**可能会有用的Clion技巧—— Mac篇**

字数1657 阅读9934 评论21 喜欢12

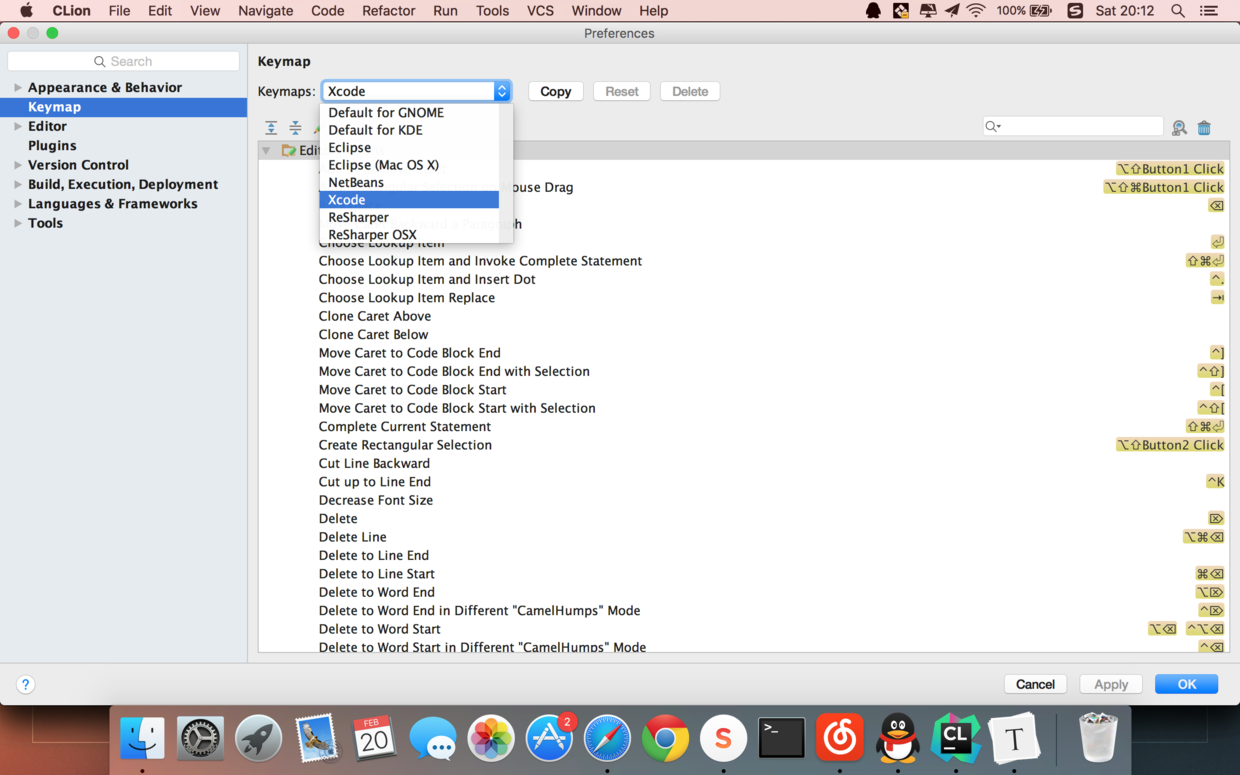
本文已更新，[点击访问](http://www.jianshu.com/p/18c5e8d620b2)

最近学习 C++，首选的代码编辑器是 Sublime Text，这款**轻便、颜值高、功能强大**的编辑器相信是很多人敲代码的首选，同样 Sublime 也是我最喜爱的编辑器，但是编辑器不如 IDE 的一点可能就是调试功能了，其实在 Clion 之前 Mac 平台上难说有一款综合体验非常好的 C++ IDE，很多人都翘首以盼 Clion 的出世，毕竟 JetBrains 出品，必属精品。

先提一句，JetBrains 家几乎所有的产品都支持 student license，一次授权时间为一年，到期需要重新验证 edu 邮箱，建议学生党们通过该方式激活，其他有能力的用户可以购买授权支持正版。

* 快捷键配置

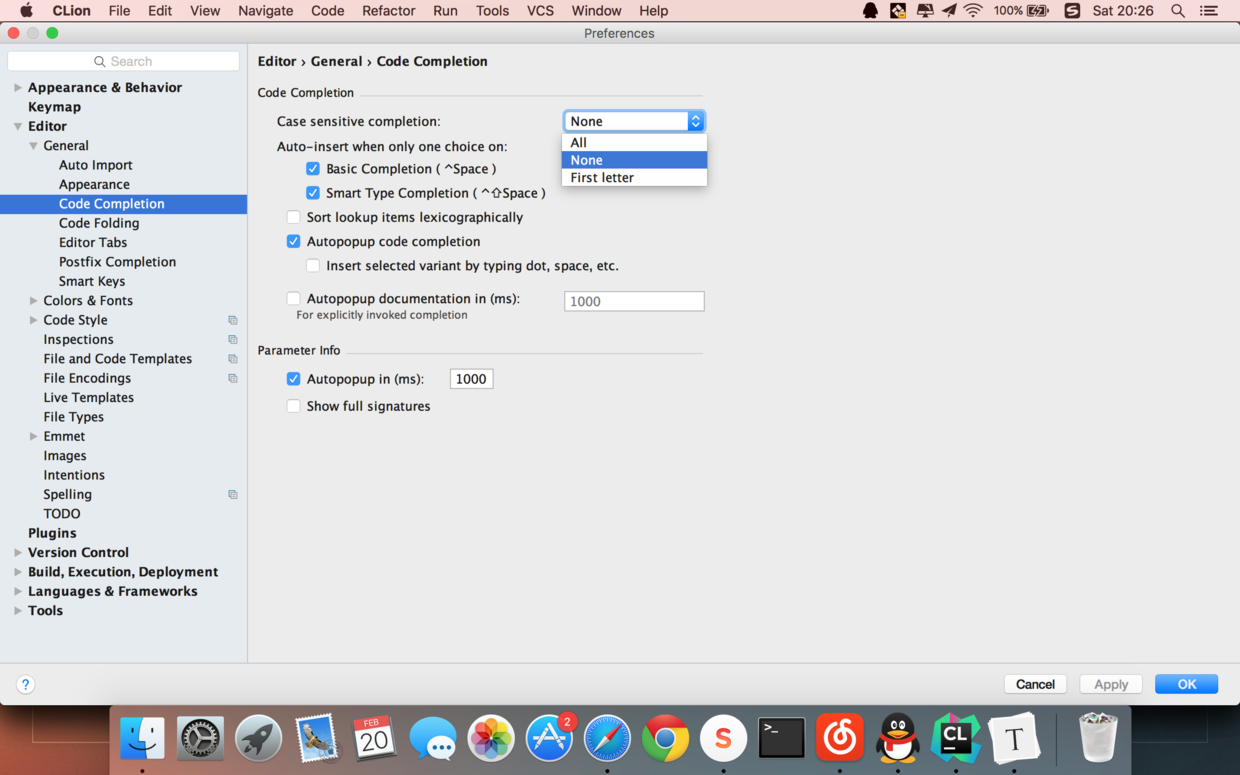
Clion 的默认快捷键风格是 Visual Studio，由于 Mac 上一般用 ⌘ 代替 control，在 Setting—>Keymap 里修改成 Xcode 风格即可，Clion 默认提供8种风格，也支持用户 DIY。



*快捷键风格*

* 代码提示的匹配模式

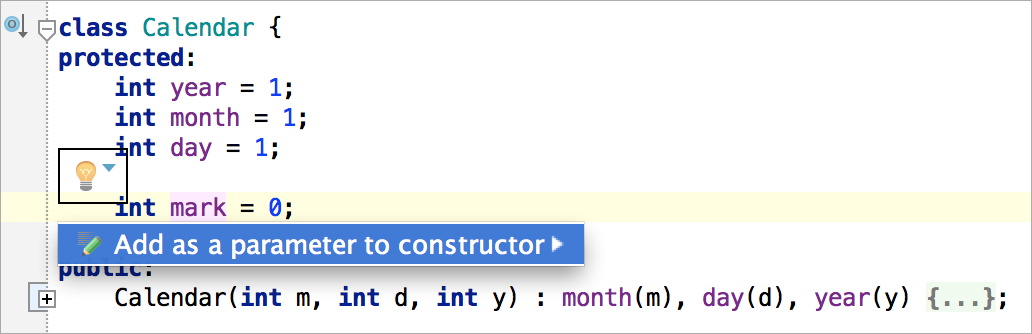
Clion 代码提示和补充的匹配有 All, None, First Letter 三种方式，All 需要关键词完全一致才会提示，First Letter 只有关键词第一个字母（区分大小写）或下标匹配时才会提示，而 None 则是匹配要求最为宽松的一种，不区分大小写、不要求首字匹配，我个人比较喜欢这种，懒人福音~~



*设置匹配模式*

* 智能提示

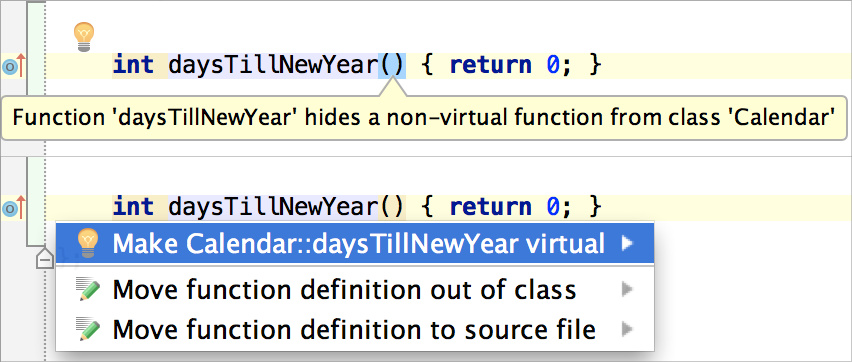
官方给出的名字是 intention actions，这个和 auto-complete 不同，会根据你写的代码给出即时提示，比如调用某个函数漏写了一个参数。厉害的是不仅提示这些问题或建议，同时还提供解决方案，点击左侧的小灯泡或者按 Alt + Enter（快捷键参考自己的配置）会显示这些提示确认后即可应用。



*intention actions*

* 快速修复

这个功能作用和智能提示有点类似，相比之下区别在于快速修复专注于一些容易出现的小错误，比如忘记;, 判断中用=等很容易出现的小错误，Clion 也会显示小灯泡引起你的注意，按 ⌥ + ⏎ 就可以快速修复。



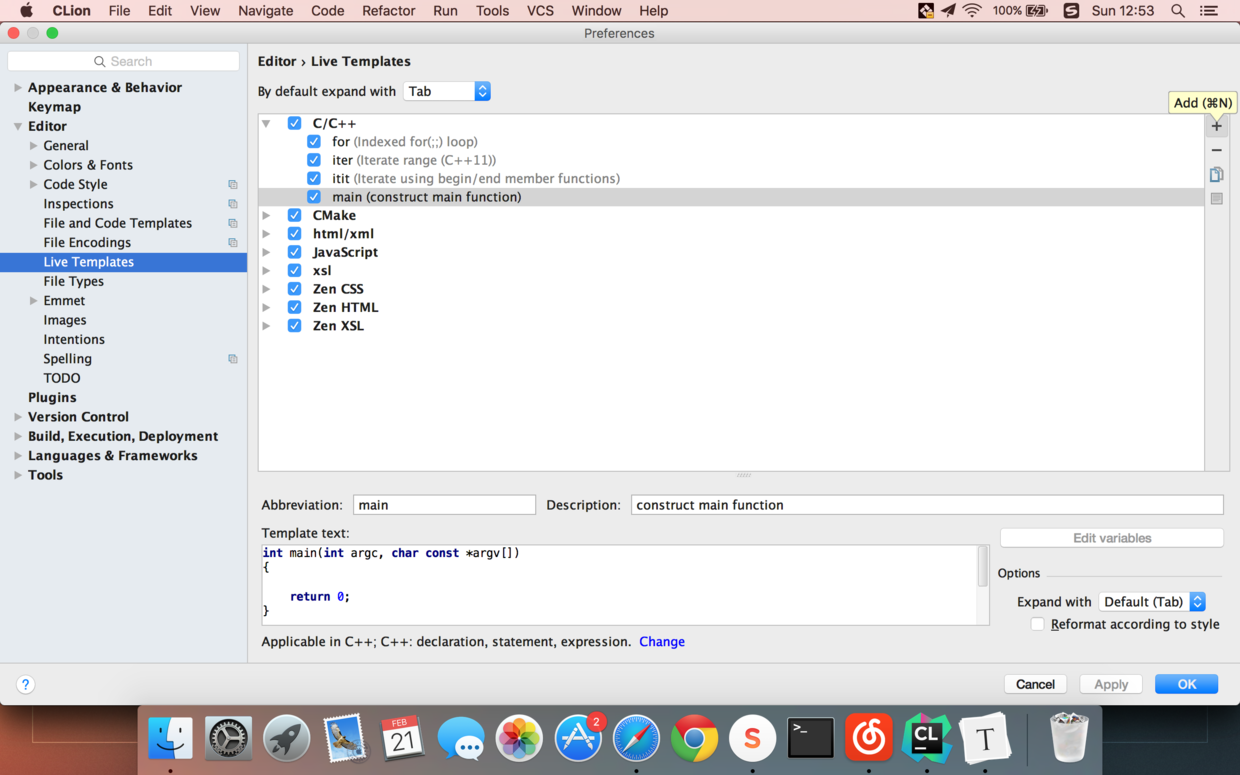
*快速修复*

* 快速注释

和 Sublime Text 一样，在选中代码后按 ⌘ + / 会以行形式注释选中的代码行，按 ⌘ +⇧ + / 则会以代码块的形式完成注释，是敲代码时经常用到的功能之一。

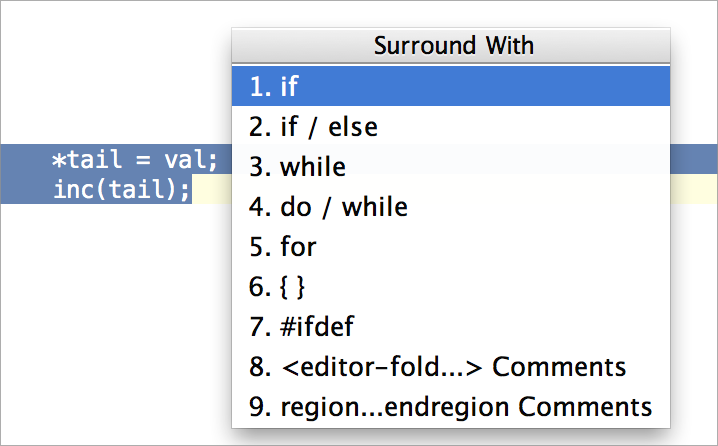
* 代码生成

这个功能会在特定情境下帮你自动生成一些代码，比如类的构造函数和析构函数等，在需要生成的地方按⌘ + N 即可。另外你还可以⌘ + J 插入完整的代码块，默认只有 for, iter, itit，也可以自己在Setting—>Live Templates 里添加自定义的代码块，比如我创建了一个main()函数的代码块，输入 main就会在提示框里出现我们添加的代码块，这个真的是很方便。



*添加 main 函数*

最后还有一个 surround with 的功能，选中任意的代码块按 ⌥ + ⌘ + T 就可以选择给这些代码包裹上 if for 等。



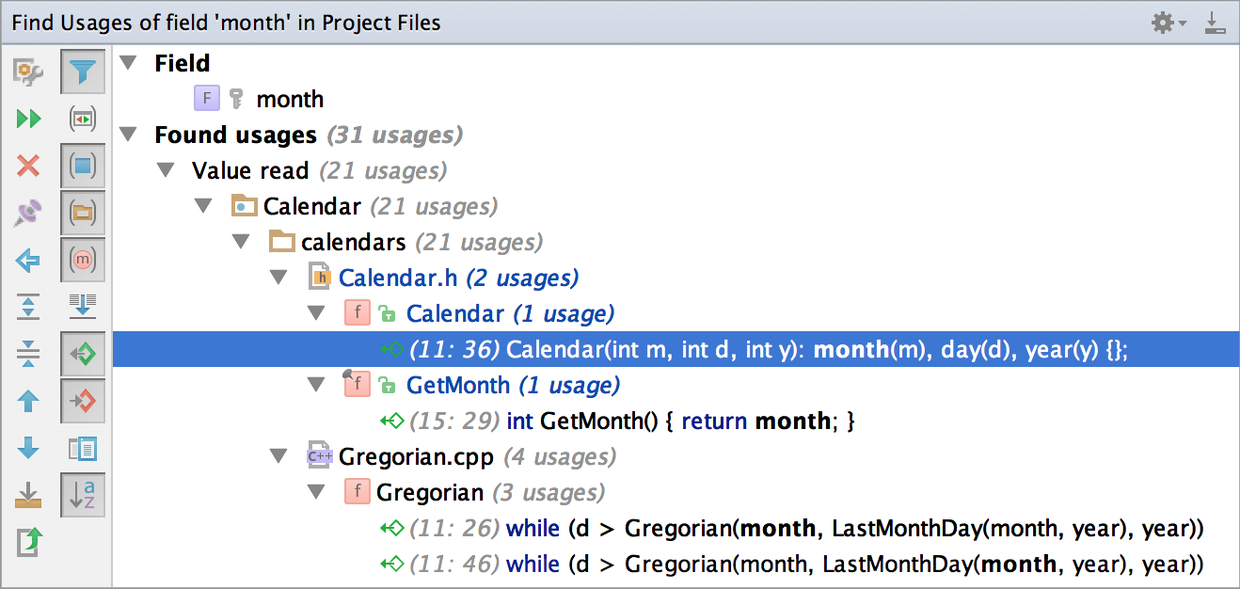
*surround with*

* 巧用搜索功能

搜索又是一个非常实用的功能， Clion 的搜索功能非常强大。

**基本搜索**

对于基本的搜索（也是最常用到的）， ⌘ + F 相信大家都已经非常熟悉了，⇧ + ⌘ + F 则可以在整个工程中进行搜索；按 ⌥ + F7 可以找到某个类、变量、函数等在文中被使用到的的地方；

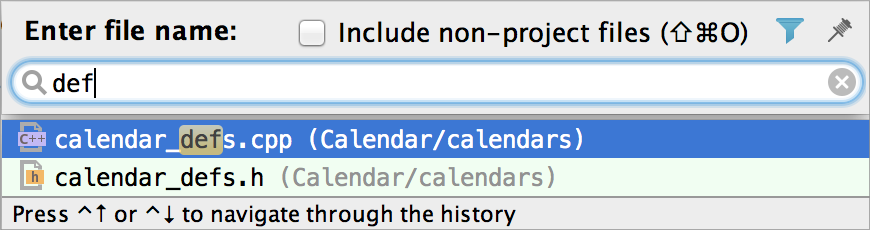


*find usages*

**项目导航**

对于一个大的项目，很多时候需要理清它的层次结构，Clion 为我们提供了项目内文件、类等的导航搜索：

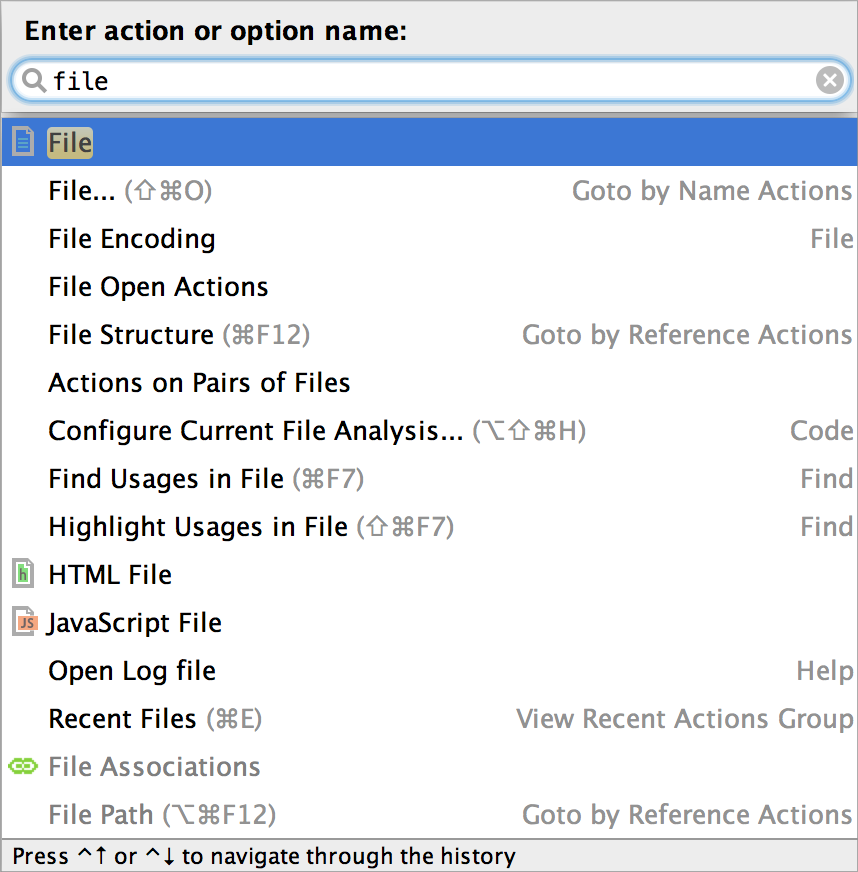
⌘ + O 通过类的名字进行搜索；  
⇧ + ⌘ + O 通过文件名进行搜索；  
⌥ + ⌘ + O 通过符号名进行搜索；  
⌘ + B 搜索声明；  
⌘ + U 基类/函数搜索；



*文件搜索*

**功能搜索**

这个真的是厉害啊，按 ⇧ + ⌘ + A 然后输入你想用到的功能，然后选中就可以了，尤其是当快捷键太多记不住的时候真的是很实用啊。



*功能搜索*

**全局搜索**

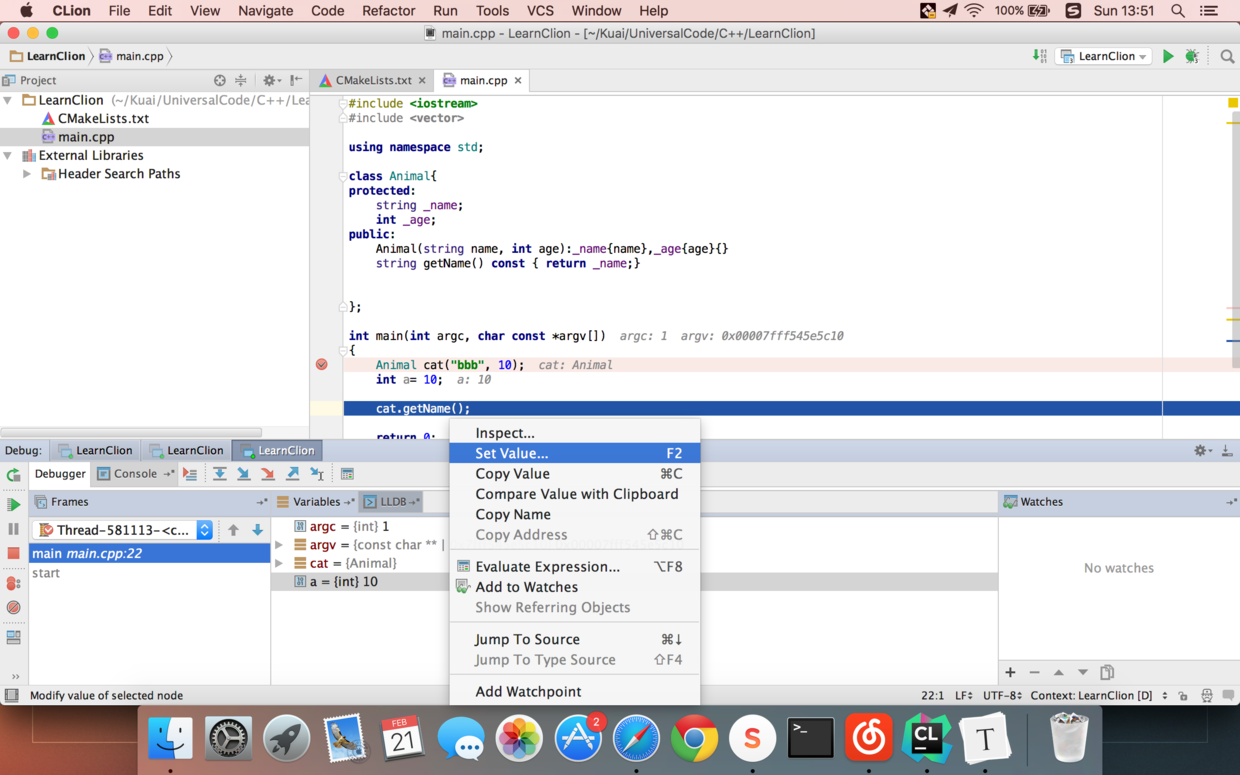
当你不知道你要搜索的东西是属于上面哪一类的时候，就试试 Search Everywhere，只需要连按两下 ⇧，就会呼出 Search Everywhere 的搜索框。

* 调试运行

说实话现在很多编辑器在代码编辑上的功能和体验都不输 IDE 了，但是在项目调试和运行上 IDE 还是有着不小的优势，这也是很多人在大的项目里选择 IDE 而不是代码编辑器的原因。

首先你可以在 Run | Edit Configurations 里面配置调试运行的选项，如果你希望在编译之前或者运行之后运行一段脚本，你可以自己添加这些额外的配置。

在 OS X 上，Clion 支持 GDB 和 LLDB ，建议使用 LLDB，调试的过程和很多 IDE 类似，你可以创建断点，按右上方的小虫或者⌥ + ⌘ + U 进行调试，你可以逐行、逐语句进行调试，Clion 很人性化的一点是调试时变量的值会直接以斜体显示在右侧，而且 Clion 支持调试过程中更改变量的值（在调试输出框内选择相应变量右击 set value 或者按 F2）。



*修改变量*

* 最后

以上提到的一些技巧针对最基本的 Clion 使用，限于篇幅 Clion 还有很多的功能这里都没有提到，那些更为高阶的技巧需要你持续使用之后去发现，一开始说太多反而会让大家抓不住重点，如果你之前有使用过 JetBrains 家其他的产品，那么我相信你很快就能上手 Clion，如果你是第一次使用他家的产品，那么这篇文章可能会对你有一点用处。

如果你需要全面了解 Clion 的功能，可以访问 JetBrains 的[官方指南](https://www.jetbrains.com/clion/help/meet-clion.html)，本文也有部分内容参考了这个指南。

来自：[fangs.in](http://fangs.in/post/cc/awesomeclion)