|  |  |
| --- | --- |
| 1. 行为像值的类 2. 类值拷贝赋值运算符 | HasPtr(const HasPtr &p):  Ps(new std::string(\*p.ps)),i(p.i){}  // 错误，应该先  Auto newp = rhs;  HasPtr&  HasPtr::operator=(const HasPtr &rhs)  {delete ps;  Ps = new string(\*(rhs.ps));  I=rhs.i;  Return \*this;} |
| 1. 行为像值的类 2. 每个对象都应该拥有一份自己的拷贝   如：类中有一个string的指针ps，  在拷贝构造函数需要完成string的拷贝，而不是拷贝指针  在析构函数来释放string  在拷贝赋值运算符来释放当前的string，并从右侧运算对象拷贝string   1. 类值拷贝赋值运算符 2. 组合了析构函数和构造函数的操作   关键概念：赋值运算符   1. 如果将一个对象赋予它的自身，赋值运算符必须能正确工作 2. 大多数赋值运算符组合了析构函数和拷贝构造函数的工作   好的模式写法是：先将右侧运算对象拷贝到一个局部临时对象中。再销毁左侧运算符对象  Note:特别是对象赋予它的自身，如果先销毁左侧对象，那么右侧对象也跟着销毁，所以一定先要保存右侧运算对象 | |