**[selenium处理页面等待时间的几种方式](http://blog.csdn.net/russ44/article/details/52032412)**

最近在项目过程中使用selenium 判断元素是否存在的时候 遇到一个很坑爹的问题，当某些元素不存在时，页面可能出现响应很久的情况，以下是本人总结的webdriver处理页面等待时间的几种方式：

1.显式等待

显式等待,就是明确的要等到某个元素的出现或者是某个元素的可点击等条件,等不到,就一直等,除非在规定的时间之内都没找到,那么就跳出Exception.

/\*\*
\* 设置元素等待时间
\*
\* @param driver
\* @param by
\* @param timeOut  //等待时间，以秒为单位
\*
\*/
public static void waitForLoad(WebDriver driver, final By locator, int timeOut) {
WebDriverWait wait = new WebDriverWait(driver, timeOut);  // timeOut为等待时间，单位秒
wait.until(new ExpectedCondition<Boolean>() {
public Boolean apply(WebDriver d) {
boolean loadcomplete = d.findElement(locator).isDisplayed();
return loadcomplete;
}
});
}

2.隐式等待

driver.manage().timeouts().implicitlyWait(second, TimeUnit.SECONDS); //second为等待时间，单位秒

设置driver内findElement等方法的超时时间，该设置为全局设置，使用起来比较方便。

隐式等待, 此处的隐式等待是针对Driver 每次执行命令的 最长执行时间也可以理解为超时时间，  一些人对此处有误解，认为是让Driver等一段时间，  确实某些时候能让Driver等一段时间， 但是影响是全局的，每次Driver执行 找不到元素都会等待此处设置的时间，  假设某处将此值设置的太长，必须在执行完成之后还原回来，否则判断一个元素是否存在的时候，就会遇到很坑爹的问题。   改进后的方法如下：

WebDriver会进行一个隐式等待,但参数只有时间,这就导致我需要什么元素出现,我不一定能等到它,

  private boolean isElementPresent(By by) {
    try {
      driver.manage().timeouts().implicitlyWait(1, TimeUnit.SECONDS);
      driver.findElement(by);
      return true;
    } catch (NoSuchElementException e) {
      return false;
    }
  }

得不到某个元素,我们就延迟一下...

3.线程休眠

Thread.sleep(3000); //线程停止3秒

使当前线程进入等待，Thread.sleep();这种等待属于死等，会强制等待一段时间，很容易让线程挂掉，使程序抛异常，不建议使用，影响自动化执行效率。所以我们要慎用此方法，适用于无法识别控件的情况下。

个人而言，建议方法2和3结合使用效果较好！