

## 01. 微实战策略

学习要点：

1. 微实战策略
2. 实战内容

本节课我们来开始进入 ThinkPHP6.x 微实战 Demo 模块的演练课程。

### 一. 微实战策略

1. 什么微实战？顾名思义：小的，Demo 级别，独立模块的开发学习；
2. 下面分几点说明 ThinkPHP 微实战的特性：
  - (1) .尽可能独立模块，避免模块之间的关联导致复杂度提升；
  - (2) .尽可能不引入其它技术(如 UI)，避免专注度偏移和复杂度提升；
  - (3) .尽可能不使用前端(如 JS 框架/库)，同上；
  - (4) .尽可能不考虑各种规范、思想等，避免深度提升从而导致难度提升；
  - (5) .尽可能缩短课程内容，让学员能够快速完结掉，从而能提升信心；
  - (6) .综上所述：微实战适合刚学完基础，还没有做过项目的学员；
3. 说完微实战的特性，我们再谈一下课程的意义以及后续学习的补足；
  - (1) .我个人对于课程的意义理解：高完成率/低劝退率，一切为此服务；
  - (2) .真实项目，模块之间一定是千丝万缕的联系，后期还需要综合实战训练；
  - (3) .框架本身底层已经强行让你遵循很多这些规范和思想了，原生时需要了解；
4. 也就是说微实战是为后续综合实战做准备的，有了这个跳板，劝退率大大降低；
5. 说到劝退率，分析一下对于刚开始实战的同学有哪些因素导致劝退：
  - (1) .老司机是不需要微实战的，因为技术技能点多，思维盲区少；
  - (2) .对于没有项目经验的，技能少，盲区多，初期最好不要引入各种第三方技术；
  - (3) .在项目中每增加一个额外技术(比如 UI)，劝退率+10%；
  - (4) .在项目中每增加一个技术思想，劝退率+10%；
  - (5) .实际上，光 UI 和 JS 前端这两项，就可能占用项目 60~70%的代码量；
  - (6) .就算引入 UI，不讲解，只讲解后端，可是庞大前端代码会让你混乱；
  - (7) .所以，把这些控制在 10%以内，把专注度全部放在实现模块功能上；
6. 可是，如果不引入 UI，界面会变得丑陋且没有条理性，又不利于讲解；
7. 而如果用原生 HTML5/CSS 来构建界面，代码量往往不比引入 UI 少；
8. 而对于初学实战的同学，就好像没谈过恋爱的男生，是一个标准的颜值动物；
9. 他认为哪个好看，哪个就好。只有被爱情伤过、被社会毒打过，才知道真相；
10. 所以，为了迎合初学实战的同学，我们引入 Bootstrap(UI 库)；
11. 但是，引入仅仅只用它的表格、表单和按钮样式，其它基本不用；
12. 也就是说：虽然引入了 Bootstrap，却只用它不到 5%的功能，并不需要你专门学；
13. 这样，就非常好的兼顾学员的颜值要求，又不影响 ThinkPHP 实战的专注度；

## 二. 实战内容

1. 实战是对于学完基础，还没有接触过实战开发的同学制定的；
2. 比如常用模块：增删改查、用户认证、权限管理等等；
3. 由于不同项目，开发方式不同，我们这里会兼顾一部分：
  - (1) .数据库操作上，至少一个只使用 DB 方式的，剩下可以用 Model；
  - (2) .有些同学这辈子都不会用模板引擎，微实战会安排一个模块原生 PHP；
  - (3) .对于模态框、验证和 Ajax 异步通信，既然已经全面避免了，就不用；
  - (4) .能不能兼容两年多前的 5.1 的版本？答：思路和方法一样，代码 90%一样；