# TODO

<https://juejin.im/post/5df8a5a5e51d4557f0460990>

# IDE

[http://idea.lanyus.com/](javascript:%20void%200)

CLion是Jetbrains公司旗下新推出的一款专为开发C/C++所设计的跨平台IDE，它是以IntelliJ为基础设计的，同时还包含了许多智能功能来提高开发人员的生产力。

具体的各种详细的功能请自行翻阅资料了解，此处我只简单介绍下CLion在windows的安装使用。

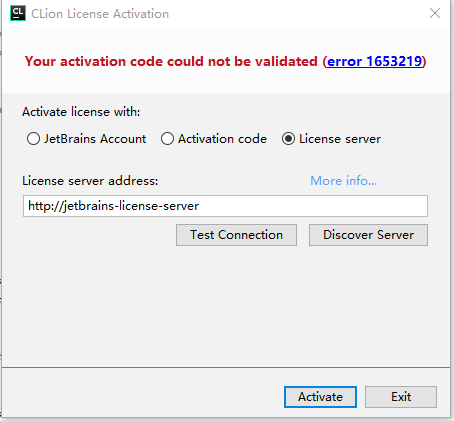
## CLion安装

下载CLion， https://www.jetbrains.com/clion/download/

激活

### 2020.1.1XXXX

选择最后一种License server激活（联网）方式，地址填入：http://jetbrains-license-server （应该会自动填上），完成激活。



### 191226XXX

<https://williamlfang.github.io/post/2019-11-27-clion-%E6%BF%80%E6%B4%BB%E7%A0%81-%E8%BD%AC/>

812LFWMRSH--ti4tUsQISyJF/zfWxSHCr+IcYrX2w24JO5bUZCPIGKSi+IrgQ0RT2uum9n96o+Eob9Z1iQ9nUZ6FJdpEW5g0Exe6sw8fLrWMoLFhtCIvVgQxEEt+M7Z2xD0esmjP1kPKXZyc/i+NCxA2EO2Sec9uifqklBGP1L3xoENAw2QsIWBfttIe6EPWhbS8TIMMr2vF/S3HrN8To5Hj5lwD/t1GHgFK1uWrhsuifAiKcVzqogybzGiR1h2+yNYTMbKxP7uPCcdYMsIyrBNVRGA3IuEJgyGQTQlFbnVQoVUTGPW2tQxprmC464wMjKi40JHh27WzjOHPwgzxDaigwn4Z0EbSpA==-

### 不能激活

破解<https://blog.csdn.net/l297969586/article/details/78183671>

19.8.30的步骤2既可

191109是可以用的（直接使用）

191210是可以用的（用的最后一行），以下是记录

812LFWMRSH--ti4tUsQISyJF/zfWxSHCr+IcYrX2w24JO5bUZCPIGKSi+IrgQ0RT2uum9n96o+Eob9Z1iQ9nUZ6FJdpEW5g0Exe6sw8fLrWMoLFhtCIvVgQxEEt+M7Z2xD0esmjP1kPKXZyc/i+NCxA2EO2Sec9uifqklBGP1L3xoENAw2QsIWBfttIe6EPWhbS8TIMMr2vF/S3HrN8To5Hj5lwD/t1GHgFK1uWrhsuifAiKcVzqogybzGiR1h2+yNYTMbKxP7uPCcdYMsIyrBNVRGA3IuEJgyGQTQlFbnVQoVUTGPW2tQxprmC464wMjKi40JHh27WzjOHPwgzxDaigwn4Z0EbSpA==-

### MinGW 安装

**MinGW**（**Min**imalist **G**NU for **W**indows），又称**mingw32。**是将GCC编译器和GNU Binutils移植到Win32平台下的产物，包括一系列头文件（Win32API）、库和可执行文件。

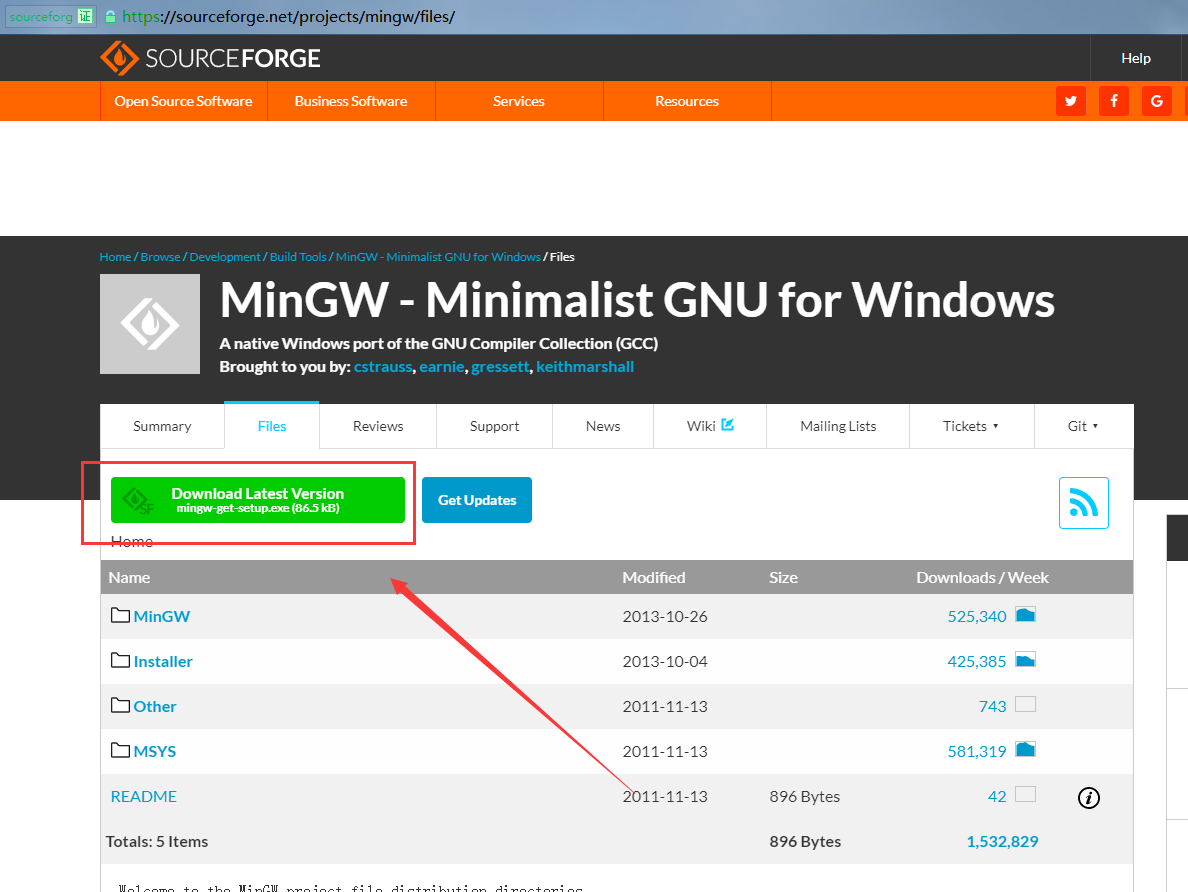
另有可用于产生32位及64位Windows可执行文件的MinGW-w64项目，是从原本MinGW产生的分支[1]。如今已经独立发展[2]。

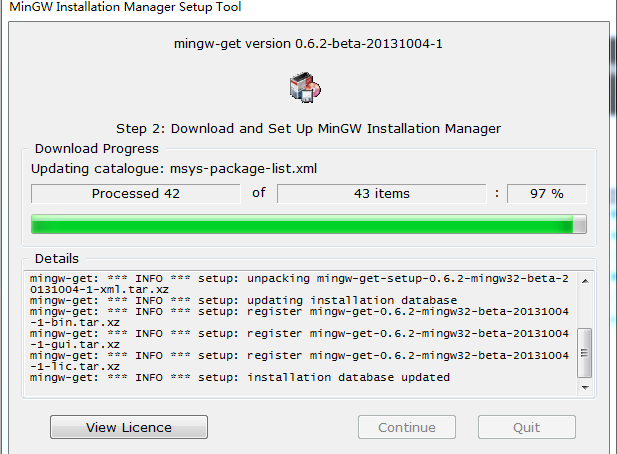
GCC支持的语言大多在MinGW也受支持，其中涵盖[C](https://zh.wikipedia.org/wiki/C%E8%AA%9E%E8%A8%80)、[C++](https://zh.wikipedia.org/wiki/C%2B%2B)、[Objective-C](https://zh.wikipedia.org/wiki/Objective-C)、[Fortran](https://zh.wikipedia.org/wiki/Fortran)及[Ada](https://zh.wikipedia.org/wiki/Ada)

Clion 下载安装完之后并不能直接运行C++项目，需要配置一个环境

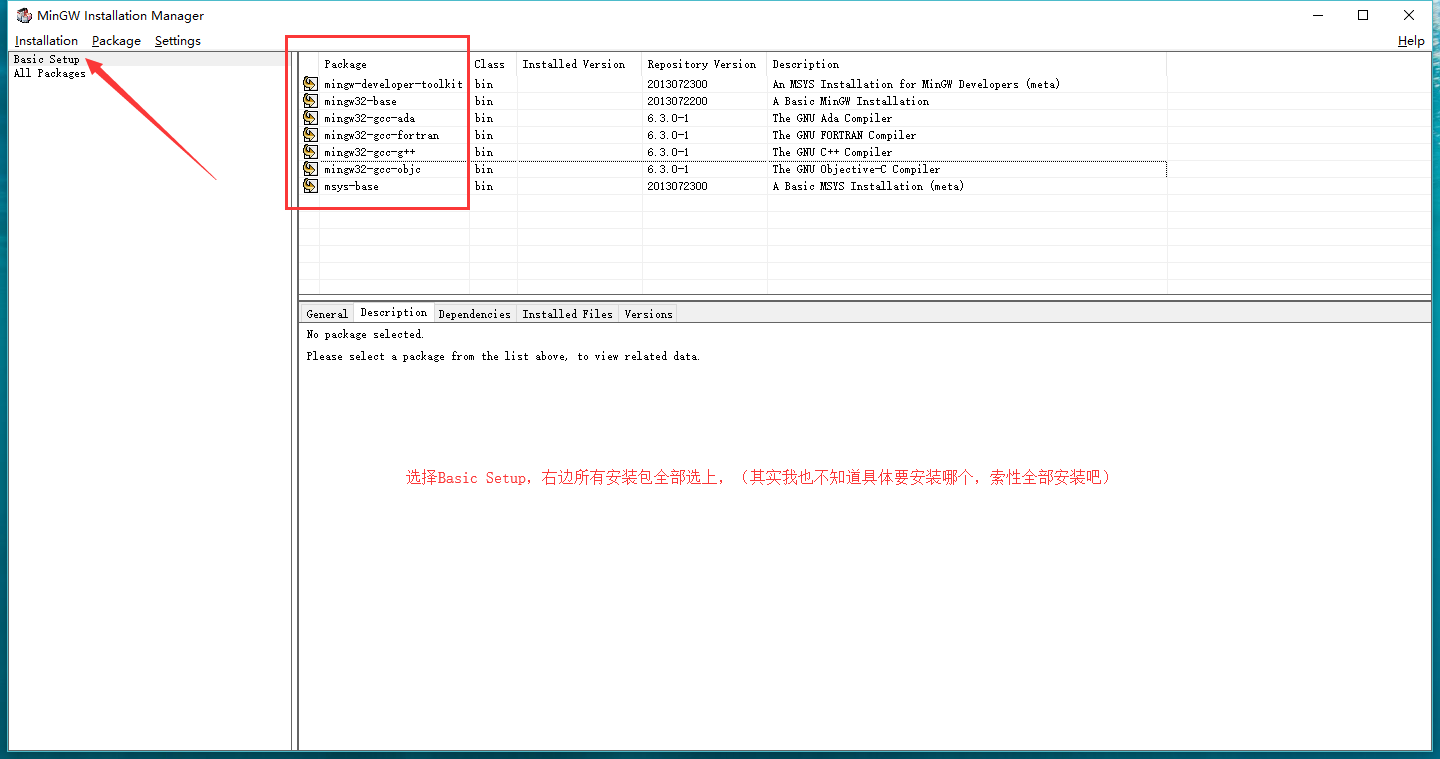
链接：http://sourceforge.net/projects/mingw/files/

#### 下载安装

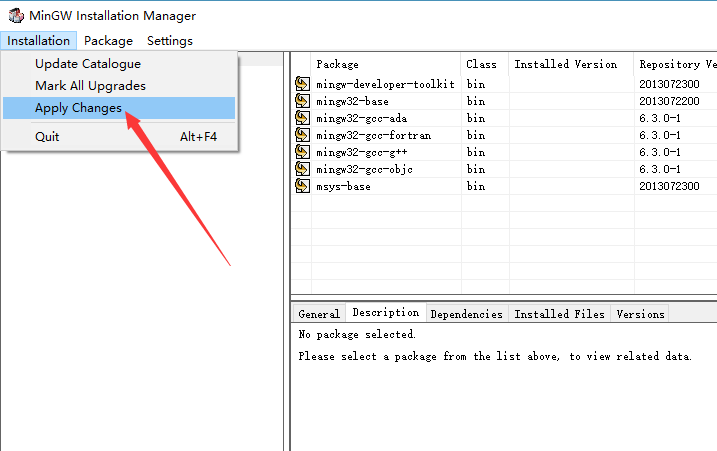




#### 安装其他工具



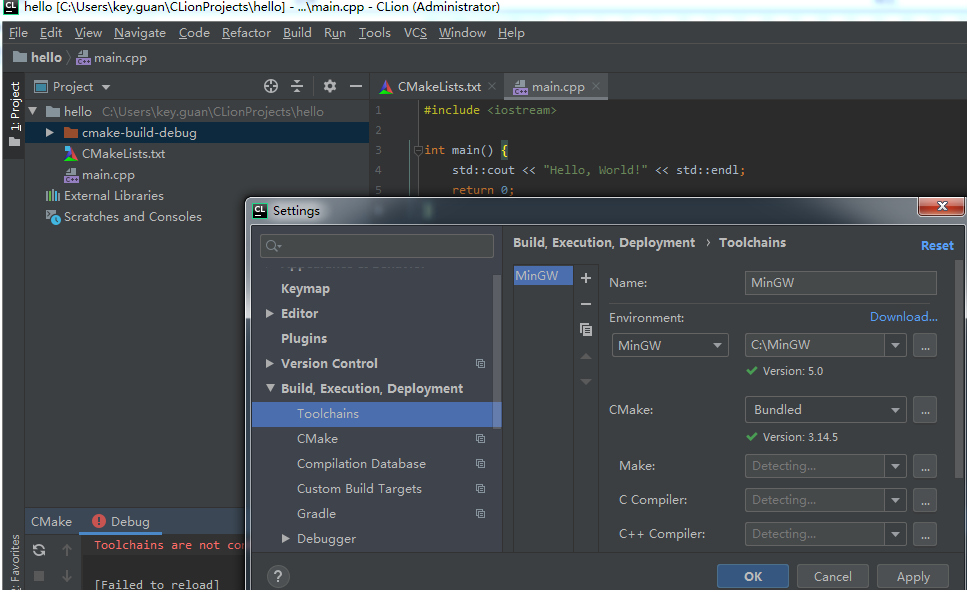
更新配置



开始配置子工具链

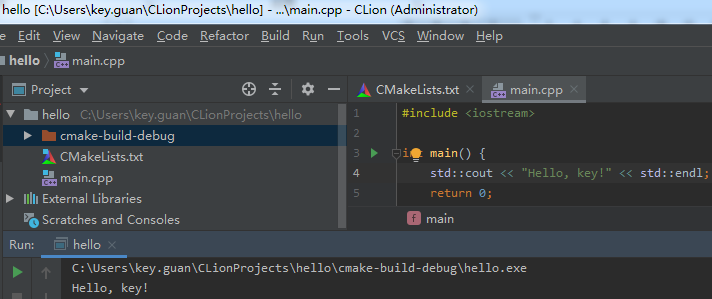
### 让clion识别MinGW

打开Clion，会有大概如下的界面，由于是刚下载好的环境，会自动识别你安装的东西



如果是先创建了工程，也会自动识别的

默认已生成了main.c，但是我们并不能运行，因为我们还缺少编译环境，点击Setting打开设置，搜索Toolchains，我们发现clion为我们提供的环境有这么几种MinGW，CygWin，Visual studio等，我此处选择的是MinGw，点击右侧download，下载后运行安装即可，然后在Enviroment右侧选择mingw-w64的位置即可，如下图：



## 开发环境

### IntelliJ换行CRLF, LF, CR的解释和默认设置

CR是MAC老版本的做法，就是回车，但是后来的MAC系统统一换成LF了，

LF是Linux下的做法，就是换行，这个做法比较自然，为什么要回车换行呢，是吧。

微软采用CRLF，看上去好像是兼容了CR和LF，但是实际完全不是那么回事，就是回车并换行，好鸡肋啊，微软一直保持这种做法。

在window下开发换行默认是CRLF，也就是回车换行，但是Linux下只有换行LF，这样代码提交后，会出现编译问题，所以最好的办法是在IntelliJ下设置默认为LF（目前的编译器clion默认就是LF的？？）。如果不是，请修改

开发人员大多在Linux下，所以对于开发人员来说还是比较坑的。下面介绍设置详解：

第一步：File->Settings… Editor->Code Style

默认是System-Dependent，根据系统自动配置，但是你是windows系统，默认是CRLF，服务器是Linux

参考：<https://blog.csdn.net/ECHO_FOLLOW_HEART/article/details/48314523>

## 配置加速

玩意儿是JAVA写的，应该是内存设置的问题

clion64.exe.vmoptions

-Xss8m

-Xms1024m

-Xmx8192m

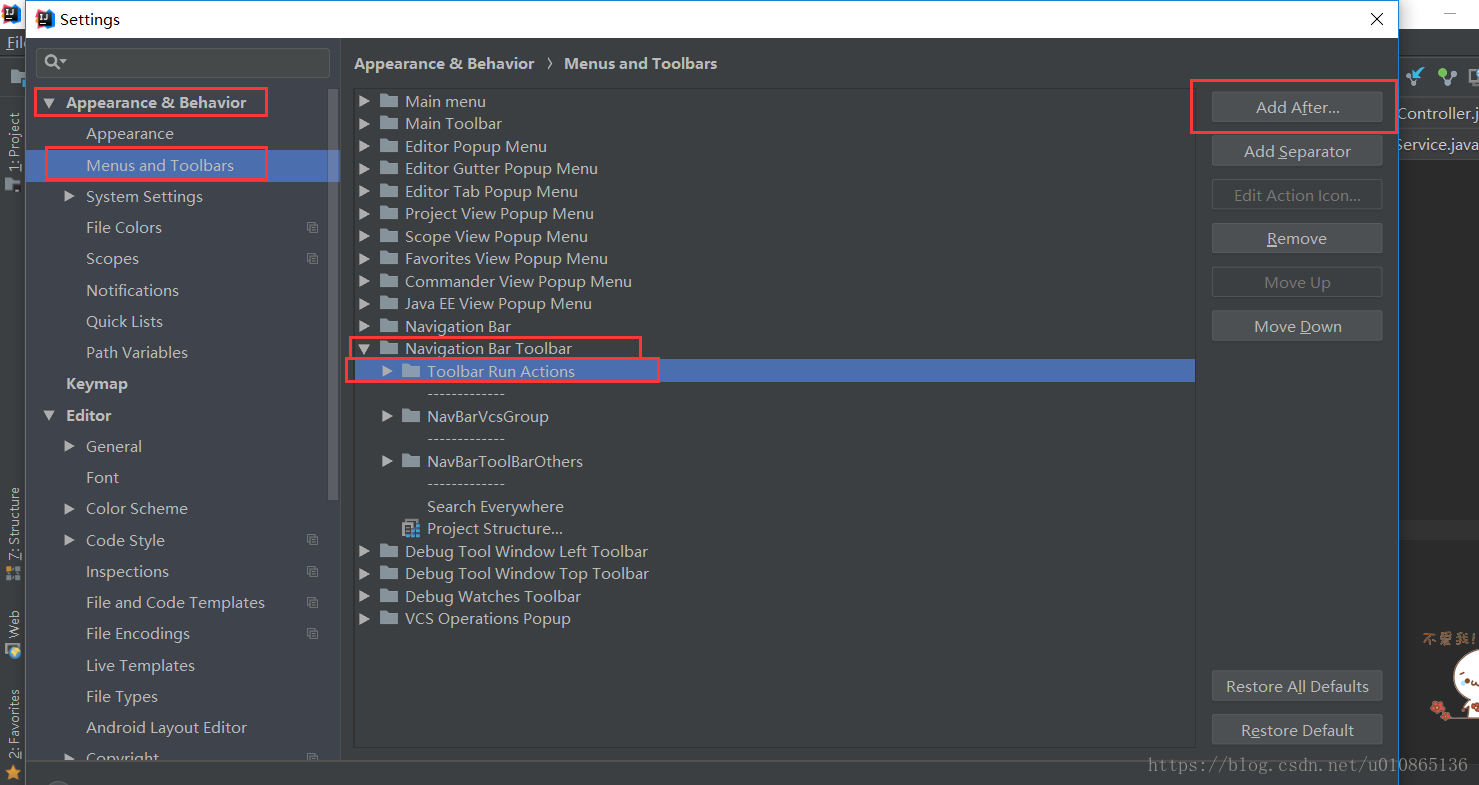
-XX:NewSize=128m

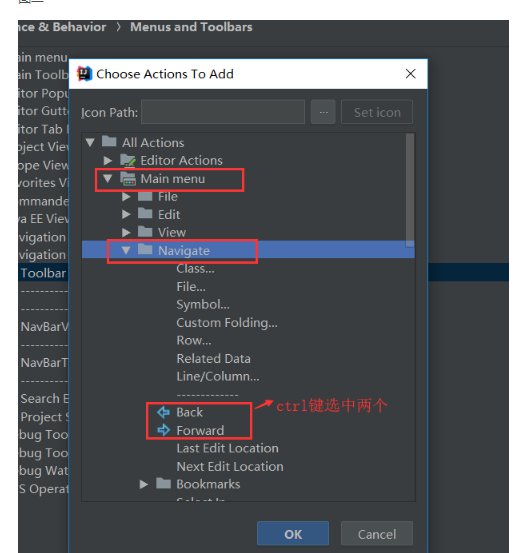
-XX:MaxNewSize=128m

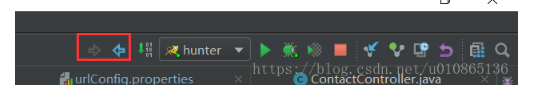
# 实用功能

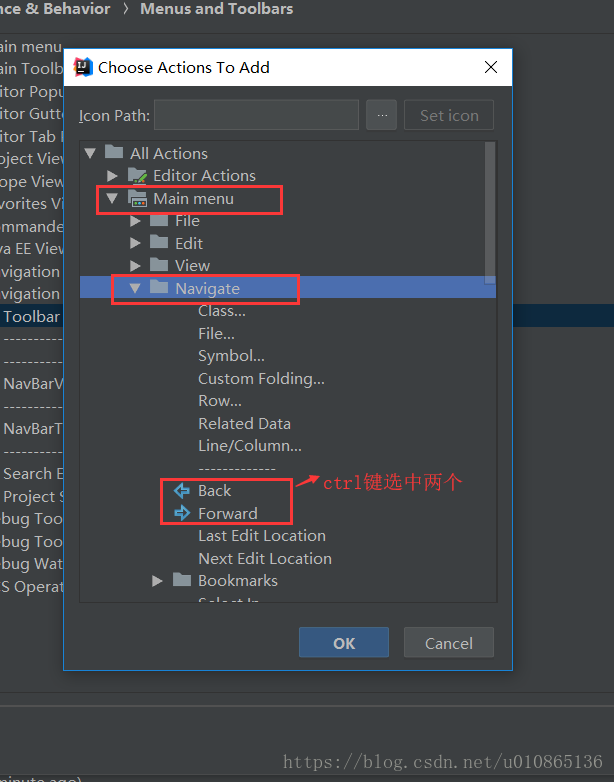
## 前进 后退 添加到工具栏

<https://blog.csdn.net/u010865136/article/details/80394677>









## 同一个文件创建两个编辑器窗格

垂直拆分，两个编辑器窗格彼此相邻