# django orm关联查询  
# 多方Hero   一方Book  
# 1多找一， 多方对象.关系字段    exp: h1.book  
# 2一找多， 一方对象.小写多方类名\_set.all()   exp:  b1.hero\_set.all()

1工具：Git(给特)

目的：公司团队开发项目时的版本控制软件

目前流行版本控制系统（VCS version control system）有：**GIT**  SVN

分布式VCS（GIT）：没有明确服务器概念，每一个客户端都具有完整的代码库

集中式VCS （SVN）：需要指定服务器， 服务器挂掉，版本控制系统失败，所有客户端都需要通过服务器获取代码

GIT使用：

必知概念：

工作区：项目代码或者资料所在目录，即磁盘上的工作目录

暂存区：临时存储目录,用于存储项目更改内容

本地仓库：实际存储位置，和工作区，暂存区都位于开发机器上

远程仓库：位于另外一台电脑上（充当git服务器）只是作为中转代码

Git branch 查看所有的分支  
Git branch dev1 创建分支  
Git checkout dev1 切换分支  
Git checkout -b dev1 创建并且切换分支  
Git merge dev1 先切换到主分支  然后合并dev1到主分支  
Git branch-d dev1 删除分支

一:本地仓库使用

1. 建库（repository）

新建空目录，使用git init建库，会生成.git隐藏文件

2，添加到暂存区

Git add 文件名

注意添加到暂存区之后如果想要从暂存区移除有两种情况

该文件没有提交过：可以使用 git rm –cached 文件名

如果该文件之前commit过（此时状态为modify） 则需要使用git checkout – 文件名

3，暂存区提交到本地仓库

Git commit -m ‘备注’

4，撤销修改内容

git checkout – 文件名

5，撤销暂存区到工作区

Git reset HEAD 文件名

4，查看工作区，暂存区状态

Git status

红色 代表有修改但是没有提交到暂存区

绿色 暂存区有内容没有提交到本地仓库

5 查看仓库日志

Git log

6 查看所有版本

Git reflog

7 回档到指定版本

Git reset version

8 使用HEAD 可以回退到上一个/ 上几个版本

Git reset –hard HEAD^

Git reset –hard HEAD^^^

Git reset –hard HEAD~n

Git：

系统用户:位于C盘用户Administrator目录下方.gitconfg

Git config –global user.name zzy

Git config –global user.email [496575233@qq.com](mailto:496575233@qq.com)

项目用户:位于项目文件夹下方.git目录 config文件

Git config user.name zzy

Git config user.email [18137128152@163.com](mailto:18137128152@163.com)

分支 master 主分支

Git branch dev1 创建分支

Git checkout dev1 切换分支

Git checkout -b dev1 创建并且切换分支

Git branch-d dev1 删除分支

Git push 命令

完整写法可以为git push origin master 即将本地master分支推动到远程master分支

也可以将本地分支dev1 推送到远程分支 使用 git push origin dev1

删除远程dev分支需要使用git push origin –delete dev1

由项目经理在github新建仓库，注意不要勾选创建readme.md

git remote add origin https://github.com/zzy0371/vueproject.git

git push -u origin master