

Описание файлов:

objects.txt

– Файл с объектами

DrawRule.lua

– файл с правилами рисования объектов

LuaRuleMashine.h / LuaRuleMashine.cpp

– Класс для работы с правилами рисования из lua-файла

Object.h / Object.cpp

– Класс для работы с объектами

ObjectLoader.h / ObjectLoader.cpp

– Класс для загрузки настроек объектов

Описание фалов

objects.txt

В данном файле содержится информация об объектах. Их тип, имя и координаты, количество которых зависит от типа объекта.

DrawRule.lua

В файле содержится информация о правилах рисования объектов. Список `fieldTypes` содержит в себе названия полей правил, которые должны быть определены у каждого объекта. Далее идет описание параметров для каждого объекта, где имена объектов соответствуют именам из *objects.txt*. Если потребуется изменить значения полей или даже изменить их количество и название, достаточно произвести изменения лишь в этом файле.

Описание классов

Класс *LuaRuleMashine*

Открытые члены:

LuaRuleMashine

```
LuaRuleMashine(std::string fileName)
```

Конструктор производит подготовку класса к работе со скриптом `fileName`.

~LuaRuleMashine

```
~LuaRuleMashine()
```

Деструктор выполняет очистку информации о выполняющемся скрипте

getProperty

```
PropertyFields getProperty(const Object& obj)
```

Возвращает [PropertyFields](#) (см. опр. синонима в [ObjectLoader](#)) для переданного объекта `obj`

Закрытые члены класса:

L_

```
lua_State *L
```

Хранит информацию о виртуальном стеке для работы со скриптом Lua.

L_FIELDS_TYPES_

```
const char* L_FIELDS_TYPES = "fieldTypes"
```

Константа, соответствующая имени переменной из Lua-скрипта, которая хранит список полей (см. опр. `fieldTypes` в описании [DrawRule.lua](#))

`L_STACK_TOP_, L_STACK_DOWN_`

```
const int L_STACK_TOP = -1, L_STACK_DOWN = 1
```

Константы для облегчения обращения к элементам виртуального стека.

`field_`

```
std::vector<std::string> field_
```

Аналог `fieldTypes` из [DrawRule.lua](#)

`initVectField_`

```
void initVectField_();
```

Служебная функция для заполнения [field](#)

Класс `ObjectLoader`

Открытые члены:

`ObjectLoader`

```
ObjectLoader(const std::string& fileName)
```

Конструктор производит загрузку объектов из файла `fileName`.

`getObjects`

```
std::vector<Object> getObjects() const
```

Возвращает вектор загруженных из файла объектов.

Закрытые члены класса:

`objects_`

```
std::vector<Object> objects_
```

Вектор, хранящий описание всех загруженных объектов.

Класс `Object`

Открытые члены:

`PropertyFields`

```
typedef std::vector<std::pair<std::string, std::string> > PropertyFields
```

Синоним `PropertyFields`. Объект данного типа содержит информацию о правилах рисования для определенного объекта, представленную массивом пар ключ-значение

`setType`

```
void setType(std::string& objectType)
```

Устанавливает тип в виде строки для данного объекта.

`setName`

```
void setName(std::string& objectName)
```

Устанавливает строковое имя для данного объекта.

`setCoordinates`

```
void setCoordinates(const std::vector<std::pair<double, double>>& objectCoordinates)
```

Устанавливает координаты для данного объекта.

setDrawProperty

```
void setDrawProperty(const PropertyFields& objectDrawProperty)
```

Устанавливает правила рисования [PropertyFields](#) для данного объекта.

getType

```
std::string getType() const
```

Возвращает строковое представление типа данного объекта.

getName

```
std::string getName() const
```

Возвращает строковое представление имени данного объекта.

getCoordinates

```
std::vector<std::pair<double,double>> getCoordinates() const
```

Возвращает координаты данного объекта.

getDrawProperty

```
void setDrawProperty(PropertyFields objectDrawProperty)
```

Возвращает правила рисования [PropertyFields](#) для данного объекта.

Закрытые члены класса:

type_

```
std::string type_
```

Строка, хранящая тип объекта.

name_

```
std::string name_
```

Строка, хранящая наименование объекта.

coordinates_

```
std::vector<std::pair<double,double>> coordinates_
```

Вектор, хранящий координаты объекта.

drawProperty_

```
std::vector<PropertyFields> drawProperty_
```

Хранит правила рисования [PropertyFields](#) для данного объекта.