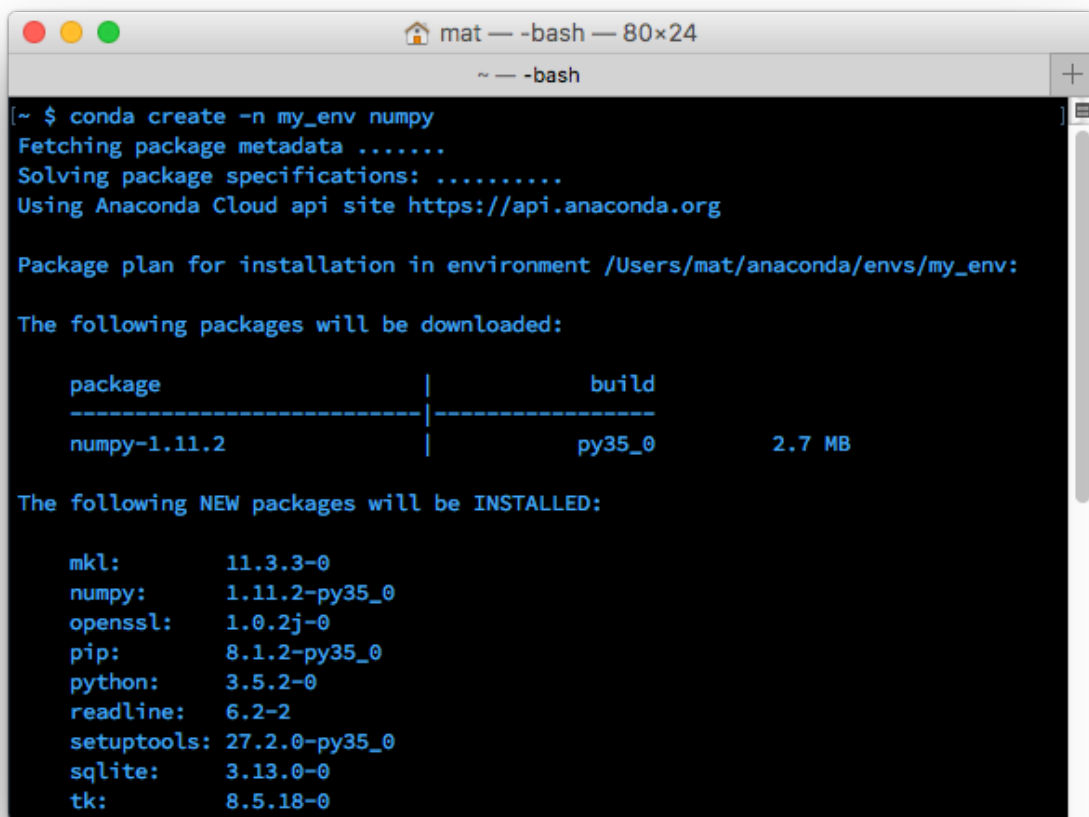




## 管理环境

如前所述，你可以使用 conda 创建环境以隔离项目。要创建环境，请在终端中使用 `conda create -n env_name list of packages`。在这里，`-n env_name` 设置环境的名称（`-n` 是指名称），而 `list of packages` 是要安装在环境中的包的列表。例如，要创建名为 `my_env` 的环境并在其中安装 `numpy`，请键入 `conda create -n my_env numpy`。



```
mat ~ -bash — 80x24
~ -bash
[~ $ conda create -n my_env numpy
Fetching package metadata .....
Solving package specifications: .....
Using Anaconda Cloud api site https://api.anaconda.org

Package plan for installation in environment /Users/mat/anaconda/envs/my_env:

The following packages will be downloaded:

package | build | size
-----|-----|-----
numpy-1.11.2 | py35_0 | 2.7 MB

The following NEW packages will be INSTALLED:

mkl:          11.3.3-0
numpy:        1.11.2-py35_0
openssl:      1.0.2j-0
pip:          8.1.2-py35_0
python:       3.5.2-0
readline:     6.2-2
setuptools:   27.2.0-py35_0
sqlite:       3.13.0-0
tk:           8.5.18-0
```

创建环境时，可以指定要安装在环境中的 Python 版本。这在你同时使用 Python 2.x 和 Python 3.x 中的代码时很有用。要创建具有特定 Python 版本的环境，请键入类似于 `conda create -n py3 python=3` 或 `conda create -n py2 python=2` 的命令。实际上，我在我的个人计算机上创建了这两个环境。我将它们用作与任何特定项目均无关的通用环境，以处理普通的工作（可轻松使用每个 Python 版本）。这些命令将分别安装 Python 3 和 Python 2 的最新版本。要安装特定版本（例如 Python 3.3），请使用 `conda create -n py python=3.3`。



上，请使用 `activate my_env`。

```
mat — -bash — 90x24
~ — -bash

[~ $ source activate my_env
(my_env) ~ $ conda list
# packages in environment at /Users/mat/anaconda/envs/my_env:
#
mkl                  11.3.3                0
numpy                1.11.2               py35_0
openssl              1.0.2j                0
pip                  8.1.2               py35_0
python               3.5.2                0
readline             6.2                  2
setuptools           27.2.0              py35_0
sqlite               3.13.0                0
tk                   8.5.18                0
wheel                0.29.0              py35_0
xz                   5.2.2                0
zlib                 1.2.8                 3
(my_env) ~ $
```

进入环境后，你会在终端提示符中看到环境名称，它类似于 `(my_env) ~ $`。环境中只安装了几个默认的包，以及你在创建它时安装的包。你可以使用 `conda list` 检查这一点。在环境中安装包的命令与前面一样：`conda install package_name`。不过，这次你安装的特定包仅在你进入环境后才可用。要离开环境，请键入 `source deactivate`（在 OSX/Linux 上）。在 Windows 上，请使用 `deactivate`。

### 练习题

你将使用哪个命令来创建名为 `data` 且安装了 Python 3.5、numpy 和 pandas 的环境？

- ☐ `conda env create -n data python=3.5 numpy pandas`



## 管理环境

☐ conda create -n data python=3.5

conda create -n data python=3.5 numpy pandas

提交

下一项