****Atitit bug管理法 面对 Bug 的态度****

只要你持续地写代码，就一定会持续的产生 Bug，所以第一个事情是要摆正对 Bug 的心态。我遇到过两个极端。

第一个极端。

遇到过一个leader，对系统质量相当重视，对我们写的代码要求很高，每次设计并写完一个新的系统，他喜欢跟你算这次的系统上线，产生了多少次故障，这半年时间产生了多少个 Bug，每个 Bug 的影响范围如何。

我们一堆人被搞得特别累，战战兢兢，到后面，大家都比较排斥去做优化，去重构代码，只求无过，不求有功了。所以，我觉得这种方式不好，对 Bug 带着一种比较包容的态度去看待，可以减少不少的心理负担。

第二个极端。

后来去了一个新的团队。新的团队很重视业务和工程迭代的速度，所以对代码质量和 Bug 的容忍度很高。如果是一个新上线的业务，是默许 Bug 存在的。

这种对质量过于松散的要求，在后面也带来不好的影响。大家对 Bug 太免疫了，以至于出现 Bug 和 故障的时候，大家都不够紧张。系统质量有一段时间出现比较大的问题，还因此被部门经理特训了一番，后面通过各种措施，才慢慢提高了整体的系统质量。

上面两种极端都不可取，应该很重视 Bug，尽量避免 Bug，但也不应该唯 Bug 多少论业绩。

具体到 Bug 的查找上，我说说我的一些经验。