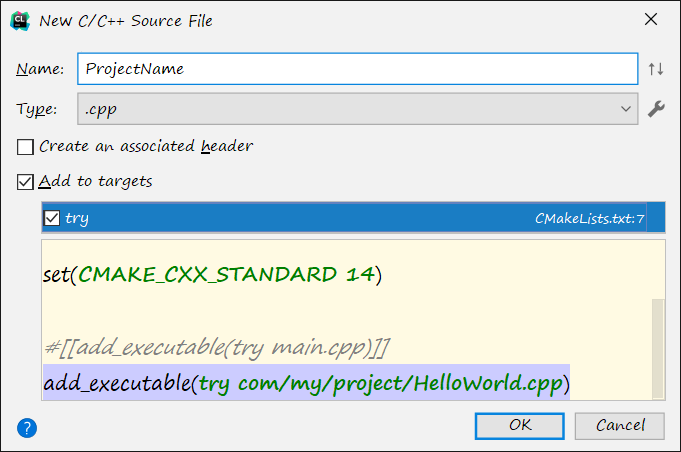
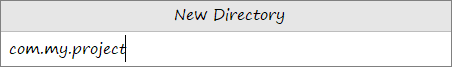
**C++**

1.1 新建文件并运行

一定要勾选Add to targets，这样能自动在CMakeLists.txt的add\_executable(**try main.cpp com/my/project/ProjectName.cpp**)

自动生成路径（自行删除 **main.cpp**）



Extr：新建Directory不能以.分隔，呈现效果如下（老老实实一个个建吧）

1.2 可嵌套注释及测试版代码可用方法【待后期使用】

/\*...\*/不可嵌套使用  
#if 0 ... #endif 属于条件编译，0为参数；用作注释可嵌套  
具体（待使用，执行测试代码，发布后用#if 0 屏蔽）：  
#if 0 //改为#if 1 ,执行code  
 code  
#endif

//condition条件为true执行code1，否则执行code2  
#if condition  
 code1  
#else  
 code2  
#endif

1.3 关键字

新：asm、operator、bool、explicit、extern、tupedef、register、typeid、reinterpret\_cast、typename、union、const\_cast、

friend、unsigned、signed、using、sizeof、virtual、inline、delete、static\_cast、volatile、struct、wchar\_t、mutable、dynamic\_cast、

namespace、template

Java中就是关键字：else、new、this、enmu、throw、private、break、protected、try、case、public、catch、char、float、class、

for、return、goto、continue、if、default、static、void、int、do、long、double、switch、while

在Java中竟然不是关键字：true、export、false、short

在Java中是关键字但是忘了：const

莫名感觉是关键字的：auto

1.4 三字符组

??= # ??/ \ ??' ^ ??( [  
??) ] ??! | ??< { ??> }  
??- ~

2.1 HelloWorld

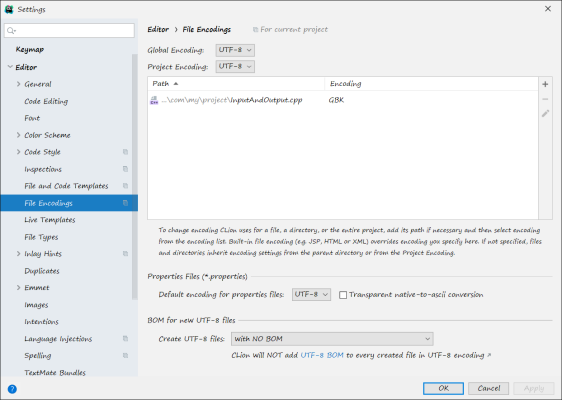
*//头文件<iostream>*#include **<iostream>***//使用std命名空间【其他具体命名空间 & 是否能自己设置新的命名空间？】***using namespace** std;  
  
**int** main(){  
 *//输出语句：cout <<* cout << **"Hello,World!"**;  
 *//终止main()函数并向调用进程返回0* **return** 0;  
}

2.2基本输入输出

#include **<iostream>  
using namespace** std;  
  
**int** main(){  
 **int** number;  
 cout << **"输入一个整数："**;  
 */\*Scanner sc = new Scanner(System.in);  
 \*int number = sc.nextInt();\*/* cin >> number; *//输入* cout << **"输入的数字为："** << number;  
 **return** 0;  
}

Extr：

中文语句输出乱码，具体调试如下页面



将右下角UTF-8改为GBK

2.3换行\n和endl

#include **<iostream>  
using namespace** std;  
*//换行之\n*  
**int** main(){  
 cout << **"This is a sentence without line-feed."**;  
 cout << **"This \n"**;  
 cout << **"is \n"**;  
 cout << **"another \n"**;  
 cout << **"sentence. \n"**;

**return** 0; *//别忘了！！！！！！*  
}

*//换行之endl*#include **<iostream>  
using namespace** std;  
**int** main(){  
 cout << **"This is a sentence without line-feed."**;  
 cout << **"This "** << endl;  
 cout << **"is"** << endl;  
 cout << **"another"** << endl;  
 cout << **"sentence."** << endl;

**return** 0; *//别忘了！！！！！！*  
}

3.1手动输入两个数并返回两数之和

#include **<iostream>  
using namespace** std;  
**int** main(){  
 **int** num1,num2,sum;  
 cout << **"请输入两个整数，并以Enter隔开："**;  
 cin >> num1 >> num2;  
 sum=num1+num2;  
 cout << num1 << **" + "** << num2 << **" = "** << sum;

**return** 0;  
}

3.2.1 各种数据&变量类型

#include **<iostream>***//使用string需调用？*#include **<string>  
using namespace** std;  
**int** main(){  
 **int** a = 1; *//unsigned int 、 signed int 、 short int 、 unsigned short int 、 signed short int 、 long int 、 signed long int 、 unsigned long int* **float** b = 2.22;  
 **double** c = 3.33; *//long double* **char** d = **'ABCD'**; *//unsigned char 、 signed char* **bool** e = **'true'**;  
 string f = **"Let me test 日本語."**;  
 size\_t g = 23;  
 cout << **"int:"** << a << endl;  
 cout << **"float:"** << b << endl;  
 cout << **"double:"** << c << endl;  
 cout << **"char:"** << d << endl;  
 cout << **"bool:"** << e << endl;  
 cout << **"string:"** << f << endl;  
 cout << **"size\_t:"** << g<< endl;  
 **return** 0;  
}

Extr：运行报错，搜索相关问题和GBK&UTF\_8相关；后期再看看？

#include **<iostream>**#include **<locale>**#include **<cstdlib>  
using namespace** std;  
locale loc(**"chs"**);*//windows下ok  
//这段貌似在ubuntu下ok  
//locale loc("zh\_CN.UTF-8");  
//而且还需要在ubuntu的终端中执行：  
//sudo locale-gen***int** main()  
{  
 **wchar\_t** wStr[]=**L"这是一段中文"**;  
 wcout.imbue(loc);  
 wcout<<wStr<<endl;  
 getchar();  
 getchar();  
 **return** 0;  
}

Extr：typedef为已有类型取新名字，有机会看看实例，有点懵？

#include**<iostream>  
using namespace** std;  
**typedef int** (\*A)(**char**,**char**);  
**int** fun0(**char** a,**char** b);  
**int** fun1(**char** b,**char** a);  
**int** main()  
{  
 A a;  
 A b;  
 a = fun0;  
 a(**'a'**,**'b'**);  
 b = fun1;  
 b(**'b'**,**'a'**);  
 **return** 0;  
}  
**int** fun0(**char** a,**char** b)  
{  
 cout<<**"fun0"**<<endl;  
 **return** 0;  
}  
**int** fun1(**char** b,**char** a)  
{  
 cout<<**"fun1"**<<endl;  
 **return** 0;  
}

Extr：枚举类型enum，由用户定义的若干枚举常量的集合#include **<iostream>**;  
**using namespace** std;  
**int** main(){  
 **enum** songs {***Devil*** = 5,***Trigger*** = 4} a,b;  
 a = ***Devil***;  
 b = ***Trigger***;  
 cout << a << b;  
 **return** 0;  
}

3.2.2各种数据&变量类型

#include **<iostream>  
using namespace** std;  
*//全局变量声明***extern int** a,b,c;  
**extern float** f;  
**int** func(); *//函数声明***int** main(){  
 **int** a = 5,b = 6; *//全局变量定义并赋值* **int** c = a+b; *//全局变量定义* **float** f = 6.0/2.5; *//全局变量定义并赋值* **int** i = func(); *//函数调用* **int** d = 2; *//局部变量定义并赋值* cout << c<< endl << f <<endl << d;  
 **return** 0;  
}  
**int** func(){ *//函数定义* cout << **"nothing"** << endl;  
 **return** 0;  
}

Extr:左值lvalue、右值rvalue，后期数据结构？再说