|  |
| --- |
| **DBConnection类修改** |
| class DBConnections {  private:  static std::vector<DBConnections\*> instances; // 存储三个实例  static const int MAX\_CONNECTIONS = 3; // 最大实例数量  static int nextInstance; // 跟踪下一个实例  // 私有构造函数  DBConnections() {  std::cout << "Creating DBConnection instance\n";  }  public:  // 获取一个实例  static DBConnections\* getInstance() {  if (instances.size() < MAX\_CONNECTIONS) {  // 如果还没有达到最大实例数，创建新的实例  DBConnections\* instance = new DBConnections();  instances.push\_back(instance);  return instance;  } else {  // 如果已经有三个实例，循环返回已有的实例  DBConnections\* instance = instances[nextInstance];  nextInstance = (nextInstance + 1) % MAX\_CONNECTIONS;  return instance;  }  }  // 打印连接信息（示例方法）  void ConnectionInfo() {  std::cout << "DB Connection Info\n";  }  // 析构函数  ~DBConnections() {  std::cout << "Destroying DBConnection instance\n";  }  }; |