安装基础部分：

图形化问题：先按最小安装linux，再安装图形化工具，解决 登陆黑屏等问题

网络：

1. 最小化安装linux

磁盘先分配100G以内

选择中文支持

1. yum源配置

yum -y install make zlib zlib-devel gcc-c++ libtool openssl openssl-devel

yum install openssl-devel -y

yum install wget

wget <http://mirrors.163.com/.help/CentOS7-Base-163.repo>

cd /etc/yum.repos.d

mv \*.repo ../yum.repos.d\_backup/

cp /tmp/CentOS7-Base-163.repo ./

yum clean all

yum makecache

yum update

参考:

https://www.cnblogs.com/catlee/p/7417680.html

1. 关闭防火墙

firewall-cmd --state

systemctl stop firewalld.service

1. 基础编译环境

yum install ksh bad interpreter

1. 安装图形化

yum groupinstall "X Window System"

yum groupinstall "GNOME Desktop"

systemctl set-default graphical.target //设置成图形模式

#systemctl set-default multi-user.target //设置成命令模式

重启后，等待弹出设置图形化界面的配置界面--gnome-initial-setup，设置中文语言支持

参考：

<https://blog.csdn.net/duchao123duchao/article/details/72617768>

**Linux CentOS 7安装GNOME图形界面并设置默认启动方式**

1. 语言支持

*yum install kde-l10n-Chinese*

*yum reinstall glibc-common*

参考：

<https://blog.csdn.net/ysm_sd/article/details/51144975>

1. VNC登陆

yum install tigervnc-server –y

vncserver :1 -depth 24

1. 网络配置

[pcre-8.32.zip](https://ftp.pcre.org/pub/pcre/pcre-8.32.zip)

nginx

django

uwsig

1. 基础编译环境

yum install ksh bad interpreter

yum install -y gcc gcc-c++ make

yum install openssl-devel bzip2-devel expat-devel gdbm-devel readline-devel sqlite-devel gcc gcc-c++ openssl-devel

yum install libffi-devel

Python3.6

virtualenv

更换pip源

pandas

数据库连接

Python3.6:

yum install -y gcc gcc-c++

export CFLAGS="-I/usr/include"

export LDFLAGS="-L/usr/lib64"

export LD\_LIBRARY\_PATH=/usr/lib64

修改makefile（import ssl报错问题）

<https://www.cnblogs.com/mqxs/p/9103031.html>

#修改Setup文件,在解压文件夹下的Modules文件夹下的Setup.dist

vim Modules/Setup.dist

#修改结果如下：去掉最后4行注释，得到如下的结果

# Socket module helper for socket(2)

\_socket socketmodule.c timemodule.c

# Socket module helper for SSL support; you must comment out the other

# socket line above, and possibly edit the SSL variable:

SSL=/usr/local/ssl

\_ssl \_ssl.c \

-DUSE\_SSL -I$(SSL)/include -I$(SSL)/include/openssl \

-L$(SSL)/lib -lssl –lcrypto

make distclean

./configure --prefix=/usr/local/python3.6

make && make install

检查bz2、ssl是否加载

python

import bz2,ssl

运行虚拟环境：

root运行：

/home/pyetl/soft/virtualenv-16.0.0

python setup.py install

cd /user/local/

/usr/local/python3.6/bin/virtualenv --no-site-packages venv3.6

chown -R pyetl:pyetl venv3.6/

修改环境配置“vi .bash\_profile”：

. $HOME/venv3.6/bin/activate

更换pip3源：

mkdir ~/.pip

vi ~/.pip/pip.conf

[global]

index-url = http://mirrors.aliyun.com/pypi/simple/

[install]

trusted-host = mirrors.aliyun.com

参考：

<https://www.cnblogs.com/lpl521/p/6778048.html>

安装python依赖工具：

python -m pip install --upgrade pip -i <https://pypi.tuna.tsinghua.edu.cn/simple>

下载安装setuptools

pip3 install openpyxl xlwt xlrd xlutils

pip3 install Cython numpy pandas scipy

pip3 install matplotlib pyecharts pyecharts\_snapshot

pip3 install sqlalchemy

#pip3 install uwsgi flask django ibm\_db\_django

#pip install PyExecJS psycopg2 bs4 pychecktype

pip install psycopg2 bs4 pychecktype

#爬虫

pip install requests

# 加解密 参考打包入坑手册安装vs以及环境配置

#pyinstaller 加密打包的需要Crypto

pip install pyinstaller uncompyle rsa pycrypto

#用pycryptodome替代在site-packages下把crypto 改成Crypto

#pip3 install pycryptodome

**#**github：<https://github.com/dlitz/pycrypto>

**# 下载的安装不了情况下，使用pip install pycrypto 也行可以了**

压缩文件:

pip install rarfile filetype rarfile patool

安装<http://www.winrar.com.cn/>文件，并把安装路径加入到path中

安装7z、unrar第三方软件

上传ibm\_db目录下3个文件:

ibm\_db-2.0.8.tar.gz linuxx64\_odbc\_cli.tar.gz (windows: ntx64\_odbc\_cli.zip )

tar xvf ibm\_db-2.0.8.tar.gz

cd ibm\_db-2.0.8

修改setup.py：

注释掉170行~183行

tar xvf ../linuxx64\_odbc\_cli.tar.gz

python setup.py install

tar xvf ibm\_db\_sa-0.3.3.tar.gz

cd ibm\_db\_sa

python setup.py install

ibm\_db 加载错误:

<http://www-01.ibm.com/support/docview.wss?rs=4020&uid=swg21385217>

1. 语音

pip install pyttsx3 -i <https://pypi.tuna.tsinghua.edu.cn/simple>

**pip install pyaudio -i** [**https://pypi.tuna.tsinghua.edu.cn/simple**](https://pypi.tuna.tsinghua.edu.cn/simple)

python使用Speech\_Recognition实现普通话识别

<https://blog.csdn.net/qq_32643313/article/details/99936268>

**pip install pyowm -i** [**https://pypi.tuna.tsinghua.edu.cn/simple**](https://pypi.tuna.tsinghua.edu.cn/simple)

**pip install newsapi -i** [**https://pypi.tuna.tsinghua.edu.cn/simple**](https://pypi.tuna.tsinghua.edu.cn/simple)

1. 测试

**pip install pytest -i** [**https://pypi.tuna.tsinghua.edu.cn/simple**](https://pypi.tuna.tsinghua.edu.cn/simple)

1. 性能

pip install jupyter numba joblib

pip3 install ray（linux版本支持）

检查ray安装是否成功

import ray

pip install psutil

1. DB2

编码修改为LANG="en\_us.UTF-8"

/etc/locale.conf

https://blog.csdn.net/u013870094/article/details/51367673

Hostname

1. 图形化

之前是否需要安装QT5,待定

pip install pyqt5==5.13.0 PyQt5-tools sip

# 注意修改版本，安装后检查是否正确，验证语句: from Qt import \*

pip install PySide2-5.15.0（python3.6.3） 、 PySide2==5.12（python3.6.8）

qtpandas安装需要修改

pip install pytest\_cov future pytest\_qt easygui

-i <https://pypi.tuna.tsinghua.edu.cn/simple>

qtpandas\views\EditDialogs.py的17、176行

*if* pd.isna(defaultValue):

qtpandas\ models\DataFrameModel.py

替换 .ix[row,loc] iloc[row][loc]

#皮肤

<https://github.com/ColinDuquesnoy/QDarkStyleSheet>

<https://pypi.org/project/QDarkStyle/#files>

pip install **darkstyle**

**pip install qtmodern**

1. 视觉

pip install opencv-python imutils

1. 特例

PyFlow

<https://github.com/wonderworks-software/PyFlow?tdsourcetag=s_pcqq_aiomsg>

下载安装依赖:PySide2

<https://github.com/wonderworks-software/PyFlowMaya>

第一步：修改/etc/sysconfig/network文件

        #>vi /etc/sysconfig/network

        添加或修改:

                NETWORKING=yes

                HOSTNAME=slave3

第二步：修改/etc/hosts文件

        #>vi /etc/hosts

        修改 127.0.0.1这行中的 localhost.localdomain为 slave3

        修改 ::1这行中的localhost.localdomain 为slave3

第三步 ：修改/etc/hostname文件(此步不操作，怎么修改都没有用)

        删除文件中的所有文字，在第一行添加slave3

第四步：重启并验证

       #>reboot -f

       #> hostnamectl

初始问题:

黑屏，只有鼠标：

重安装gnoma

yum remove gnome-shell

**yum groupinstall** "GNOME Desktop" "Graphical Administration Tools"

  unlink /etc/systemd/system/default.target

  ln -sf /lib.systemd/system/runlevel5.target /etc/systemd/system/default.target

