**1，定制明确等待**

    @Test

**public** **void** testExplicitWait() {

        WebDriver driver = **new** FirefoxDriver();

        driver.get("http://www.baidu.com/");

**try** {

            WebElement page4button = driver.findElement(By.*linkText*("/to/be/done"));

            page4button.click();

            WebElement message = (**new** WebDriverWait(driver, 5))  //创建WebDriverWait对象,等待5秒

                    .until(**new** ExpectedCondition<WebElement>() {  //调用until方法，并传入ExpectedCondition<T>参数，返回类型T

                        @Override

**public** WebElement apply(WebDriver d) {

**return** d.findElement(By.*id*("/to/be/done"));

                        }

                    });

*assertTrue*(message.getText().contains("Nunc nibh tortor"));

        } **catch** (NoSuchElementException e) {

*fail*("Element not found!!");

            e.printStackTrace();

        } **finally** {

            driver.close();

        }

    }

变型-1：返回Boolean类型

            Boolean message = (**new** WebDriverWait(driver, 5))

                    .until(**new** ExpectedCondition<Boolean>() {    //改成Boolean

                        @Override

**public** Boolean apply(WebDriver d) {

**return** d.findElement(By.*id*("/to/be/done")).isDisplayed();

                        }

                    });

变型-2：当jquery正在读取或处理比较大的json文件的时候，怎样判断它的读取活动已经完成

            Boolean message = (**new** WebDriverWait(driver, 5))

                    .until(**new** ExpectedCondition<Boolean>() {

                        @Override

**public** Boolean apply(WebDriver d) {

                            JavascriptExecutor js = (JavascriptExecutor)d;

**return** (Boolean)js.executeScript("return jQuery.active == 0");  //Jquery.active=0表示ajax活动已经完成

                        }

                    });

除了jquery之外，其他的js框架也有对应的状态可用，如：

Prototype: “Ajax.activeRequestCount”

Dojo: “dojo.io.XMLHTTPTransport.inFlight.length”