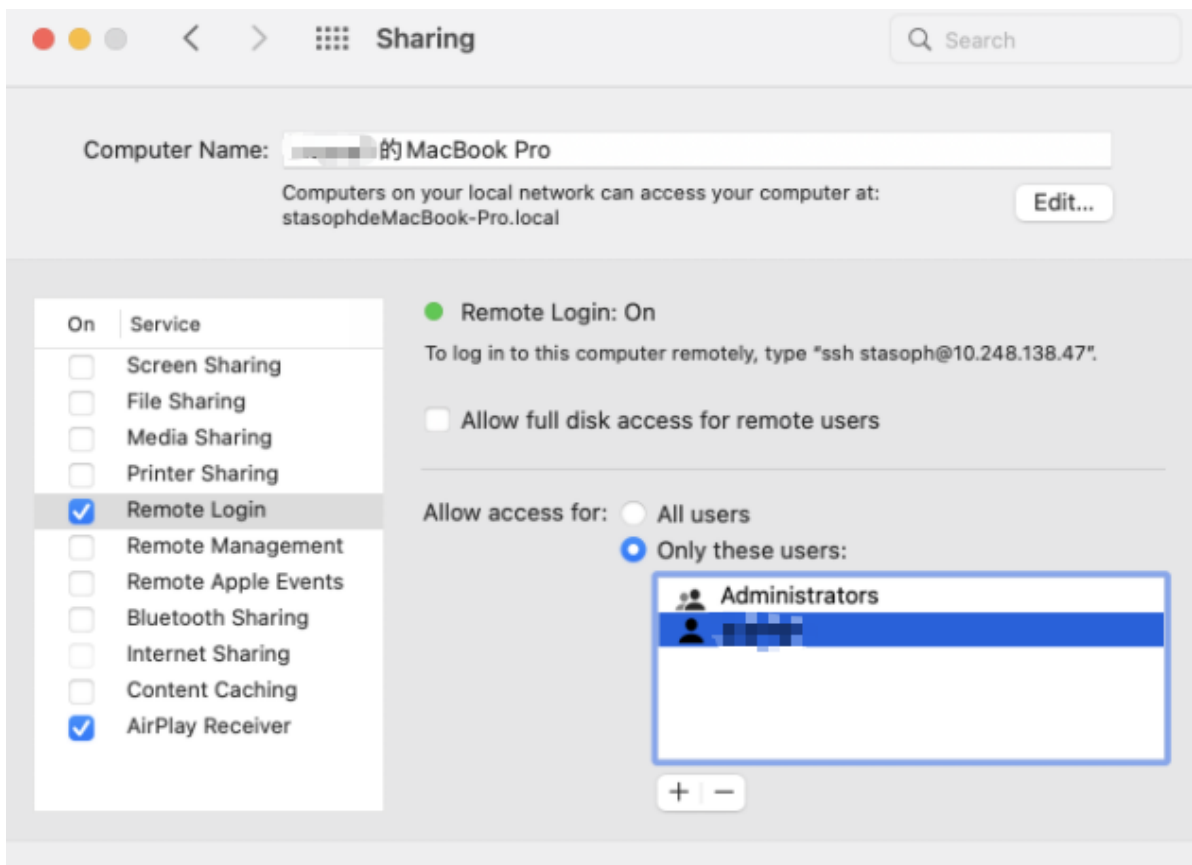
Enable remote login and add current user to allow list.

System Preferences -> Sharing -> Remote Login





Add current user to allow access list



Configure ssh

```
ssh-keygen -t rsa
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys

#verify ssh
ssh localhost
```

Install java

Confirm you have the correct version of java (**version 8 or 11**)

you can download java11 from the link below or install java 8 or java 11 on your own

jdk download link

```
https://aka.ms/download-jdk/microsoft-jdk-11.0.15-macos-x64.tar.gz
```

unzip and install jdk

```
#unzip jdk
tar xvf microsoft-jdk-11.0.15-macos-x64.tar.gz
#remove downloaded file
rm microsoft-jdk-11.0.15-macos-x64.tar.gz
#make directories
sudo mkdir -p /Library/Java/JavaVirtualMachines
#move jdk to JavaVirtualMachines
sudo mv jdk-11.0.15+10 /Library/Java/JavaVirtualMachines
```

Open the file zshrc (bashrc if you use linux)

```
open -a TextEdit ~/.zshrc
```

Then add the following lines to the **end** of this file (Note that you can find java home path by using `/usr/libexec/java_home` in terminal window):

```
export JAVA_HOME=/Library/Java/JavaVirtualMachines/jdk-11.0.15+10/Contents/Home
```

Save the file, and then run the following command to take these configurations into effect:

```
source ~/.zshrc
```

Download and configure Hadoop

1. Download Hadoop and Configure HADOOP_HOME

```
mkdir ~/workdir
cd ~/workdir

#Download the Hadoop package, you can also use wget or other tools to download
the file
curl -O https://d1cdn.apache.org/hadoop/common/hadoop-3.3.2/hadoop-3.3.2.tar.gz

#unpack the package
tar xvf hadoop-3.3.2.tar.gz
```

2. Configure HADOOP_HOME

Open the file zshrc (bashrc if you use linux)

```
open -a TextEdit ~/.zshrc
```

Then add the following lines to the **end** of this file (Note that you can find java home path by using `/usr/libexec/java_home` in terminal window):

```
export HADOOP_HOME=~/workdir/hadoop-3.3.2
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop
export PATH=$HADOOP_HOME/bin:$HADOOP_HOME/sbin:$PATH
export JAVA_HOME=/Library/Java/JavaVirtualMachines/jdk-11.0.15+10/Contents/Home
```

Save the file, and then run the following command to take these configurations into effect:

```
source ~/.zshrc
```

3. Configure HDFS

- **hadoop-env.sh**

open the hadoop environment file, hadoop-env.sh, using text edit (or use vim, emacs or nano):

```
open -a TextEdit $HADOOP_CONF_DIR/hadoop-env.sh
```

add the following to the **end** of this file

```
export JAVA_HOME=/Library/Java/JavaVirtualMachines/jdk-11.0.15+10/Contents/Home
```

- **core-site.xml**

open the HDFS core configuration file

```
open -a TextEdit $HADOOP_CONF_DIR/core-site.xml
```

Note that it is in xml format, and every configuration should be put in between `<configuration>` and `</configuration>`. You need to add the following lines:

Please change `username` to your own username (You can use `whoami` or `ls /Users` to check your username)

```
<property>
  <name>hadoop.tmp.dir</name>
  <value>/Users/username/workdir/hadoop-3.3.2/tmp</value>
</property>

<property>
  <name>fs.defaultFS</name>
  <value>hdfs://localhost:9000</value>
</property>
```

- **hdfs-site.xml**

Open hdfs-site.xml

```
open -a TextEdit $HADOOP_CONF_DIR/hdfs-site.xml
```

```
<property>
  <name>dfs.replication</name>
  <value>1</value>
</property>

<property>
  <name>dfs.namenode.name.dir</name>
  <value>file://${hadoop.tmp.dir}/dfs/name</value>
</property>

<property>
  <name>dfs.datanode.data.dir</name>
  <value>file://${hadoop.tmp.dir}/dfs/data</value>
</property>
```

- **mapred-site.xml**

Open mapred-site.xml

```
open -a TextEdit $HADOOP_CONF_DIR/mapred-site.xml
```

```
<property>
  <name>mapreduce.framework.name</name>
  <value>yarn</value>
</property>

<property>
  <name>yarn.app.mapreduce.am.env</name>
  <value>HADOOP_MAPRED_HOME=$HADOOP_MAPRED_HOME</value>
</property>
```

```
<property>
  <name>mapreduce.map.env</name>
  <value>HADOOP_MAPRED_HOME=$HADOOP_MAPRED_HOME</value>
</property>

<property>
  <name>mapreduce.reduce.env</name>
  <value>HADOOP_MAPRED_HOME=$HADOOP_MAPRED_HOME</value>
</property>
```

- **yarn-site.xml**

```
open -a TextEdit $HADOOP_CONF_DIR/yarn-site.xml
```

```
<property>
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle</value>
</property>
```

4. Start HDFS

```
# work in the Hadoop home folder
cd $HADOOP_HOME

#Format the NameNode
hdfs namenode -format

#Start HDFS
start-dfs.sh

#Use the command "jps" to see whether Hadoop has been started successfully. You
should see "SecondaryNameNode", "NameNode", "Jps" and "DataNode"
jps
```

You can browse the web interface for the information of NameNode and DataNode at: <http://localhost:9870>.

5. Start YARN

```
start-yarn.sh
```

Try jps again, you will see "NodeManager" and "ResourceManager", and these are the main daemons of YARN.

Browse the web interface (for supervision and debugging) for the ResourceManager at: <http://localhost:8088/>.

6. Using HDFS and running MapReduce in the pseudo-distributed mode

Please change `username` to your own username (You can use `whoami` or `ls /Users` to show your username)

```
hdfs dfs -mkdir -p /user/username
```

See the rest in file "**Lab 1**".