## IDEA

汇总：idea的使用方法

一：快捷键

ctrl+p 罗列出这个方法的所以形参

Alt+enter 快速修复问题（给出问题的几个解决的方案）:

Ctrl+shift+u: 将选中的字符进行大小写切换（全部改成大写或者小写）

怎么回到初始界面：file->close project

Alt+F7:察看方法在什么地方被调用

二：项目结构

Idea中退回到初始界面，file->close project

.idea文件夹：存放项目的配置信息，包括数据源，编码，历史记录，版本控制信息等。删除之后，从新打开项目可以重新生成在使用GitHub上传的时候，并不需要将其上传，要放在忽略文件当中。

.iml文件夹：项目标识文件（表示设置项目的module）。项目会有方括号，就是sell.iml的名称和项目文件夹的名称不一样。通过将sell.iml文件名称改成和项目文件夹一样，就不会有方括号。如果这杯文件被误删除或者从GitHub上下载没有这个文件，就不会显示为一个项目，

删除.idea和.iml文件夹后，将两个文件同时删除，重启项目会自动生成。如果骨灰自动生成，子再使用项目结构目录。

.mvn文件夹：存放maven-wrapper相关jar包和相关配置文件.properties

mvnw(maven wrapper)的缩写，当发现当前的maven版本和.mvn文件夹中设置的版本不一致的时候，下载期望的版本，然后用期望的版本来执行（.mvn mvnw mvnw.cmd）

mvnw.cmd：执行mvnw命令的入口

项目没有被加入或者识别成maven项目：点击pom.xml文件，add as maven project。

将文件隐藏：file->settring->edit->file types->ignored files and folders

Spring中的工程分为project，module，libraries，facets，aritfacts。

三．Idea中git的使用

下载git，将git安装到本地电脑上

再idea中关联github,

将git关联到git或者是github，在进行克隆的时候，会被要求输入GitHub的账号密码（会有一个凭证冲突，就是换一个GitHub账号push的时候，这时候要在控制面板里面删除凭证）

先建好忽略文件，将本地新建一个git仓库，然后将本地文件追踪到仓库中去。如果将本地仓库删除了，就将git中的记录删除。

## Web项目中target

要把生成的target项目添加的out中，勾选include in

在项目结构中，artifacts添加lib目录，把相关的依赖包加上

报错：artifact中，没有添加相关依赖，新建文件夹。添加jar。为了防止重复，先将out，target文件夹删除。

Libraty中是相关jar包，即maven中导入的，真正有用的是class，source和Java Doc是给开发人员看的，爆红也没事，解决就是跟新maven

将一个项目标记为web，从0到1，添加框架支持，拷贝的就是设置facet和artifacts.project和modular也是一样

## Postman

1. **错误的url**
2. **错误的协议：http还是https**
3. **本地防火墙没关**
4. 代理服务器问题
5. 访问某些服务器需要客户端有证书
6. **使用https连接的时候，要进行安全验证（需要ssl安全证书）：要么在客户端关闭安全验证，要么发送安全证书**
7. **排错：postman中设置超时响应时间**

## Git

使用ssh连接，要有公钥：公钥生成方法，使用命令 ssh-keygen

git config –list 察看全局变量

git config --global user.name “zhangkuan”

git config --global user.eamil [3360094030@qq.com](mailto:3360094030@qq.com)

git init 在文件夹下初始化一个仓库

git clone 仓库连接 将远程仓库克隆下载

git status 检查当前的状态

git add 逆向操作：git reset HEAD [file]

git commit 逆向操作：git reset –-hard HEAD^ 或者git reset –hard HEAD~1

rm hello.text (将本地的删除，但是仓库中还有)

1. 要回来 git checkout -- hello.txt
2. 彻底删除 git rm hello.txt

git reset HEAD file

git reset -- hard HEAD^

git checkout -- hello.txt

git rm hello.txt