base：里面有个基类 （BasePrepare.java）,这个类的主要作用是启动和关闭浏览器的作用，启动浏览器使用了TetsNG的@BeforeClass ，在这个注释下写了driver的初始化以及打开浏览器等操作；关闭浏览器使用了TestNG的@AfterClass，在这个注释下写了driver退出的操作，测试用例中则是以@Test注释为主体的，在@Test注释下，书写了完整的测试步骤。测试用例继承这个基类，这样在运行测试类的时候，它先会去基类中找@BeforeClass下的代码去执行，然后再去执行@Test的内容，最后在执行@AfterClass的内容。

pages: 页面元素类，每一个java类，都是一个页面，存放的都是对应页面的所有元素。pages包下还有一个子包：pageshelper,这个包主要存放的是对应页面的帮助类，这是什么意思？简单的说：在这个页面上做了什么操作都写在这个pageshelper中，比如我登录页面的方法，都写在登录帮助类中。

plugins：主要存放的是arrow插件以及第三方插件。

testcases：顾名思义就是存放测试用例的地方，在这个包下，还会有很多子包，子包的个数根据你测试的系统的模块来划分，比如你有登录模块，首页模块等，那么我们起子包的名字就应该写成login、home。

util：这个包主要是封装了各种工具类，包括读取excel，Selenium api封装类，读取数据库类，读取属性文件类和选择浏览器类等。

代码目录以及介绍完毕，下面介绍下剩下的文件夹都是干什么的。

bin目录：项目编译生成class文件存放目录，这个目录不用去管，Eclipse会自动控制。

config目录：存储框架类所需的所有属性文件，比如arrow的配置文件以及自定义的一些属性文件。

doc目录：可以存放一些项目文档，比如你自己框架的说明文档，编码规范等。

data目录：各模块需要用到一些测试数据，以Excel文件形式存放该目录下。每个模块对应一个Excel文件，每个sheet对应的是一个测试用例的测试用例。

res目录：主要存放的是driver，比如chromediver和iedriver等。当然也会存放一些图片，第三方脚本（主要是为了辅助测试）等。

result目录：存储测试框架运行测试用例生成的报告（包含log，截图等）

src和target目录：可以不用管，由maven控制即可，整个框架，没有用到此目录，大家可以忽略。

接下来还剩几个xml文件，我们也来一一介绍一下：

pom.xml：maven的配置文件，项目核心配置，用于构建项目、自动下载项目依赖以及后续的和testng、jenkins配合持续集成等

runAll.xml：这是testng的配置文件，用于存放部分测试数据以及测试的平台，浏览器的配置、加入第三方插件监听（arrow插件）、设置用例执行策略（多线程还是单线程，顺序执行还是无序执行以及是否依赖执行等）以及设置要执行的用例。之所以改成runAll顾名思义就是说执行整个web项目的所有模块的用例的测试

runSingle.xml：和runAll.xml一样，不多解释，不一样的地方就是用它来做单个用例的调试改错，只涉及到单个的类（用例），所以调试改错专用。比如我用runAll.xml跑完了所有的用例，但是发现个别用例失败，此时先去分析log，如果是代码问题就去调试代码，调试完毕之后就去用runSingle.xml去跑一边此用例。

default.xml: 只存储了测试参数（parameter）以及监听器（arrow），没有任何测试用例放在里面，这个xml文件是为了本框架的一个默认testng配置文件，主要是为了，直接在测试用例中右键运行测试用例，如果不配置一个默认的文件，直接在测试用例类中直接运行会跳过用例。因为xml文件中存储了一些必须的参数，比如测试浏览器是什么，测试平台是什么等等。框架搭建过程中会讲到如何设置default.xml文件。

runSmoke.xml：这个主要执行冒烟测试用的，冒烟测试我们会先一些基本和核心功能作为测试重点，迅速跑一遍这些功能点，看看这个软件版本能不能released。