# 2023MSC社团精品课程圆满结束~~计算机实用软件教学讲座活动回顾与总结

## 活动回顾

为提高同学们的工作学习效率，浙江大学学生视窗俱乐部（原微软学生俱乐部）于10月14日举行了以计算机实用软件为主题的精品课程。课程分别介绍了Markdowm、VS Code、Git三种计算机实用工具的配置方式和使用方法，收获了同学们的一致好评！

讲座介绍了Markdown语言编辑器Typora的基本操作，接着基于Typora编辑器介绍Markdown语言的基础语法，最后介绍了Markdown语言的进阶技巧如HTML元素、数学公式表示、格式转换。

讲座介绍了VS Code的安装与配置、推荐的插件，并以C代码为例剖析了.vsode文件，分享了在单文件编译与多文件编译时文件的配置方法。

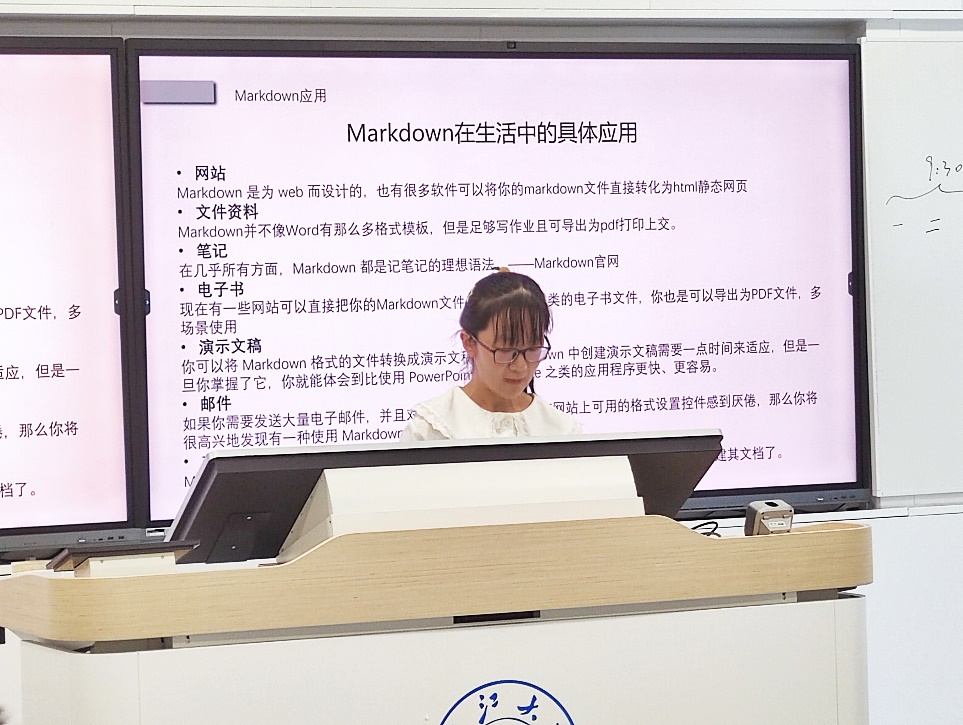
讲座介绍了Git的工作流程，Git基础的操作指令和基本的Linux指令。



## 知识卡片

### Markdown

Markdown是一种可以使用普通文本编辑器编写的轻量级标记语言，通过简单的标记语法，它可以使普通文本具有一定的格式，并可以导出 PPT、LaTex、HTML、Word、PDF、Epub、JPG 等多种格式的文档。



#### Markdown基本语法

标题：Markdown 支持两种标题的语法，类 Setext 和类 atx 形式。 类 Setext 形式是用底线的形式，利用 = （最高阶标题）和 - （第二阶标题），任何数量的 = 和 - 都可以有效果。类 Atx 形式则是在行首插入 1 到 6 个 # ，对应到标题 1 到 6 阶。

字体：Markdown 使用星号（\*）和底线（\_）作为标记强调字词的符号，你可以随便用你喜欢的样式，唯一的限制是，你用什么符号开启标签，就要用什么符号结束。

VS Code。

分割线：在一行中用三个以上的星号、减号、底线来建立一个分隔线，行内不能有其他东西。你也可以在星号或是减号中间插入空格。

表格：Markdown 制作表格使用 | 来分隔不同的单元格，使用 - 来分隔表头和其他行。

文字默认居左

-两边加：表示文字居中

-右边加：表示文字居右

代码块：用 ```或者~~~ 包裹一段代码，并指定一种语言（也可以不指定）

区块：Markdown 区块引用是在段落开头使用 > 符号 ，然后后面紧跟一个空格符号。区块是可以嵌套的，一个 > 符号是最外层，两个 > 符号是第一层嵌套，以此类推。

无序列表：无序列表使用星号(**\***)、加号(**+**)或是减号(**-**)作为列表标记，这些标记后面要添加一个空格，然后再填写内容。

有序列表：有序列表使用数字并加上 **.** 号来表示。

### VS Code

VS Code全称Visual Studio Code，是一款由微软推出的轻量级、高效的源代码编辑器。它既拥有代码编辑器的轻量、快捷的特性，同时又拥有IDE的调试、代码片段等工程师所需的高级特性。其开源，免费，海量插件，外观出色，简洁流畅，支持众多编程语言，支持三大操作系统Windows、Linux、MacOS，深受编程爱好者与程序员的喜爱。



* 安装与配置

从官网下载VS Code并安装，配置编译工具gcc，添加环境变量。



* 插件推荐

Chinese (Simplified)：简体中文插件

C/C++：C/C++语言支持插件

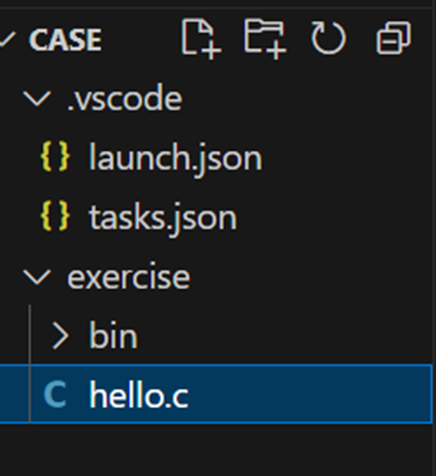
Live Server：Live Server是一个轻量级的本地开发服务器，支持实时预览。当你修改代码并保存时，Live Server会自动刷新浏览器，让你立即看到修改后的效果。

Rainbow Brackets：Rainbow Brackets可以为括号（包括圆括号、方括号和花括号）添加不同的颜色，帮助你更轻松地识别配对的括号，提高代码可读性。此外，它还支持其他符号，如尖括号等。

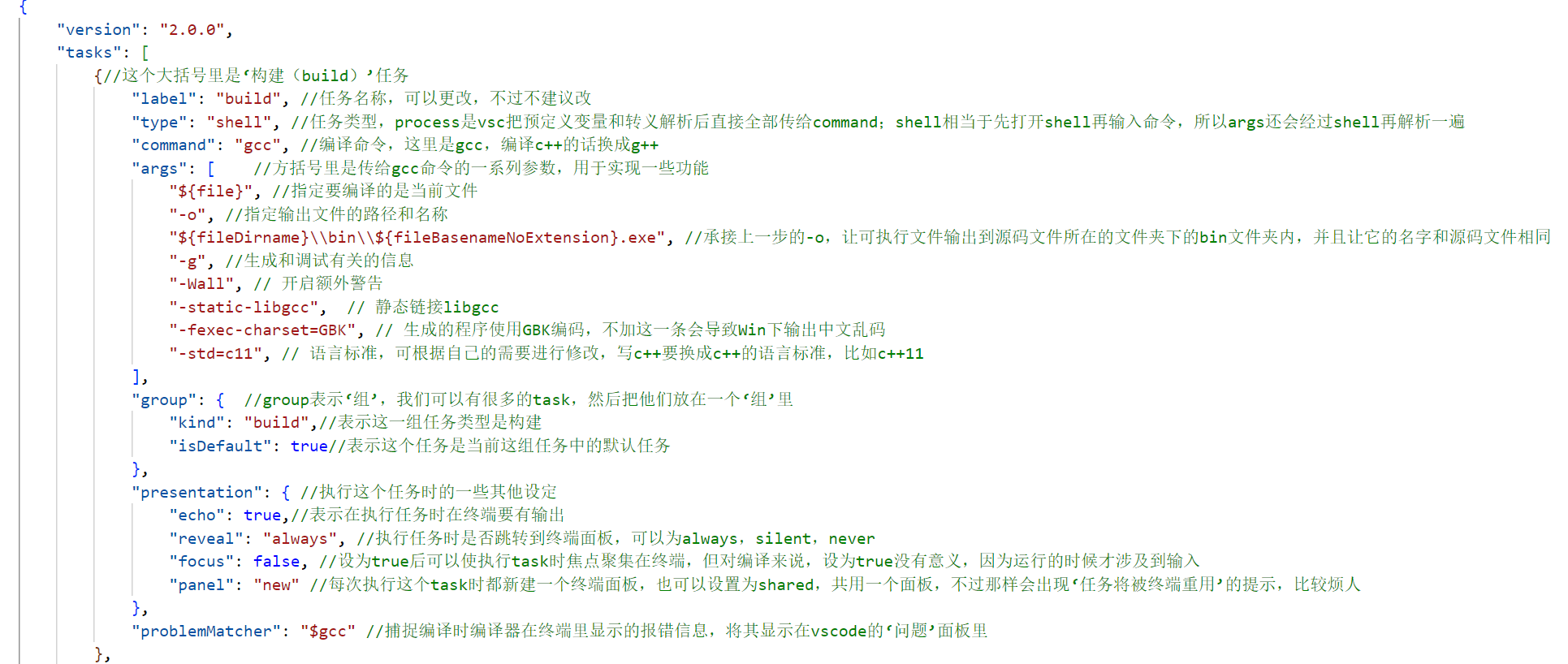
* 文件结构——以C语言为例

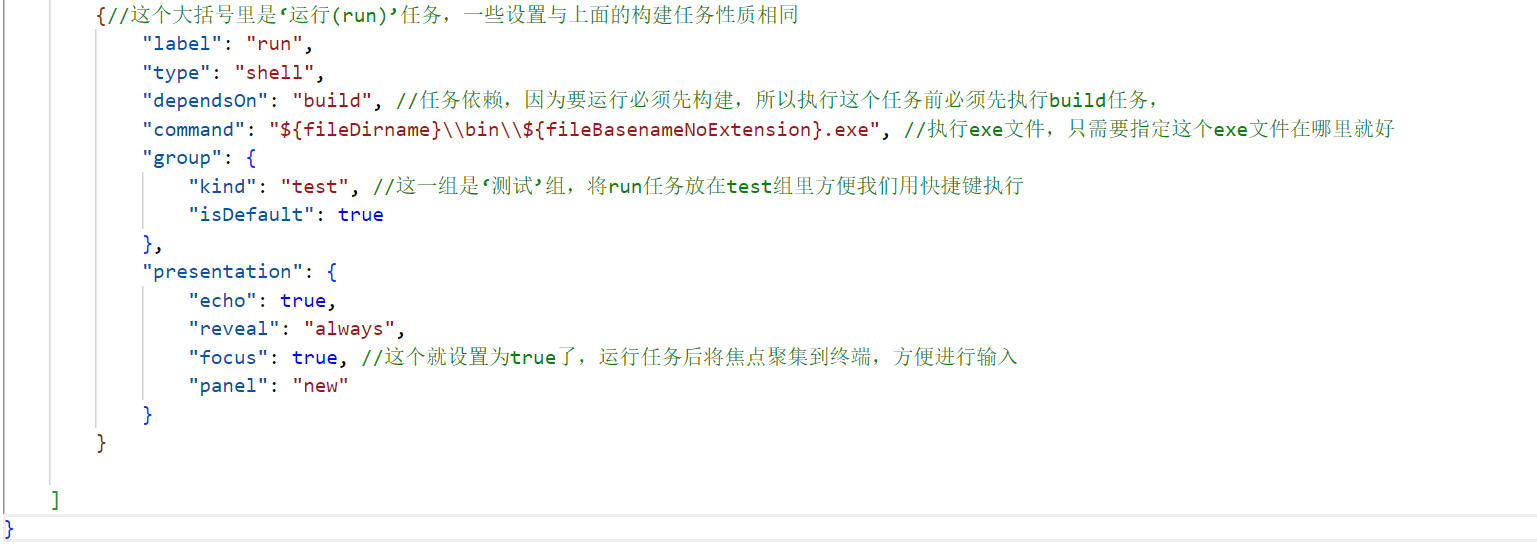
我们的编译和运行就是我们想要vscode执行的任务，为此我们要在tasks.json里写一个task：Build和Run。除了编译和运行，我们还需要进行调试(Debug)，这个就不是通过task来实现的了，而是通过launch.json文件来实现。

1、单文件编译

建立好文件结构：1.首先在CASE文件下新建一个exercise文件夹，来组织源码文件，在exercise下新建hello.c文件，然后在exercise下建一个bin文件夹（注意从属关系，不要建错了）2.CASE文件下建立.vscode文件，在里面新建tasks.json和launch.json

配置tasks文件，用于编译与运行

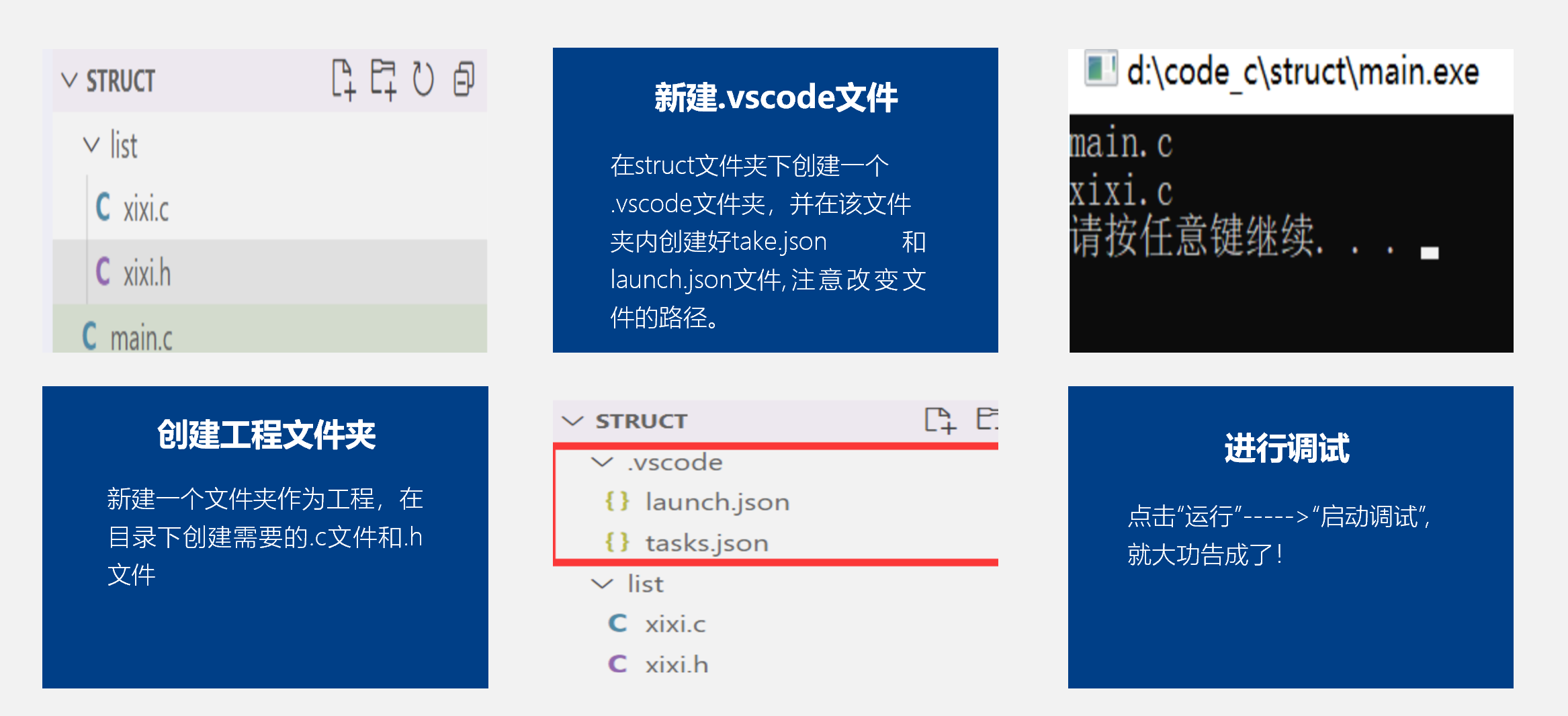




配置launch文件，用于Debug



2、多文件编译



配置task文件



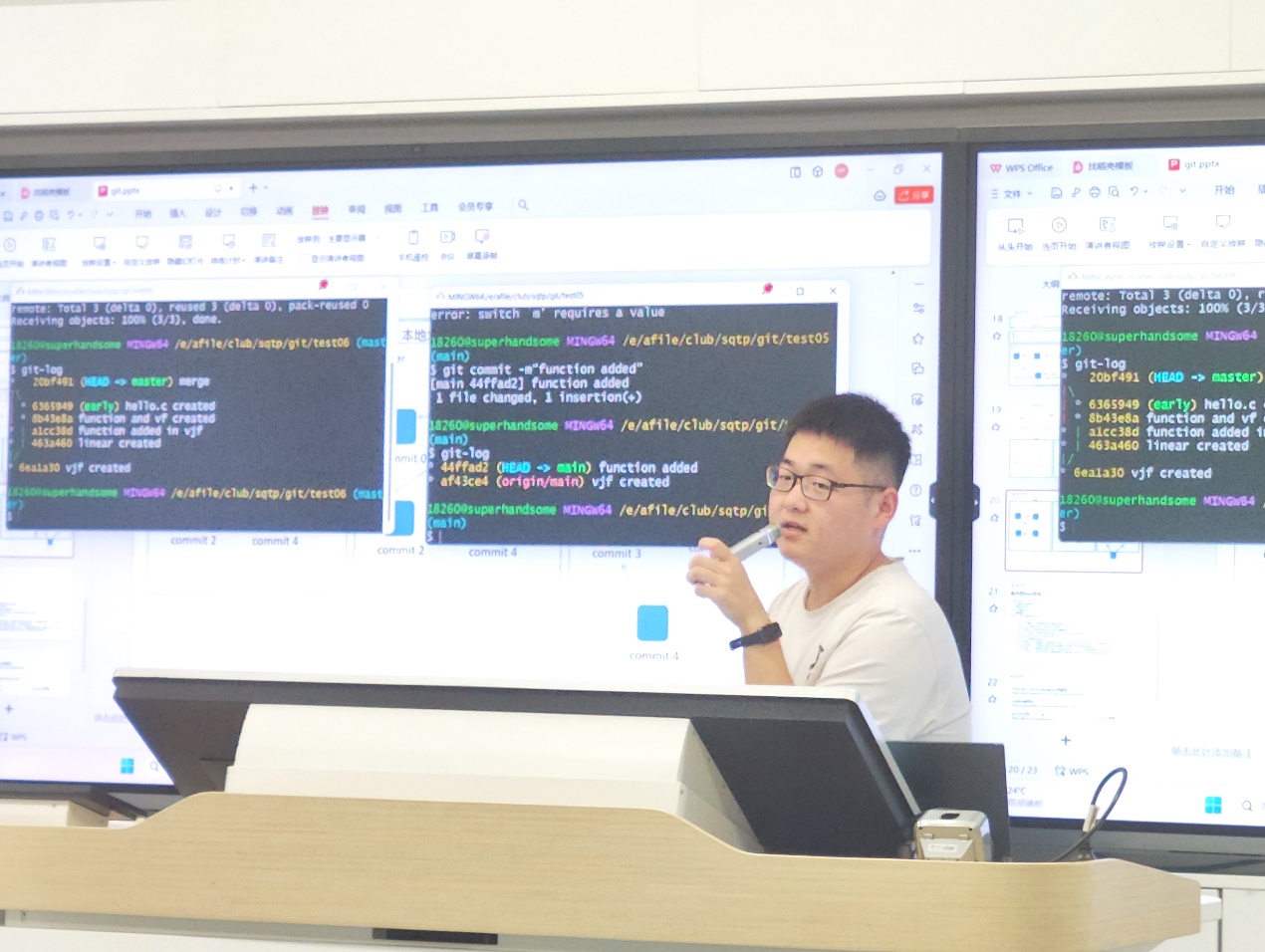


配置launch文件须注意的点和单文件时一样

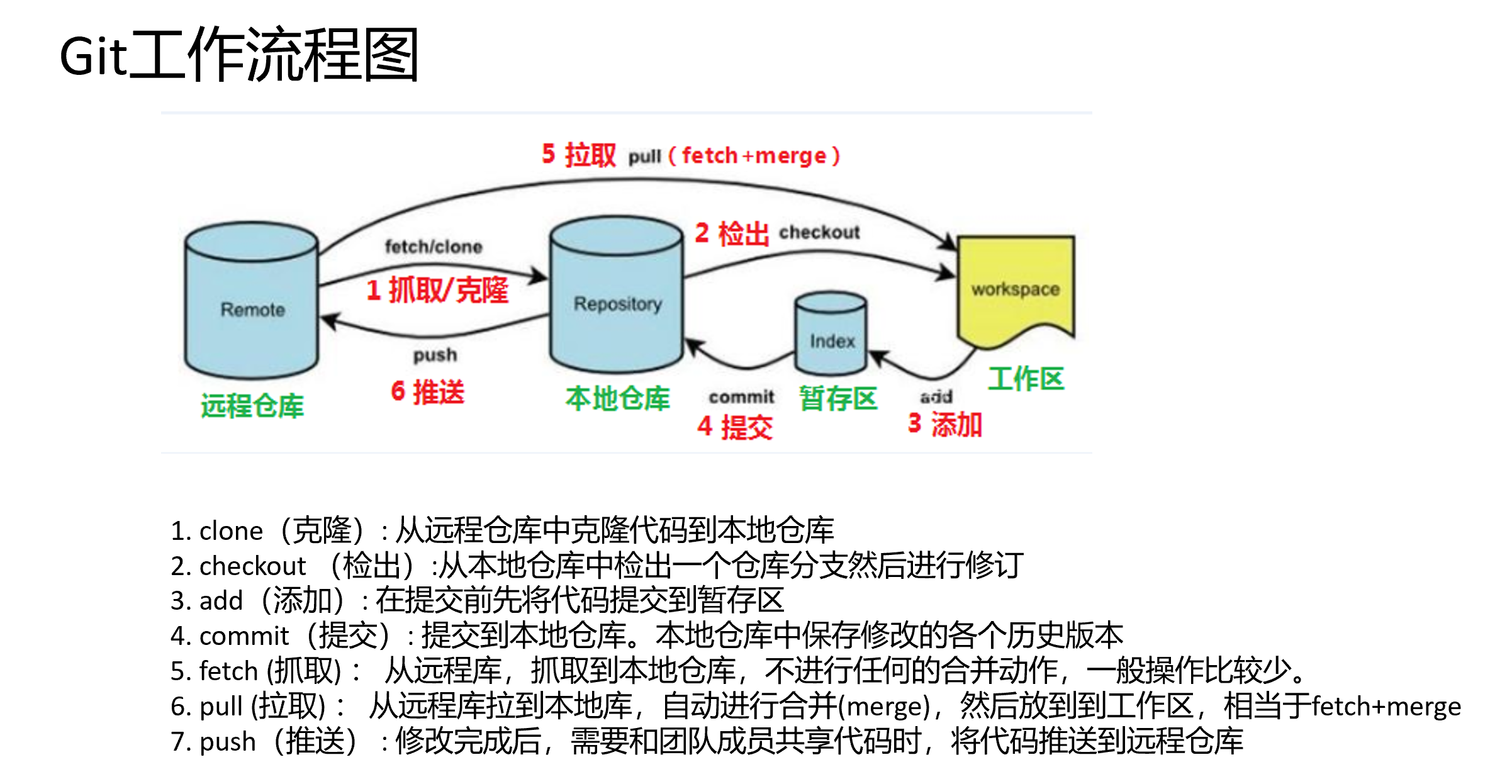


### Git

Git是一款免费、开源的分布式版本控制系统，用于高效敏捷地处理任何或小或大的项目。



* Git的工作流程



* Git基本操作指令

1. git init 创建本地仓库

1. git status (用于显示工作目录和暂存区的状态)

2. git add 文件名 （或git add .) (工作区 --> 暂存区)

3. git commit -m “注释” (暂存区 --> 本地仓库)

4. git log 查看提交日志 命令形式：git log [option]（准备工作配置别名后可直接使用git-log）

5.- options

--all 显示所有分支 --abbrev-commit`使得输出的commitID更简短

--pretty=oneline 将提交信息显示为一行 --graph`以图的形式显示

6. 版本回退

git reset --hard commitID

- - commitID 可以使用git-log或git log指令查看

7. - git reflog 这个指令可以看到已经删除的提交记录

* 基本的linux命令

-cd 进入文件

- ls / ll 查看当前目录

- cat 查看文件内容

- touch 创建文件

- vi vi编辑器

-软件 + 文件 用软件打开文件

1. clone（克隆）: 从远程仓库中克隆代码到本地仓库

2. checkout （移动HEAD）:移动HEAD指针

3. add（添加）: 在提交前先将代码提交到暂存区

4. commit（提交）: 提交到本地仓库。本地仓库中保存修改的各个历史版本

5.git remote add origin url (添加远程仓库）

6.git remote :查看远程仓库

7. fetch (抓取) ： 从远程库，抓取到本地仓库，不进行任何的合并动作，一般操作比较少。

8.merge（合并）：将目标分支合并到当前所在分支，合并后自动提交

9. pull (拉取) ： 从远程库拉到本地库，自动进行合并(merge)，然后放到到工作区，相当于fetch+merge

10. push（推送） : 修改完成后，需要和团队成员共享代码时，将代码推送到远程仓库

什么？你说你错过了这次精品课程？别灰心！~ 关注公众号，更多的计算机知识在等着你~