DocumentManager 管理存档和配置信息在内存与硬盘中交互

**数据成员**

**static DocumentManager\* instance\_**

 **作用**：单例模式的实例指针，确保只有一个 DocumentManager 实例存在。

 **被调用时机**：通过 getInstance() 方法访问。

**int current\_archive\_**

 **作用**：当前打开的存档编号。

 **被调用时机**：在创建、加载、保存和删除存档时使用。

**std::unordered\_map<std::string, rapidjson::Document\*> data\_**

 **作用**：存储已加载的 JSON 文档，键为文档名称，值为对应的 Document 指针。

 **被调用时机**：在获取、释放文档时使用。

**std::unordered\_map<std::string, std::string> name\_map\_**

 **作用**：存储文件名与文件路径的映射。

 **被调用时机**：在获取文件路径时使用。

 **可能被调用的地方**：getPath、initNameMap。

**函数成员**

**私有成员**

**DocumentManager()**

 **作用**：构造函数，初始化 DocumentManager，加载用户配置文档。

 **被调用时机**：在 instance\_ 被创建时。

 **可能调用的地方**：getInstance()。

**~DocumentManager()**

 **作用**：析构函数，释放所有加载的文档和保存的配置。

 **被调用时机**：当 DocumentManager 实例被销毁时。

 **可能调用的地方**：程序结束或 DocumentManager 被替换时。

**公有成员 接口**

**static DocumentManager\* getInstance()**

 **作用**：获取 DocumentManager 的单例实例。

 **被调用时机**：任何需要访问 DocumentManager 的地方。

 **可能调用的地方**：游戏的任何部分需要管理文档时。

**std::string getPath(const std::string& name)**

 **作用**：根据名称获取文件路径。

 **被调用时机**：当需要文件路径时。

 **可能调用的地方**：其他类或函数需要文件路径时。

**bool hasDocument(const std::string& name) const**

 **作用**：检查是否已加载指定文档。

 **被调用时机**：在需要确认文档是否存在时。

 **可能调用的地方**：其他类或函数需要检查文档存在性时。

**rapidjson::Document\* getDocument(const std::string& path)**

 **作用**：获取指定路径的 JSON 文档，如果未加载则加载它。

 **被调用时机**：当需要访问 JSON 文档时。

 **可能调用的地方**：其他类或函数需要 JSON 数据时。

**void freeDocument(const std::string& path)**

 **作用**：释放指定路径的文档。

 **被调用时机**：当不再需要文档时。

 **可能被调用的地方**：~DocumentManager()、freeArchiveDocument()。

**void createConfigDocument()**

 **作用**：创建配置文档并加载所有现有存档的键信息。

 **被调用时机**：在构造函数中，如果未找到用户配置文档时。

 **可能被调用的地方**：DocumentManager()。

**bool createArchiveDocument(const int num)**

 **作用**：创建新的存档文档。

 **被调用时机**：当需要创建新存档时。

 **可能被调用的地方**：游戏存档管理部分。

**bool loadArchiveDocument(const int num)**

 **作用**：加载指定编号的存档文档。

 **被调用时机**：当需要加载存档时。

 **可能被调用的地方**：游戏存档管理部分。

**Document\* getArchiveDocument() const**

 **作用**：获取当前存档文档。

 **被调用时机**：当需要访问当前存档数据时。

 **可能被调用的地方**：游戏逻辑需要存档数据时。

**void freeArchiveDocument()**

 **作用**：释放当前存档文档。

 **被调用时机**：当不再需要当前存档时。

 **可能被调用的地方**：deleteArchive、createArchiveDocument。

**void saveArchiveDocument()**

 **作用**：保存当前存档文档。

 **被调用时机**：在存档数据发生变化时。

 **可能被调用的地方**：createArchiveDocument、deleteArchive。

**bool deleteArchive(const int num)**

 **作用**：删除指定编号的存档文档。

 **被调用时机**：当需要删除存档时。

 **可能被调用的地方**：游戏存档管理部分。

**rapidjson::Document\* getConfigDocument() const**

 **作用**：获取用户配置文档。

 **被调用时机**：当需要访问配置数据时。

 **可能被调用的地方**：游戏逻辑需要配置数据时。

**void saveConfigDocument() const**

 **作用**：保存用户配置文档。

 **被调用时机**：在配置数据发生变化时。

 **可能被调用的地方**：createConfigDocument、deleteArchive。