对于初学者来说，对Token和Session的使用难免会限于困境，开发过程中知道有这个东西，但却不知道为什么要用他？更不知道其原理，今天我就带大家一起分析分析这东西。

## 我们先解释一下他的含义：

1、Token的引入：Token是在客户端频繁向服务端请求数据，服务端频繁的去数据库查询用户名和密码并进行对比，判断用户名和密码正确与否，并作出相应提示，在这样的背景下，Token便应运而生。

2、Token的定义：Token是服务端生成的一串字符串，以作客户端进行请求的一个令牌，当第一次登录后，服务器生成一个Token便将此Token返回给客户端，以后客户端只需带上这个Token前来请求数据即可，无需再次带上用户名和密码。

3、使用Token的目的：Token的目的是为了减轻服务器的压力，减少频繁的查询数据库，使服务器更加健壮。

了解了Token的意义后，我们就更明确的知道为什么要用他了。

## 如何使用Token？

这是本文的重点，在这里我就介绍常用的两种方式。

1、用设备号/设备mac地址作为Token（推荐）

客户端：客户端在登录的时候获取设备的设备号/mac地址，并将其作为参数传递到服务端。

服务端：服务端接收到该参数后，便用一个变量来接收同时将其作为Token保存在数据库，并将该Token设置到session中，客户端每次请求的时候都要统一拦截，并将客户端传递的token和服务器端session中的token进行对比，如果相同则放行，不同则拒绝。

分析：此刻客户端和服务器端就统一了一个唯一的标识Token，而且保证了每一个设备拥有了一个唯一的会话。该方法的缺点是客户端需要带设备号/mac地址作为参数传递，而且服务器端还需要保存；优点是客户端不需重新登录，只要登录一次以后一直可以使用，至于超时的问题是有服务器这边来处理，如何处理？若服务器的Token超时后，服务器只需将客户端传递的Token向数据库中查询，同时并赋值给变量Token，如此，Token的超时又重新计时。

2、用session值作为Token

客户端：客户端只需携带用户名和密码登陆即可。

客户端：客户端接收到用户名和密码后并判断，如果正确了就将本地获取sessionID作为Token返回给客户端，客户端以后只需带上请求数据即可。

分析：这种方式使用的好处是方便，不用存储数据，但是缺点就是当session过期后，客户端必须重新登录才能进行访问数据。

## 使用过程中出现的问题以及解决方案？

刚才我们轻松介绍了Token的两种使用方式，但是在使用过程中我们还出现各种问题，Token第一种方法中我们隐藏了一个在网络不好或者并发请求时会导致多次重复提交数据的问题。

该问题的解决方案：将session和Token套用，如此便可解决，如何套用呢？请看这段解释：



wKioL1QX85nCkJ5qAABWcdNyC0g731.png

这就是解决重复提交的方案。

总结：以上是个人对开发中使用Token和session的一点总结，如有叙述不当之处请指正，我将及时改正并感谢，我知道还有更多更好的使用方式，我在这里只是抛砖引玉，希望大家将您的使用方式提出来，我们一起讨论，学习，一起进步，同时也为像我一样对这方面理解薄弱的朋友提供点帮助，谢谢。