



[返回首页](#)

[芋道源码 —— 知识星球](#)

我是一段不羁的公告！

记得给芬芳这 3 个项目加油，添加一个 STAR 噢。

<https://github.com/YunaiV/SpringBoot-Labs>

<https://github.com/YunaiV/onemall>

<https://github.com/YunaiV/ruoyi-vue-pro>

[2019-09-20](#)

[Redis](#)

精尽 Redis 源码分析 —— 调试环境搭建 (Redis 6.X 版本)

芬芳是 MacOS ， 胖友如果是 Windows 的话，可以参考本文，略作修改。

今儿，我们来搭建一个 Redis 调试环境，目标是：

启动 Redis Server ， 成功断点调试 Server 的启动过程。

使用 `redis-cli` 启动一个 Client 连接上 Server，并使用 `get key` 指令，发起一次 key 的读取。

视频可见 B 站：<https://www.bilibili.com/video/BV12X4y1c79z>

芬芳比较腼腆，大家给个三连支持一下，感恩 1024~

1. 依赖工具

1.1 CLion

下载地址：<https://www.jetbrains.com/clion>

CLion 是 JetBrains 团队提供的 C/C++ 开发工具。所以，所以和我们平时使用的 [IDEA](#) 差别不大。

这里，芬芳使用的是 CLion 版本是 2021.1.3 。

1.2 CMake

从 CLion [2020.2](#) 版本开始，可以支持 Makefile 类型的项目，而 Redis 就是使用 Makefile 构建的项目，所以我们无需再安装 CMake 啦！！！！

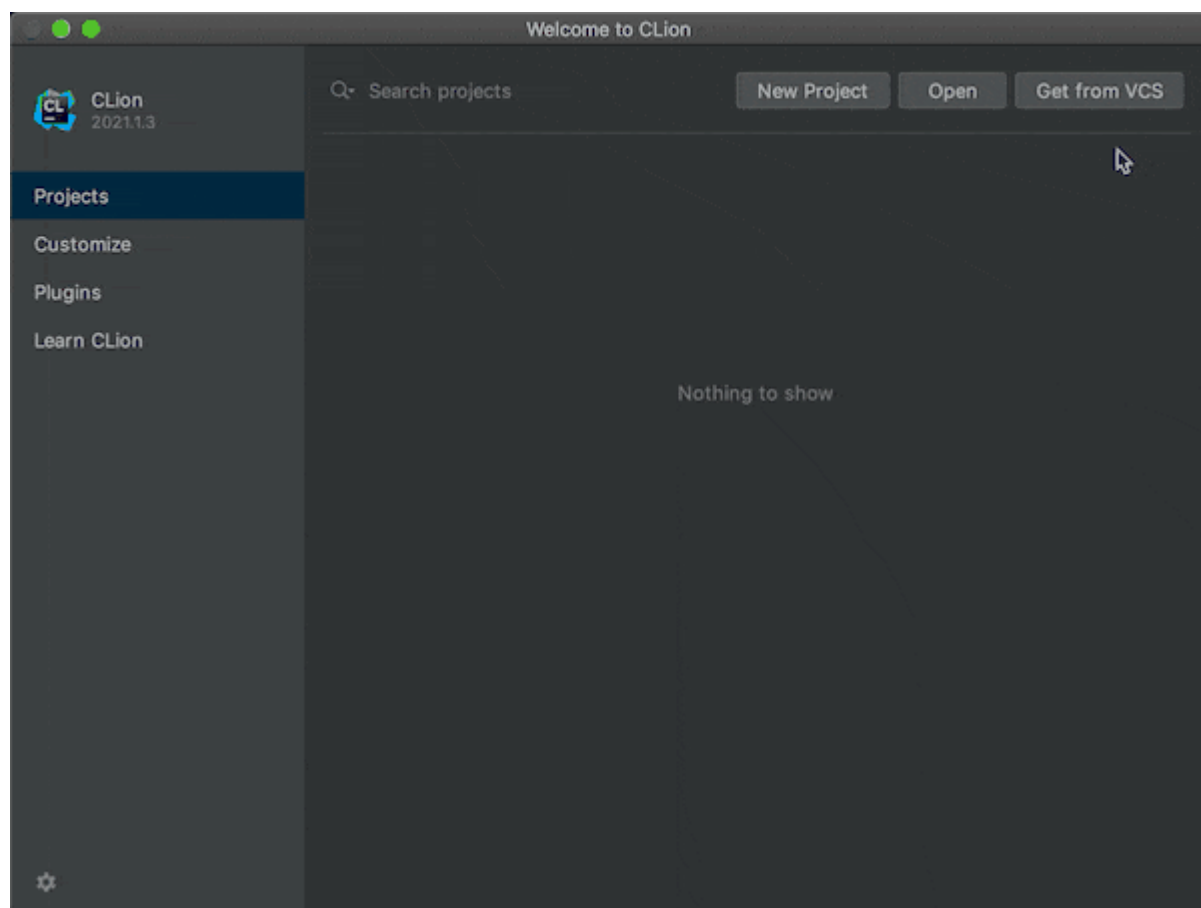
可能胖友对 Makefile 或是 CMake 不太了解？！没关系，这个小节主要是给，之前尝试搭建 Redis 调试环境，结果被 CMake 的胖友。

不影响后续的内容的学习！

2. 源码拉取

使用 CLion 从官方仓库 <https://github.com/antirez/redis> 克隆项目。操作如下图所示：

友情提示：如果网络不是很好的胖友，可以选择和芳芳一样，使用 Gitee 提供的镜像仓库 <https://gitee.com/mirrors/redis.git>



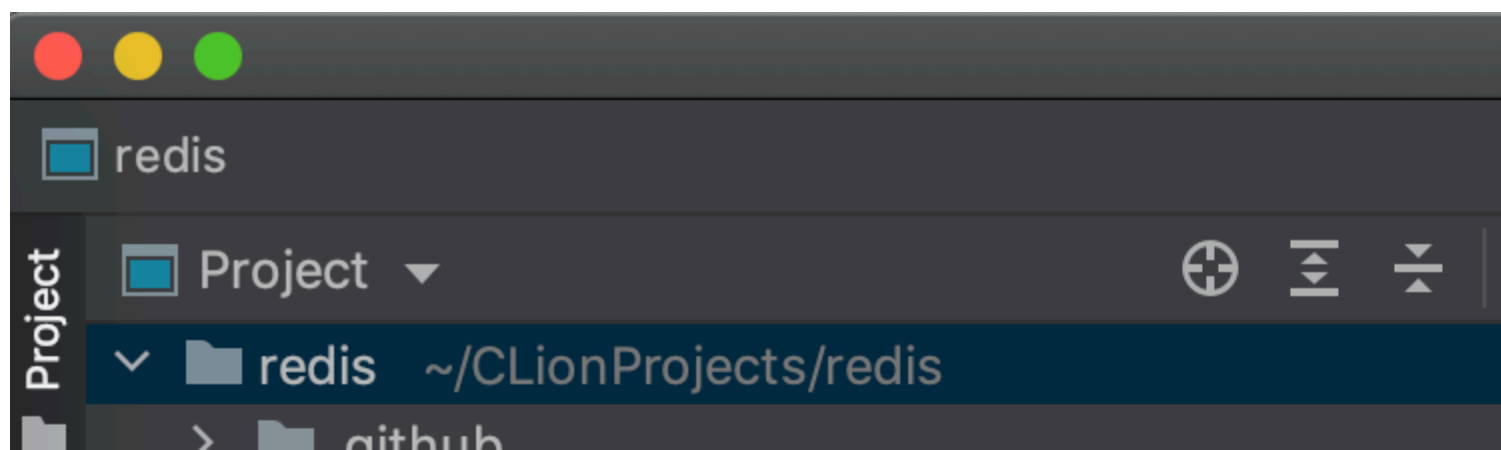
这里，我们使用的 Redis 版本是 6.2.5。

友情提示：胖友可以考虑 Fork 下[官方仓库](#)，为什么呢？

既然开始阅读、调试源码，我们可能会写一些注释，有了自己的仓库，可以进行自由的提交。

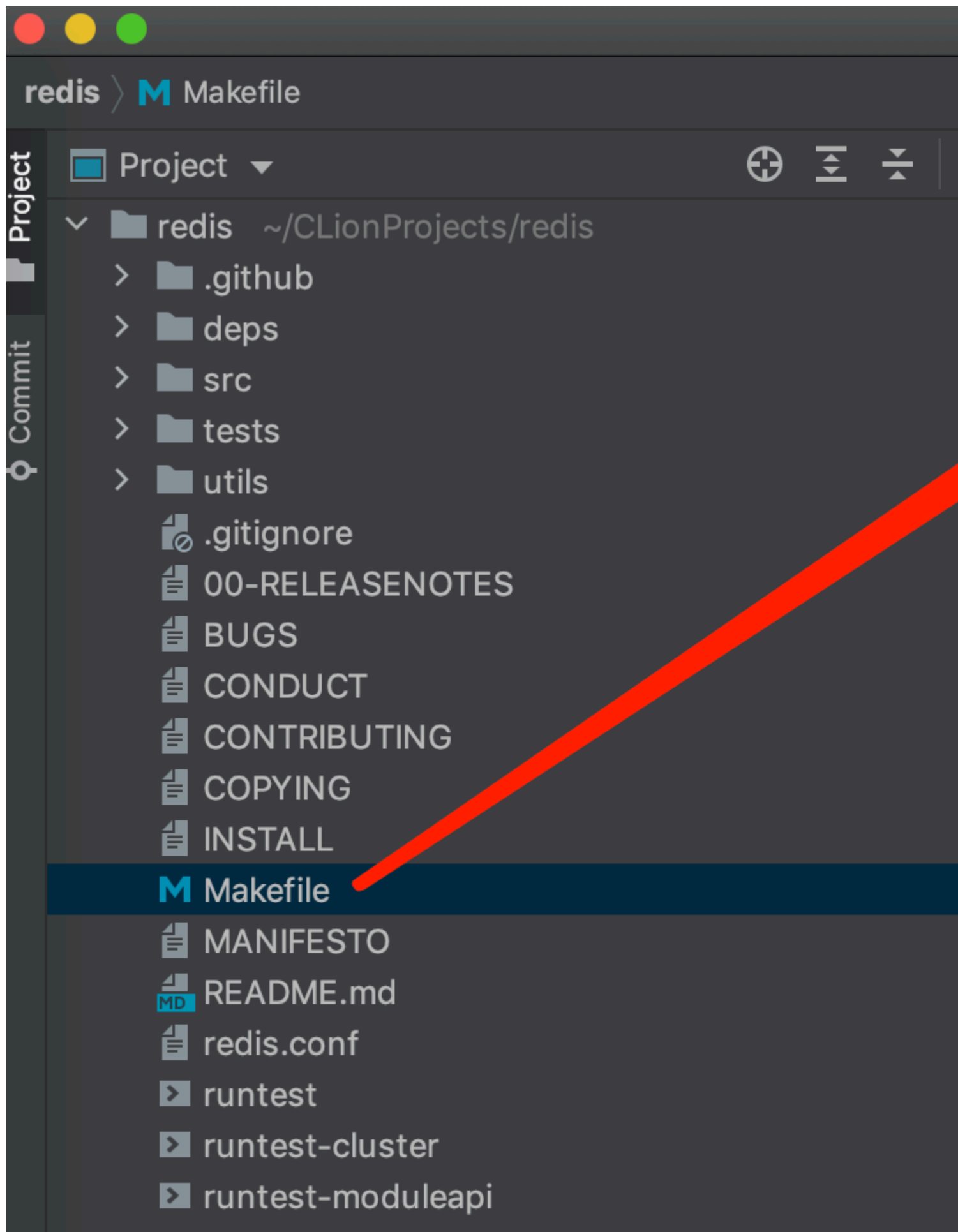
3. 导入项目

克隆完项目后，CLion 会进行 Importing 导入项目，耐心等待一下。如下图所示：



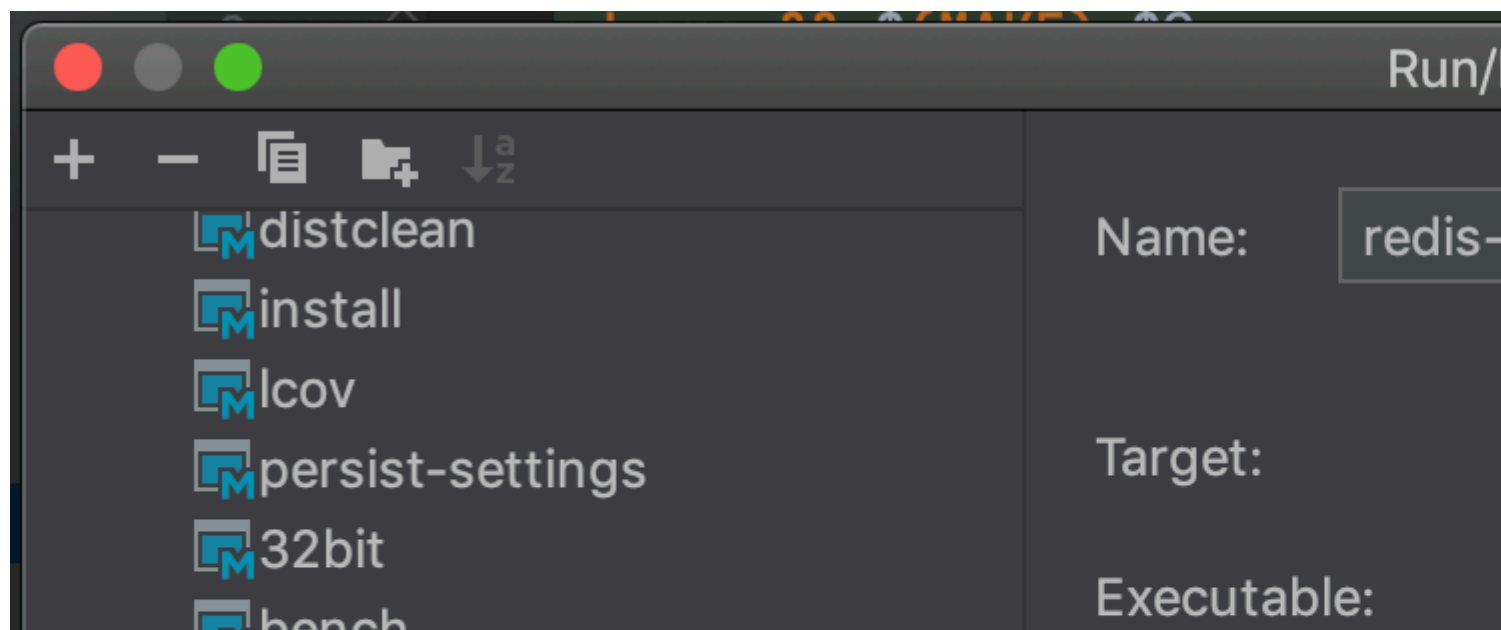
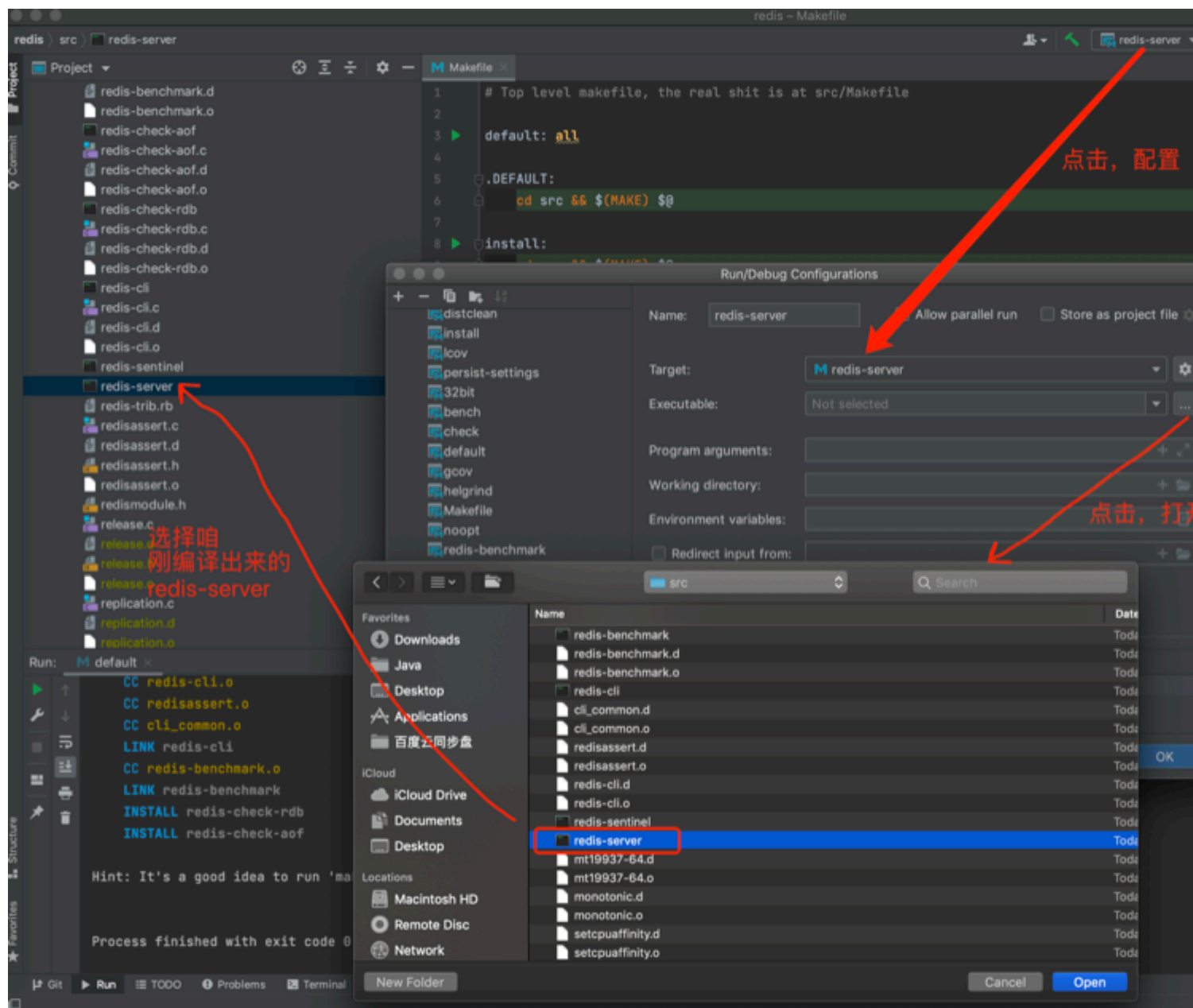
4. 编译项目

导入完成后，打开 Makefile 文件，点击 default:all 绿色箭头，开始项目的编译。如下图所示：

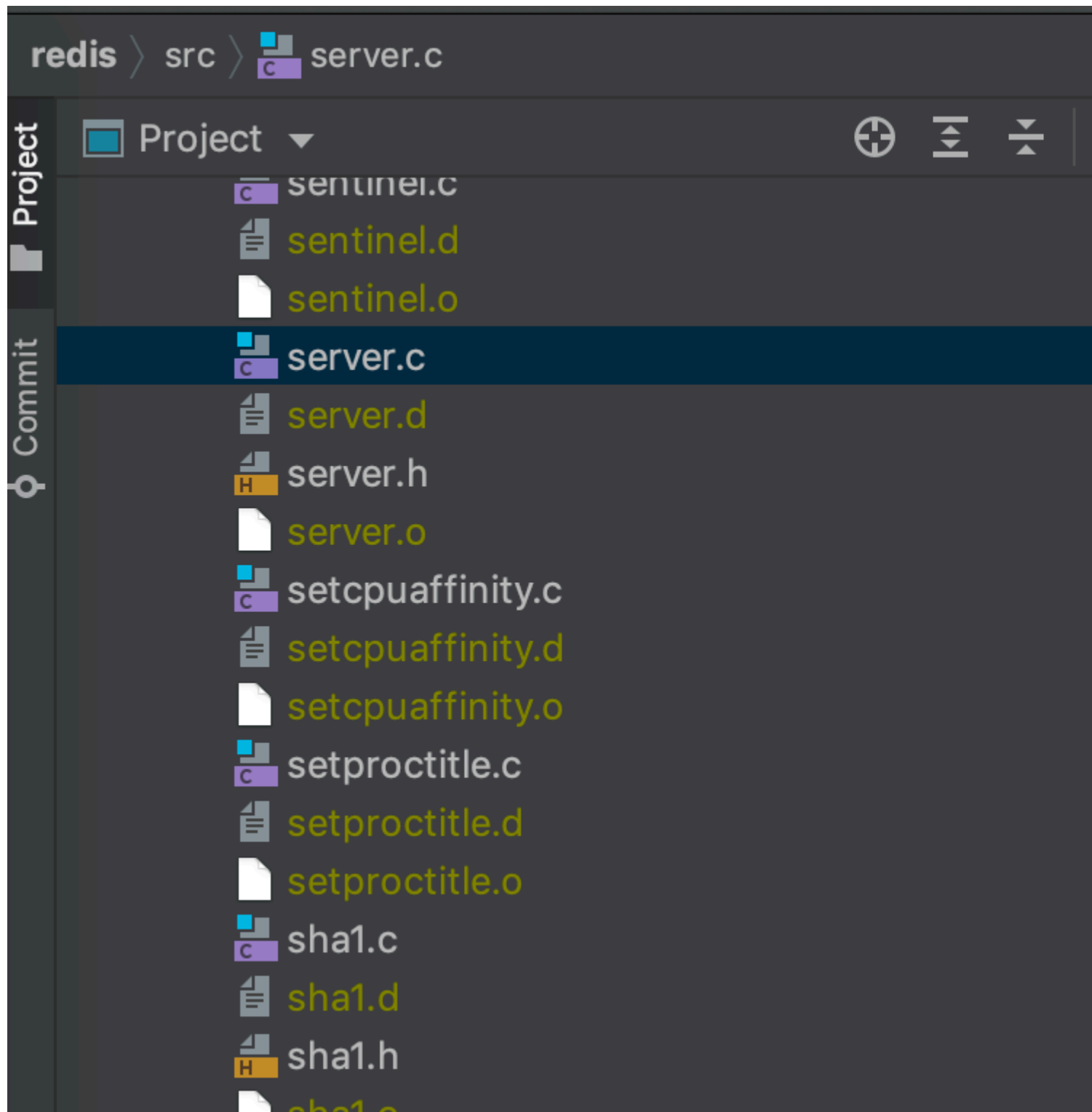


5. 运行 Redis Server

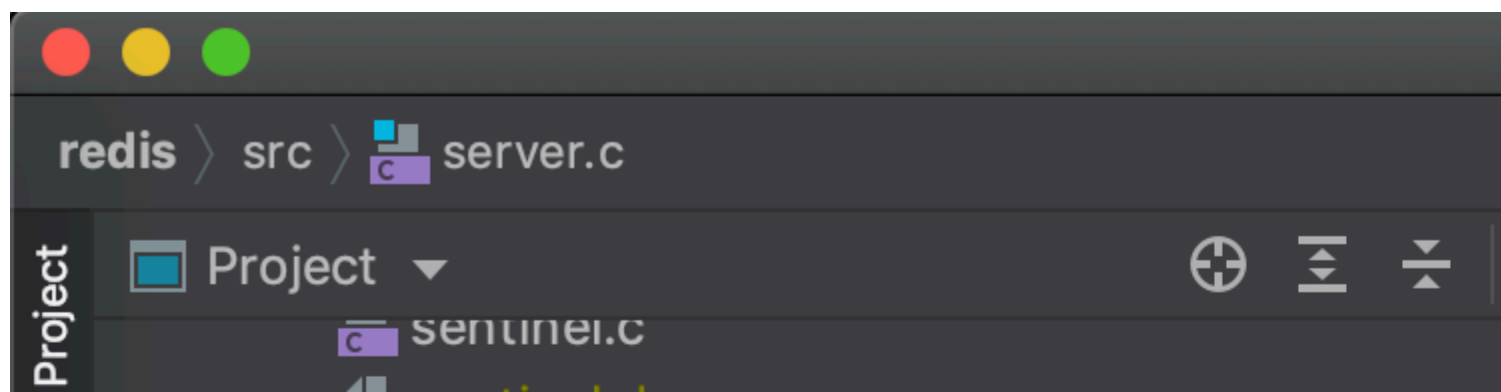
① 配置 Redis Server 的启动项，操作如下图所示：



② 打开 `server.c` 文件，在 `#main(...)` 方法上，添加一个方法断点，它是 Redis Server 的启动入口。如下图所示：



③ Debug 右上方的 `redis-server` 启动项，成功进入 `#main(...)` 方法的断点。如下图所示：



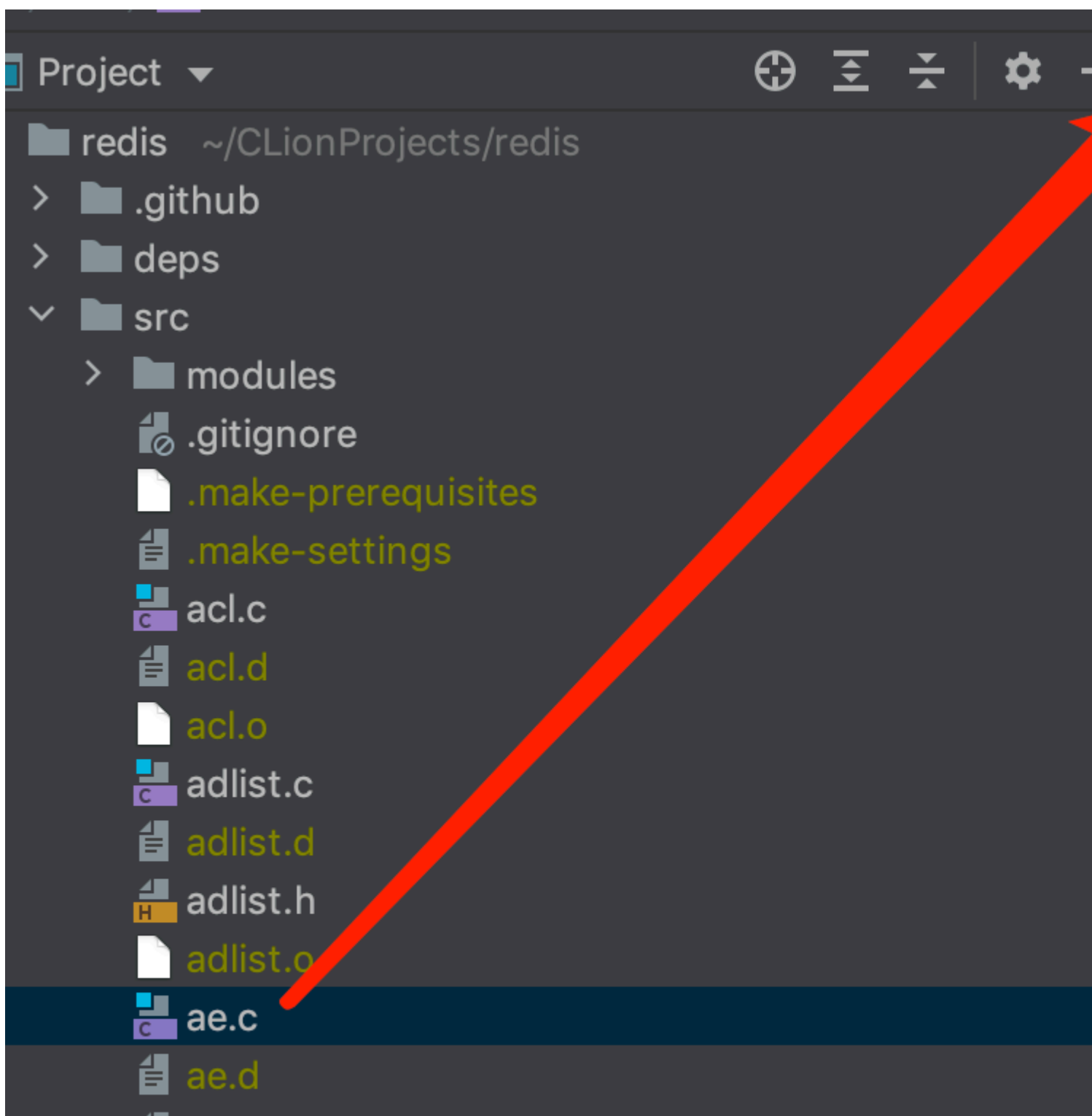
至此，我们已经完成了我们的第一个小目标“启动 Redis Server，成功断点调试 Server 的启动过程”。

点击左下方的绿色小箭头，恢复 Redis Server 的启动，不然等会 Redis Client 都连接不上来。在 CLion 的控制台，我们会看到 Redis Server 启动成功的日志如下：

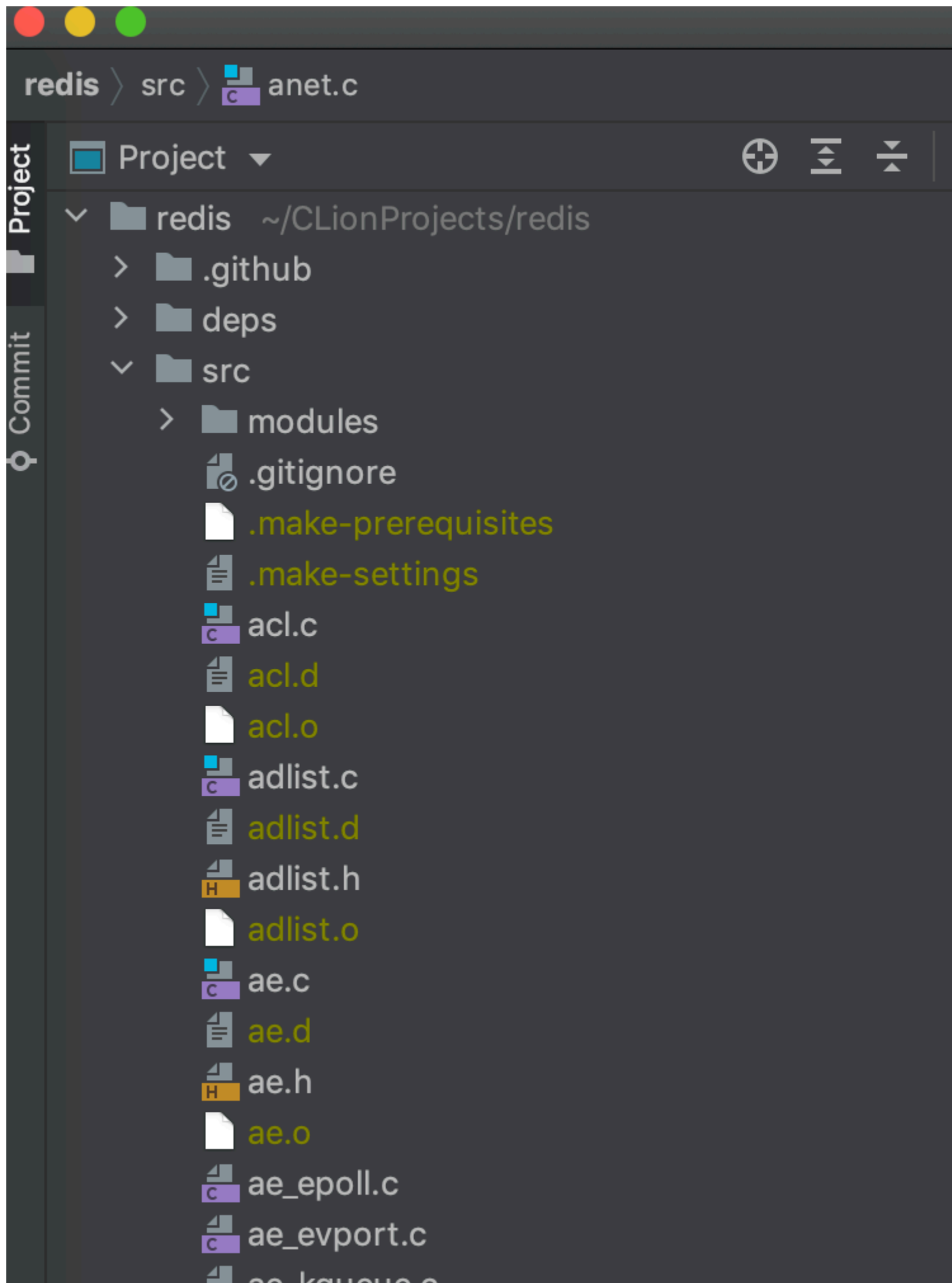
```
22315:M 28 Jul 2021 01:23:37.535 # Server initialized
22315:M 28 Jul 2021 01:23:37.535 * Ready to accept connections
```

6. 运行 Redis Client

① 打开 ae.c 文件，在 #aeProcessEvents(...) 方法的如图处，打上一个断点，用于调试 Redis Server 处理各种来自 Redis Client 的 IO 事件。如下图所示：



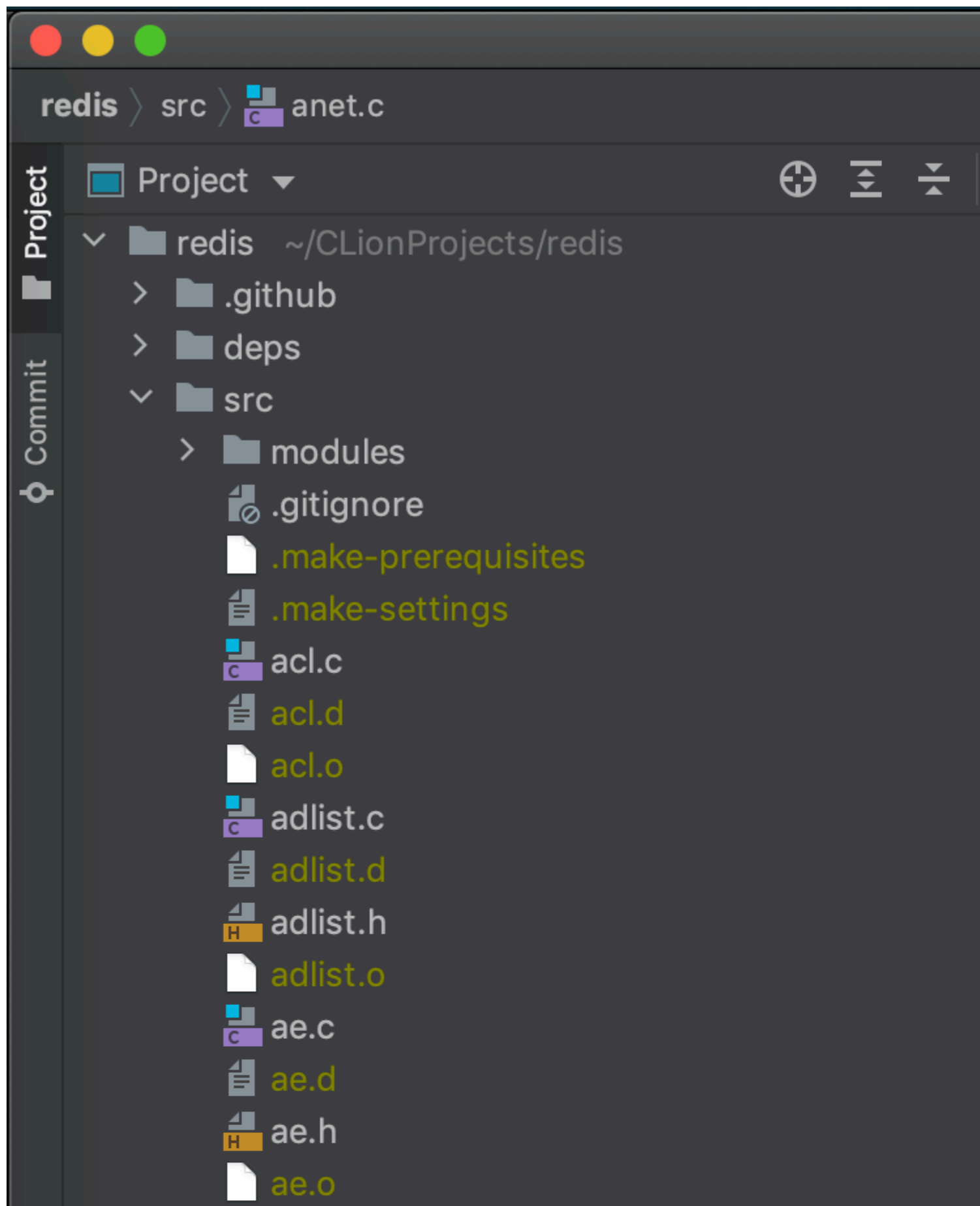
② 打开 IDE Terminal，运行 `redis-cli` 启动一个 Redis Client，连接上 Redis Server。如下图所示：



此时，我们在 `ae.c` 的 `#aeProcessEvents(...)` 的断点成功进入，Redis Server 收到 Redis Client 的连接事件。

打开 Debug 窗口，点击左下方的绿色小箭头，恢复 Redis Server 的执行。

③ 回到 `redis-cli` 命令行，输入 `get key` 指令，向 Redis Server 发起一次 `get` 请求。效果如下图所示：



此时，我们在 `ae.c` 的 `#aeProcessEvents(...)` 的断点又一次进入，Redis Server 收到 Redis Client 的 `get` 请求。

至此，我们已经完成了我们的第二个小目标“使用 `redis-cli` 启动一个 Client 连接上 Server，并使用 `get key` 指令，发起一次 `key` 的读取”。

7. 源码解析

Redis 源码是使用 C 实现的，对于 Java 程序员的我们来说，还是有一定“门槛”的，所以最好借助下市面上 Redis 相关的书籍。

① [《Redis 设计与实现》](https://github.com/YunaiV/redis-3.0-annotated.git)，针对 Redis 3.X 版本，豆瓣评分 8.6 分，对应详细注释的 Redis 仓库 <https://github.com/YunaiV/redis-3.0-annotated.git>。

这本书芳芳和蛮多朋友都看过，虽然针对的 Redis 版本比较老，考虑到 Redis 核心代码变动其实不多，还是非常不错的选择。

② [《Redis5设计与源码分析》](#)，针对 Redis 5.X 版本，豆瓣评分 6.8 分。

主要优点是针对 Redis 5.X 版本，比较新。具体内容如何，芳芳也不是很确定。可以备一本，在看到源码卡壳的时候，作为字典查一查，也是不错的选择哈~

③ [《Redis源码剖析与实战》](#)，针对 Redis 版本不详，极客时间 2021 年出的，可能是 Redis 6.X 版本。

这个专栏，目前还在更新，芳芳准备养肥之后，花 1-2 周撸一撸，嘿嘿！作者（蒋德钧）之前出的[《Redis核心技术与实战》](#)非常不错，我大飞哥好评连连！

END~

我是芳芳，一个正在某厂做 Web IDE 的死胖子，干掉本地 IDEA 和 VSCode！

欢迎添加我的微信号 `naiziguo`，一起网上冲浪 ...

文章目录

1. [1. 1. 依赖工具](#)
 1. [1.1. 1.1 CLion](#)
 2. [1.2. 1.2 CMake](#)
2. [2. 2. 源码拉取](#)
3. [3. 3. 导入项目](#)
4. [4. 4. 编译项目](#)
5. [5. 5. 运行 Redis Server](#)
6. [6. 6. 运行 Redis Client](#)
7. [7. 7. 源码解析](#)

2014 - 2023 芋道源码 |
总访客数 次 && 总访问量 次
[回到首页](#)