在容器中管理数据主要有两种方式：

1. 数据卷（Data volumes）
2. 数据卷容器（Data volume containers）

# 数据卷

1. 数据卷是一个可供一个或多个容器使用的特殊目录，它绕过UFS（Unix文件系统），可以提供很多有用的特性：
   1. 数据卷可以在容器之间共享和重用
   2. 对数据卷的修改会立马生效
   3. 对数据卷的更新，不会影响镜像
   4. 卷会一直存在，直到没有容器使用
2. \*数据卷的使用，类似于Linux下对目录或文件进行mount

**创建一个数据卷**

1. 在用docker run命令时，使用 **-v 参数**来创建一个数据卷并挂载到容器里。
2. 下面创建一个Web容器，并加载一个数据卷到容器的/webapp目录：
   1. sudo docker run -d -p --name web -v /webapp training/webapp python app.py
3. **\*注，**也可以在Dockerfile中使用VOLUME添加一个或者多个新的卷到由该镜像创建的任意容器中

**挂载一个主机目录作为数据卷**

1. 使用 -v 参数也可以指定挂在一个本地主机的目录到容器中去
   1. sudo docker run -dp --name web -v /src/webapp:/opt/webapp training/webapp python app.py
   2. 该命令加载主机的/src/webapp目录到容器的/opt/webapp目录。
   3. 这个功能在测试时十分方便，比如用户可以放置一些程序到本地目录中，来查看容器是否正常工作
   4. 本地目录的路径必须是绝对路径，如果目录不存在Docker会自动创建
2. **\*注，**Dockerfile不支持这种用法，因为Dockerfile是为了移植和分享用的，但不同操作系统的路径格式不一致，所有目前无法支持。
3. Docker挂载数据卷的默认权限是读写，用户也可以通过:ro指定为只读
   1. sudo docker run -dp --name web -v /src/webapp:/opt/webapp training/webapp:ro python app.py

**挂载一个本地主机文件作为数据卷**

1. **-v 参数**可以从主机挂载单个文件到容器中：
   1. sudo docker run --rm -it -v ~/.bash\_history:/.bash\_history ubuntu /bin/bash
   2. 这样就可以记录下再容器输入过的命令了
2. **\*注**，如果直接挂载一个文件，很多文件编辑工具，包括vi或者sed --in-place，可能会造成文件的iNode改变，从Docker1.1.0起会导致报错。所以最简单的还是直接挂载到文件的父目录

# 数据卷容器

1. 如果有一些持续更新的数据需要在容器之间共享，最好创建数据卷容器
2. 数据卷容器，其实就是一个正常的容器，**专门用来提供数据卷供其他容器挂载的**。
3. 首先，创建一个名为bddata的数据卷容器：
   1. sudo docker run -d -v /dbdata --name dbdata training/postgres echo Data-only container for postgres
4. 然后，在其他容器中使用 **--volumes-from** 来挂载dbdata容器中的数据卷
   1. sudo docker run -d --volumes-from dbdata --name db1 training/postgres
   2. sudo docker run -d --volumes-from dbdata --name db2 training/postgres
5. 还可以使用多个 --volumes-from 参数从多个容器挂载多个数据卷。也可以从其他已经挂载了数据卷的容器来挂载数据卷：
   1. sudo docker run -d --name db3 -volumes-from db1 training/postgres
6. **\*注，使用 --volumes-from 参数所挂载数据卷的容器自己并不需要保持在运行状态**
7. 如果删除了挂载的容器，数据卷并不会被自动删除。如果要删除一个数据卷，必须在删除最后一个还挂载着它的容器时使用**docker rm -v命令**来制定同时删除关岭的容器。

# 利用数据卷容器来备份、恢复、迁移数据卷

**备份**

1. 首先使用 **--volumes-from 参数**创建一个加载dbdata容器卷的容器，并从本地主机挂载当前到容器的/backup目录：
   1. sudo docker run --volumes-from dbdata -v $(pwd):/backup ubuntu tar cvf /backup/backup.tar /dbdata
2. 容器启动后，使用了tar命令来将dbdata卷备份为本地的/backup/backup.tar

**恢复**

1. 如果要恢复数据到一二容器，首先创建一个带有数据卷的容器dbdata2
   1. sudo docker run -v /dbdata --name dbdata2 ubuntu /bin/bash
2. 然后创建另一个容器，挂载到dbdata2，并使用untar解压备份文件到挂载的容器卷中：
   1. sudo docker run --volumes-from dbdata2 -v $(pwd):/backup busybox tar xvf /backup/backup.tar