|  |  |
| --- | --- |
| 1. 枚举类型 2. 枚举成员 3. 枚举也定义新的类型 4. 指定num的大小 5. 枚举类型的前置声明 6. 形参匹配与枚举类型 | 枚举类型，枚举成员  限定作用域的枚举类型  不限定作用域的枚举类型  Enum name{}// 不限定  Enum class num{}// 限定  Enum name:unsigned long long |
| 1. 枚举类型   将一组常量组织在一起，每个枚举类型定义了一种新的类型，属于字面值常量类型  Enum class name 为限定作用域，enum name 为不限定   1. 枚举成员 2. 限定作用域（带class）中在枚举类型的作用域外不可访问(就是在枚举成员的作用域在枚举定义的范围内才能使用)。   不限定作用域（不带class），与枚举类型本身的作用域相同，可在枚举外使用  2) 限定枚举成员可以隐藏不限定,限定枚举成员隐藏同名的不限定枚举成员（在限定枚举内才会隐藏）。  3) 可以 枚举name::枚举成员 显示调用  4） 枚举成员值默认从0开始，也可以显示指定16 什么的。如果不指定，为前面一个成员值+1   1. 是const的，使用常量表达式的地方用枚举成员，如constexpr表达式中 2. 可以作为switch，作为一个非类型模板形参使用 3. 枚举也定义新的类型 4. 想初始化enum对象或为enum对象赋值，必须使用enum中的一个枚举成员或者该类型的另一个对象 5. 不限定作用域的枚举成员或枚举类型的对象自动转换成整形 6. 指定enum大小 7. Enum名字加冒号指定类型 8. 限定作用下默认int，不限定没有默认，为潜在类型能容纳枚举成员最大的数值 9. 为限定和不限定指定某个类型，枚举成员不能超过范围，若超过不会自动提升，会报错。 10. 潜在类型能帮助在不同的环境下执行一样 11. 枚举类型的前置声明 12. 不限定需要显示指定类型，限定不用默认int 13. Enum的声明与定义要匹配，不能同一个作用域声明同名不限定与限定名字。并且声明与定义的指定类型要一致 14. 形参匹配与枚举类型 15. 不能直接将整形值作为enum形参，可将一个不限定enum对象或它的一个枚举成员给整形形参 16. 不限定Enum的类型提升int或更大整形，由潜在类型决定 17. 不限定的声明需要指定类型(因为没有默认类型)，但是若声明与定义一起不需要指定类型，由枚举成员中最大的值指定，但实际的对象与成员会提升为int，不会提升为unsigned char   Note:  /\*  1.限定作用默认int，不限定没有默认类型，但是潜在认为能容纳枚举成员最大的  2.为限定和不限定指定某个类型，枚举成员不能超过范围，若超过不会自动提升，会报错  3.限定作用域，必须显示指定成员，因为枚举成员的范围在枚举内  \*/ | |