Laya的Tween.to或.from是默认将创建出来的tween结束后放进对象池中，

当某个对象持有Tween对象时，在tween结束时必须要释放掉改该对象，否则当别的缓动从池中取出改对象时，如果之前的类依然持有这个对象，就会出现双方同时持有改对象的情况，所以结束时必须要动画结束时必须要清除对改对象的引用

例子：

Private tweenObj ;

Function clearTween(){

If(this.tweenObj){

Laya.Tween.clear(this.tweenObj);

this.tweenObj = null;

}

}

① this.tweenObj = Laya.Tween.to(target,prop, time,..); //没有监听结束

//上面的tween执行完后，对象会被放进池中

Let tweenObj 2 = Laya.Tween.to(target,prop, time,..); //从池中拿到了上面的对象

这时再调用this.clearTween();就会将tweenObj 2给停止了。

解决：

在①中监听缓动结束监听，结束就将this.tweenObj = null，将引用去掉