第二章
本地仓库(Repository)

内容大纲

1. 将shengxiao-bobo项目目录转成一个Git仓库(Repository)
2. 图形化展示Git的四个重要的区域
3. 向shengxiao-bobo项目目录中添加第一个文件

当前设置

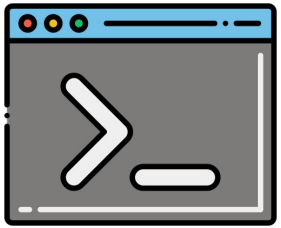
- 下载并安装了git(> 2.28)
- 创建了一个空目录shengxiao-bobo
- 打开命令行窗口并导航到shengxiao-bobo目录
- 安装了VSCode编辑器，并在shengxiao-bobo目录中打开了编辑器
- 设置了user.name和user.email全局Git配置变量

概念

- 仓库(repository/repo) ~ 表示受git版本控制的一个项目目录
  - 本地仓库(Local Repository)
  - 远程仓库(Remote Repository)
    - GitHub/GitLab/Bitbucket
- 教程前半部分学习本地仓库，后半部分引入远程仓库

当前项目目录情况

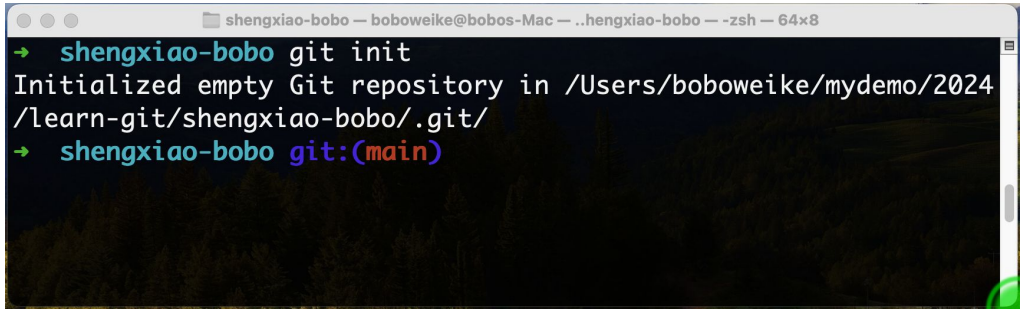
命令 ~ 初始化本地仓库

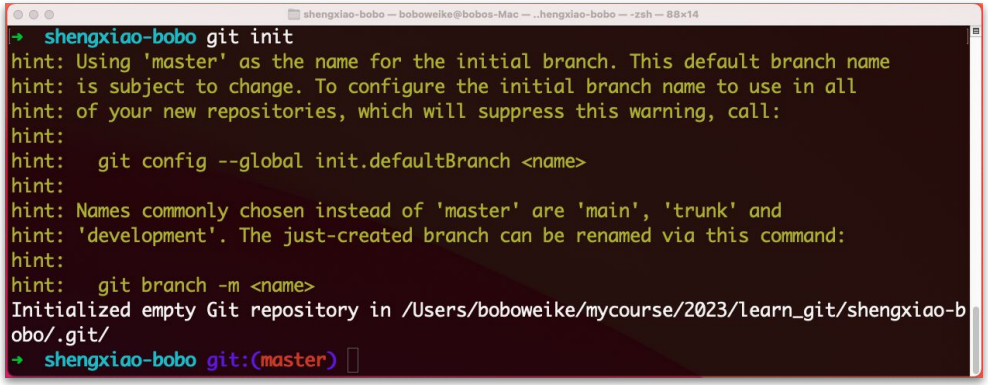


- ***git init***
  - 初始化一个Git仓库
- ***git init -b <branch_name>***
  - 初始化一个Git仓库，同时将初始分支设为<branch_name>
  - git init -b main
  - 全局配置项init.defaultBranch

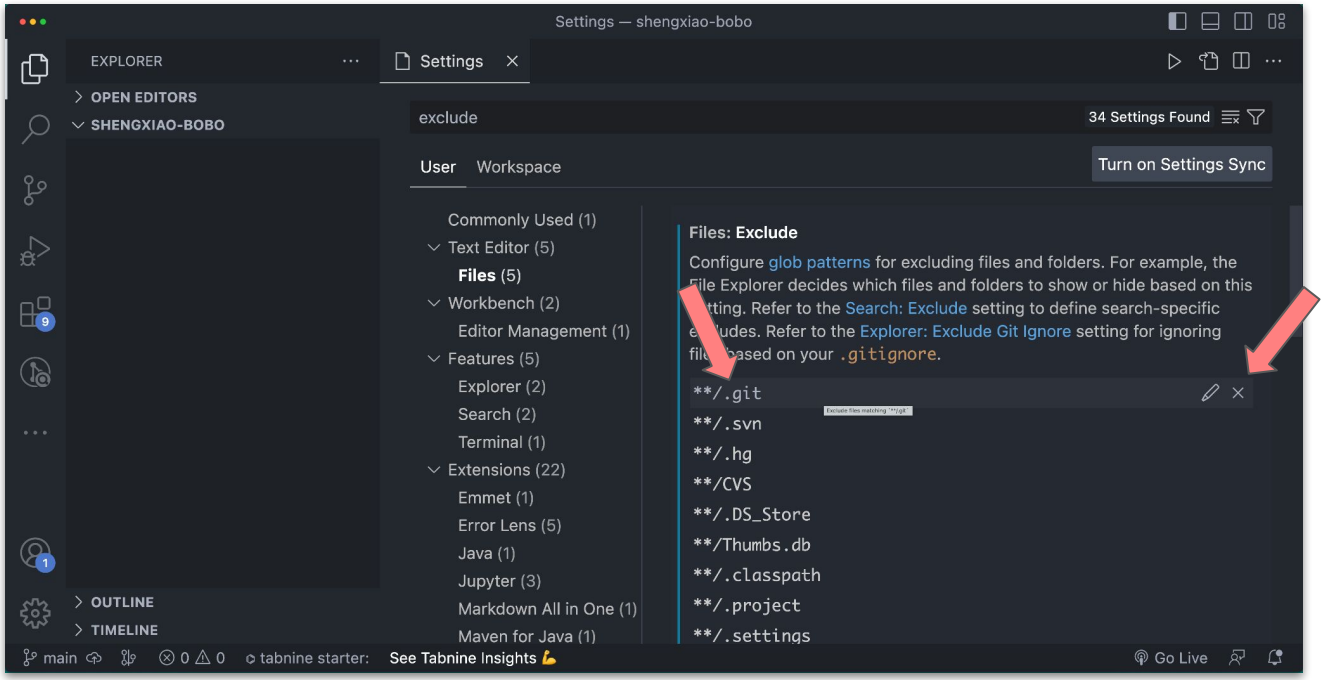
- **main vs master**
  - GitHub is working on replacing the term “master” on its service with a neural term like “main” to avoid unnecessary references to slavery
    - https://stackoverflow.com/questions/64249491/why-does-git-push-main-work-on-github-when-git-push-master-does-not-also-wh

实操

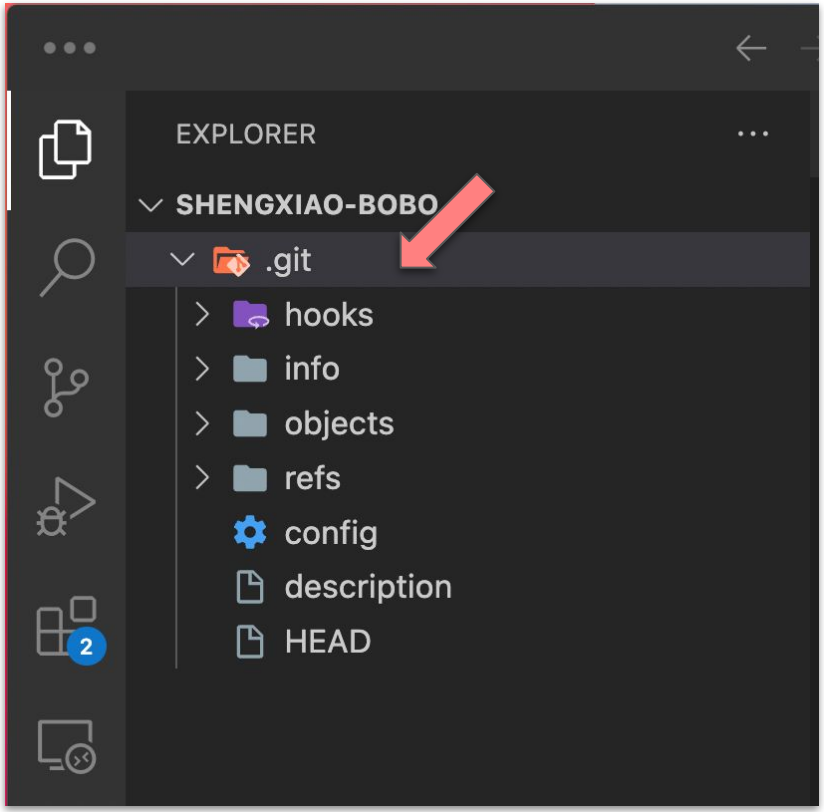

shengxiao-bobo — boboweike@bobos-Mac — ..hengxiao-bobo — -zsh — 64×8
➜ shengxiao-bobo git init
Initialized empty Git repository in /Users/boboweike/mydemo/2024/learn-git/shengxiao-bobo/.git/
➜ shengxiao-bobo git:(main)


shengxiao-bobo — boboweike@bobos-Mac — ..hengxiao-bobo — -zsh — 88×14
➜ shengxiao-bobo git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint: git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint: git branch -m <name>
Initialized empty Git repository in /Users/boboweike/mycourse/2023/learn_git/shengxiao-bobo/.git/
➜ shengxiao-bobo git:(master)

显示隐藏.git目录(1)


Settings — shengxiao-bobo
EXPLORER
Settings
OPEN EDITORS
SHENGXIAO-BOBO
exclude
34 Settings Found
User
Workspace
Turn on Settings Sync
Commonly Used (1)
Text Editor (5)
Files (5)
Workbench (2)
Editor Management (1)
Features (5)
Explorer (2)
Search (2)
Terminal (1)
Extensions (22)
Emmet (1)
Error Lens (5)
Java (1)
Jupyter (3)
Markdown All in One (1)
Maven for Java (1)
Files: Exclude
Configure glob patterns for excluding files and folders. For example, the File Explorer decides which files and folders to show or hide based on this setting. Refer to the Search: Exclude setting to define search-specific excludes. Refer to the Explorer: Exclude Git Ignore setting for ignoring files based on your .gitignore.
**/.git
**/.svn
**/.hg
**/CVS
**/.DS_Store
**/Thumbs.db
**/.classpath
**/.project
**/.settings
OUTLINE
TIMELINE
main
0 0
tabnine starter:
See Tabnine Insights
Go Live

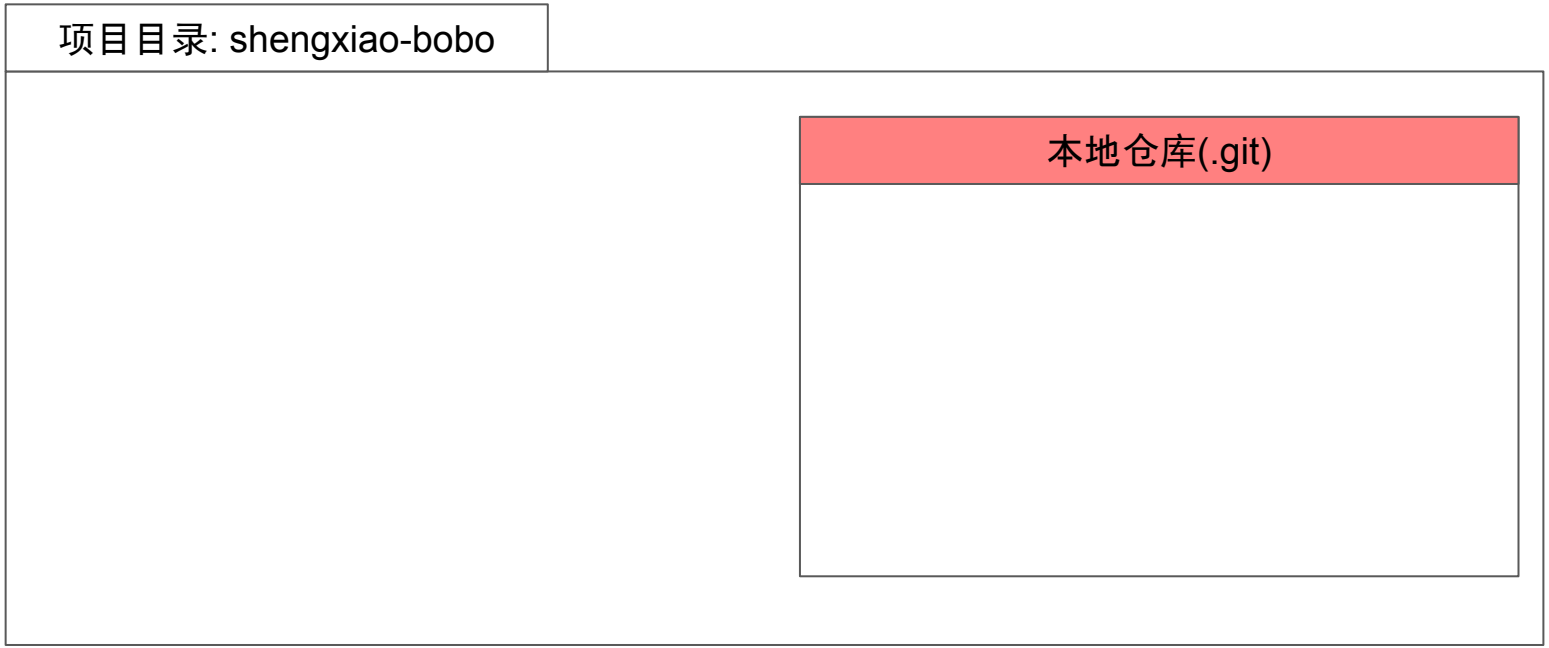
显示隐藏.git目录(2)



⚠️

勿修改或删除.git目录中的内容

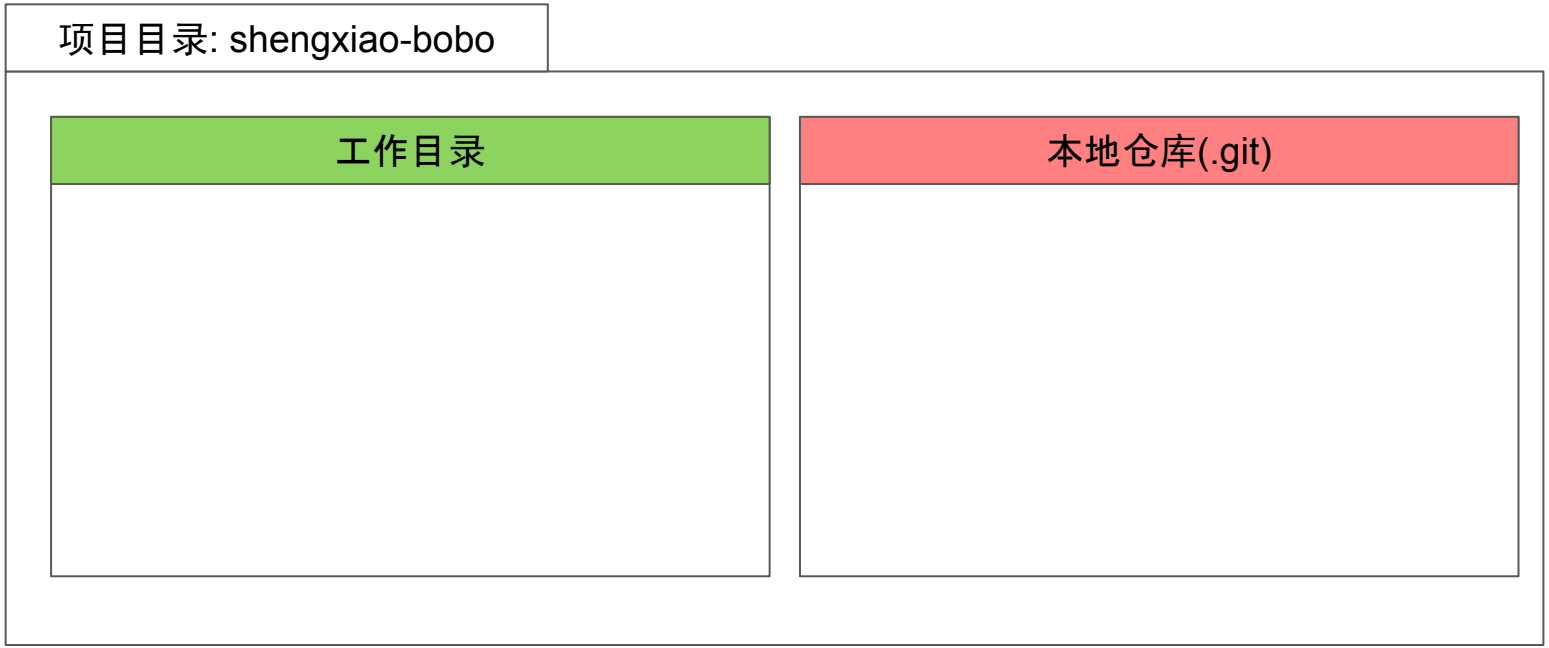
当前项目目录情况



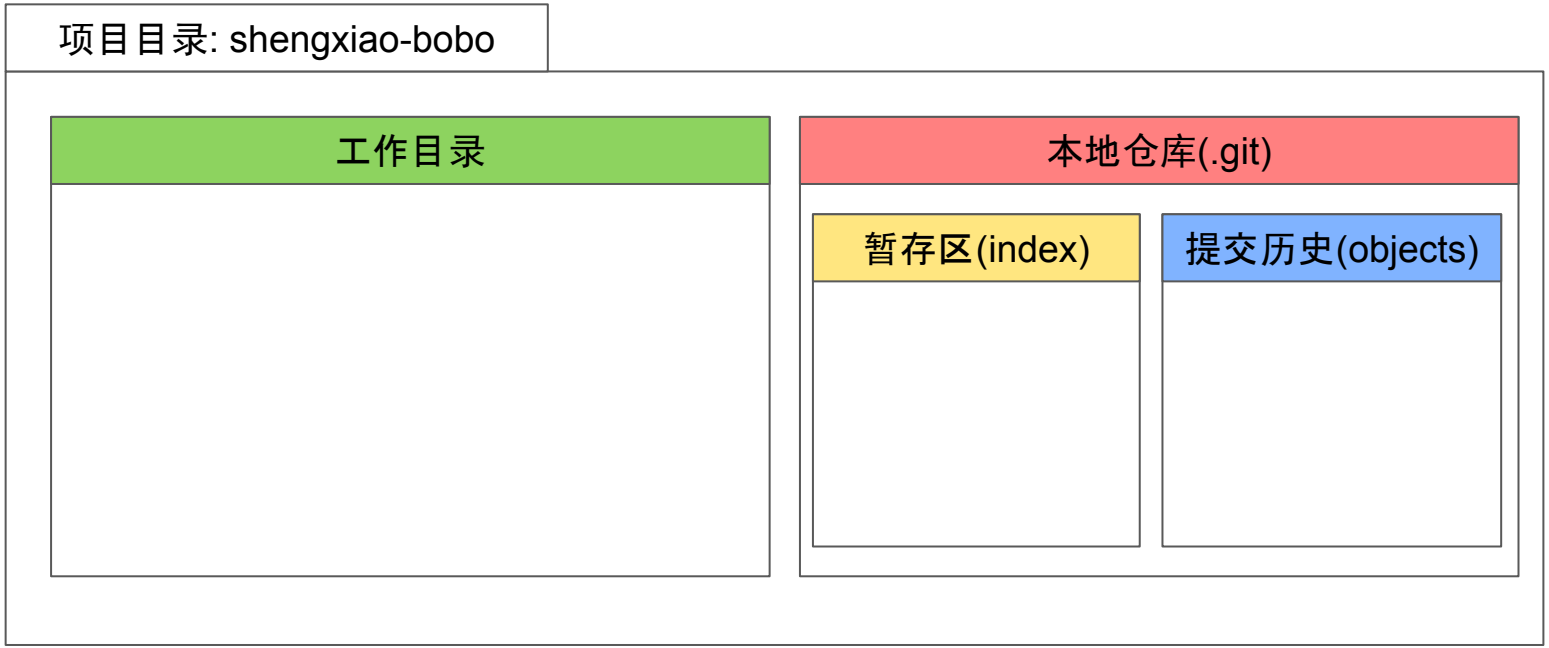
概念～Git工作区

1. 工作目录(Working directory)
    a. 项目文件/目录编辑区(增删改)
    b. 工作区(workbench)
2. 本地仓库(Local repository)
3. 暂存区(Staging area)
    a. 草稿区，可添加/移除文件
    b. 由index文件表示，在暂存区至少添加一个文件时才会创建index文件
    c. Ask ChatGPT why?
4. 提交历史(Commit History)
    a. 什么是提交(commit)？
        i. 项目的一个版本/快照(snapshot)
        ii. 包含针对这个提交的所有文件引用
        iii. 对应一个唯一的commit hash/commit ID，40个字符的hash值
    b. 保存所有提交历史记录的地方
    c. 保存在objects目录中

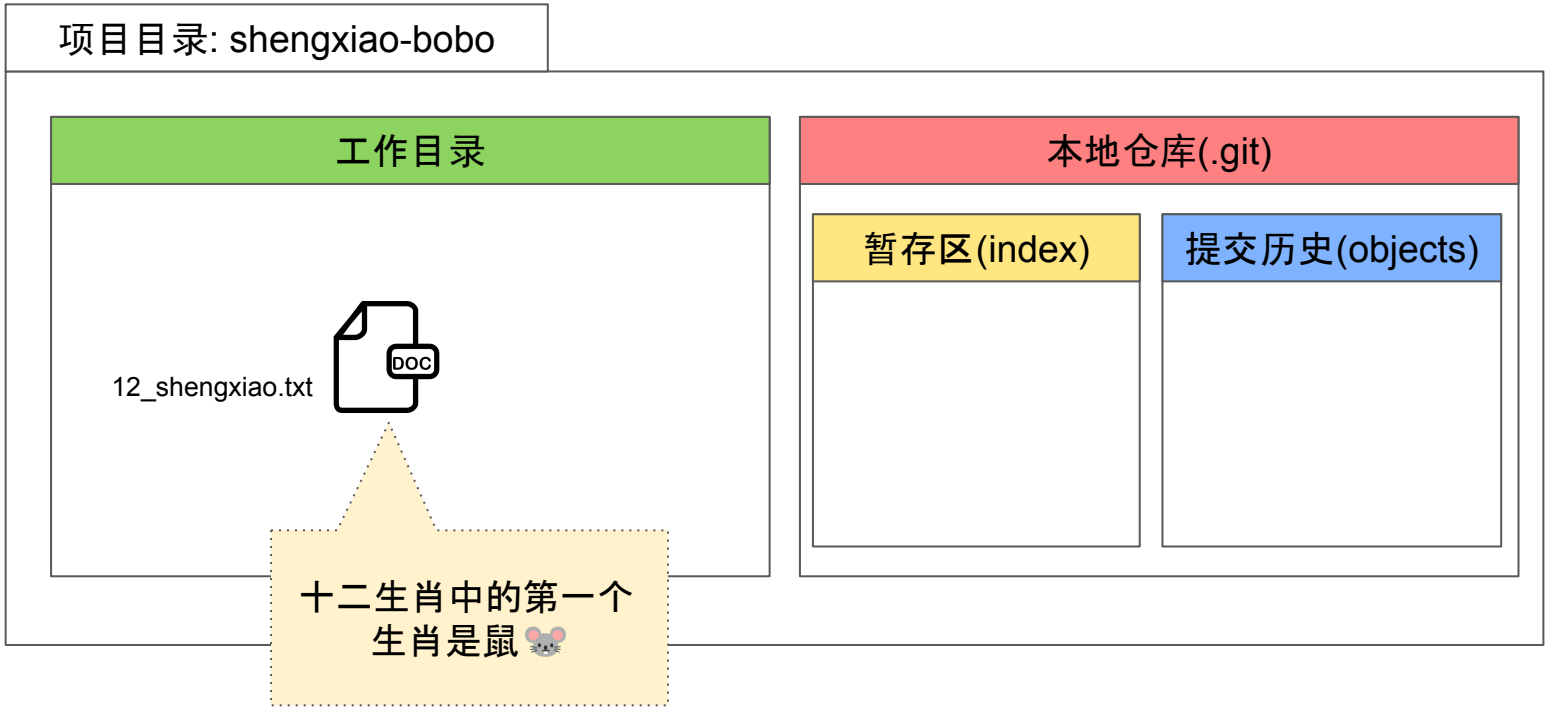
工作目录和本地仓库



暂存区和提交历史



操作:添加一个文件到工作目录



概念

- 未跟踪的文件(untracked file)
  - 在工作目录中，未受Git版本控制的文件
  - 未在暂存区和提交历史中
- 跟踪的文件(tracked file)
  - 被添加到暂存区并进入提交历史的文件
  - 受Git版本控制的文件

小结和预告

- 本课小结
  - 将12生肖项目转成了一个Git仓库(repository)
  - 图形化展示Git项目的四区域
    - 工作目录
    - 本地仓库
    - 暂存区
    - 提交历史
  - 在12生肖项目中添加了一个文件
    - 未跟踪的文件(untracked file)
    - 跟踪的文件(tracked file)
- 下一课
  - 提交commit
  - 图像化展示在提交时，Git的四个区域的变化