

《魔兽世界编程宝典》

图书基本信息

书名：《魔兽世界编程宝典》

13位ISBN编号：9787302222392

10位ISBN编号：7302222398

出版时间：2010-6

出版社：清华大学出版社

作者：James Whitehead II, Bryan McLemore, Matthew Orlando

页数：870

译者：杨柏林,张卫星,王聪

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

前言

魔兽世界游戏（WOW）自2004年11月23日发售以来，已迅速地成为有史以来最流行的视频游戏之一。目前它900多万的用户，它已经家喻户晓了。魔兽世界是一个引人入胜的环境，您不需要多少时间上手便可以定制角色、探索新的世界，以及与朋友组团合作。一些玩家晚上花费4~6小时在公会进行组队，试图打败最后、最强大的怪物。另一些玩家则喜欢PVP战斗，将时间花在竞技场（Arena）或是战场（Battleground）上，试图提高他们的声望。还有一些玩家仅仅是享受通过游戏与团队或是朋友在一起的时间。魔兽世界满足了各种玩家的需求，这很可能是它如此成功的主要原因。游戏实现各种娱乐风格的一种方式，便是以插件的方式定制用户界面。对于那些具有技术灵感，或是单纯地追求完美的玩家，暴雪公司将其用户界面的定制完全开放，允许这些玩家改变其所有功能。插件可以是简单地改变生命条的颜色、增加一个显示随机表情的新命令，也可以复杂到提供多功能的网络日历，以允许公会组织和协调事件。除此之外，暴雪继而提供了对用户界面社区的推动和支持，这种方式是其他游戏开发人员所未尝试过的。用户界面社区在过去的几年中迅速地增长，并且没有停止的迹象。编写本书的目的在于为读者提供所需要的工具来为魔兽世界用户界面进行自定义修改，包括对编程语言、术语以及插件结构的介绍。目前，急需编写的插件成千上万个，本书也为您提供了实现它们的方法。

读者定位 本书可以为以下对象提供帮助：希望学习如何修改已有插件的新手插件用户，希望向作品中添加更多令人兴奋的功能的中级插件作者，以及资深插件开发人员——他们可以将本书作为极其复杂的魔兽世界用户界面系统的参考。本书假定，读者对魔兽世界游戏十分熟悉，并且对编程感兴趣。了解任何一门编程语言的读者将会发现书中的很多概念十分熟悉。具有极少、甚至没有编程经验的读者应当首先着重于本书的第1部分，该部分讲授了Lua——一门用于编写插件的编程语言。尽管没有任何编程经验的读者会学习到足够的知识来创建和修改插件，但是他们或许仍然希望（或者需要）从其他的资料中得到更加通用的编程课程。对于具有编程经验的读者，前面的几个章节会非常简单。其中的示例可以使这些读者十分迅速地掌握Lua编程语言的基本语法。若读者已经对高级脚本语言（如Python）非常熟悉，则可以简单地跳过前几章，然后直接进入第11部分，即基本的插件编程。

《魔兽世界编程宝典》

内容概要

《魔兽世界编程宝典:World of Warcraft Addons完全参考手册》可以为以下对象提供帮助：希望学习如何修改已有插件的新手插件用户，希望向作品中添加更多令人兴奋的功能的中级插件作者，以及资深插件开发人员——他们可以将《魔兽世界编程宝典:World of Warcraft Addons完全参考手册》作为极其复杂的魔兽世界用户界面系统的参考。《魔兽世界编程宝典:World of Warcraft Addons完全参考手册》假定，读者对魔兽世界游戏十分熟悉，并且对编程感兴趣。了解任何一门编程语言的读者将会发现书中的很多概念十分熟悉。

作者简介

James Whitehead II创建的插件包括PerfectRaid、Clique、TomTom和其他可以从www.wowinterface.com网站下载的插件。他还是Hacking World of Warcraft一书的作者之一。

书籍目录

目录

第一部分 学习编程

第1章 关于魔兽世界的编程

2

1.1 定制用户界面

2

1.1.1 插件是如何工作的

2

1.1.2 插件能够做什么

3

1.2 Lua新手上路

3

1.2.1 下载和安装WowLua

4

1.2.2 在网上使用Lua

4

1.2.3 下载并安装Lua解释器

5

1.3 探究您的Lua解释器

6

第2章 探讨Lua基础知识

7

2.1 使用Lua解释器

7

2.1.1 运行命令

7

2.1.2 错误信息的理解

8

2.1.3 使用历史信息来做改变

8

2.1.4 退出解释器

9

2.2 处理数字

9

2.2.1 基本算术运算

9

2.2.2 科学记数法

10

2.2.3 十六进制表示法

11

2.2.4 理解浮点

11

2.3 理解值和变量

12

2.3.1 探讨值和它们的类型

12

2.3.2 使用变量

13	
2.4	使用字符串
16	
2.4.1	比较字符串
16	
2.4.2	多个字符串的连接
16	
2.4.3	将数字转换成字符串
17	
2.4.4	将字符串转换成数字
17	
2.4.5	引用字符串
17	
2.4.6	转义特殊的字符
19	
2.4.7	获得字符串的长度
20	
2.5	布尔值和运算符
21	
2.5.1	使用and运算符
21	
2.5.2	or运算符的使用
21	
2.5.3	not运算符的使用
22	
2.5.4	理解nil值
22	
2.6	探讨作用域
23	
2.6.1	块(Block)
23	
2.6.2	组块(Chunk)
24	
2.7	小结
24	
第3章	基本的函数和控制结构
25	
3.1	使用函数
25	
3.1.1	创建函数
25	
3.1.2	局部函数
26	
3.1.3	函数的参数和返回值
26	
3.1.4	函数作为Lua值
27	
3.2	用if语句进行判断
28	

3.2.1 简单的条件	28
3.2.2 复杂的表达式	29
3.2.3 扩展的条件语句	29
3.2.4 显示个人问候信息	29
3.3 用while语句来重复动作	30
3.3.1 计算阶乘	31
3.3.2 while和repeat之间的差异	32
3.4 用数值执行for循环	32
3.4.1 计算阶乘	34
3.4.2 循环条件的求值	34
3.4.3 循环中的变量作用域	34
3.5 小结	35
第4章 使用表	36
4.1 用表来存储数据	36
4.1.1 创建和索引表	36
4.1.2 从表中清除元素	37
4.1.3 字符串关键字的快捷方式	37
4.1.4 创建有内容的表	38
4.2 把表当作数组使用	39
4.2.1 创建数组	39
4.2.2 获取数组的长度	40
4.2.3 在数组中添加元素	40
4.2.4 从数组中删除元素	41
4.2.5 对数组中的元素排序	42
4.3 用名称空间使用表	

43	
4.3.1	创建util名称空间
43	
4.3.2	向util中添加函数
43	
4.4	表的面向对象编程
44	
4.4.1	创建非面向对象计数器
45	
4.4.2	把表作为简单的对象
45	
4.4.3	用冒号调用对象方法
46	
4.4.4	用冒号定义函数
46	
4.4.5	创建更佳的计数器
47	
4.5	利用元表对表进行扩展
48	
4.5.1	添加元表
48	
4.5.2	定义元方法
49	
4.5.3	旁路元表
54	
4.6	小结
54	
第5章	高级函数和控制结构
55	
5.1	多值返回
55	
5.2	将十六进制转换成RGB
55	
5.3	指定多个值
56	
5.4	返回值丢失
56	
5.5	WoW中的多个返回值
57	
5.6	接受可变数目的参数
58	
5.7	声明变参函数
59	
5.8	结合select()函数使用...
60	
5.9	范型for循环和迭代器
61	
5.10	范型for语句的语法
61	

5.11 遍历表的数组部分

61

5.12 遍历完整的表

62

5.13 表的清除

63

5.14 其他的迭代器

63

5.15 对表的数组排序

63

5.16 定义样例数据

64

5.17 默认的排序顺序

64

5.18 创建比较函数

64

5.19 创建更复杂的比较函数

65

5.20 小结

66

第6章 Lua标准库

67

6.1 表库

68

6.1.1 table.concat (table [, sep [, i [, j]])

68

6.1.2 table.insert (table, [pos,] value)

68

6.1.3 table.maxn (table)

68

6.1.4 table.remove (table [, pos])

69

6.1.5 table.sort (table [, comp])

69

6.2 数学函数库

69

6.3 字符串函数

72

6.4 格式化新字符串

73

6.5 模式匹配

75

6.5.1 字符类型

75

6.5.2 模式选项

77

6.5.3 模式捕获

77

6.5.4 模式锚点

78	
6.5.5 模式的例子	
78	
6.6 模式匹配函数	
78	
6.7 Lua在WoW插件中的补充	
80	
6.8 函数别名	
81	
6.9 小结	
82	
第7章 学习XML	
83	
7.1 XML——一种标记语言	
83	
7.1.1 XML与HTML的关系	
84	
7.1.2 XML的组件	
84	
7.1.3 创建规范的XML	
85	
7.2 验证XML文档	
86	
7.2.1 规范定义的样例	
86	
7.2.2 XML文件的例子	
87	
7.2.3 魔兽世界中的XML	
88	
7.2.4 使用GradientType	
89	
7.2.5 探讨模式	
89	
7.3 小结	
89	
第 部分 魔兽世界编程	
第8章 魔兽世界编程概述	
91	
8.1 游戏中运行和测试代码	
91	
8.1.1 安装	
91	
8.1.2 使用/script和/run	
92	
8.1.3 显示输出	
92	
8.2 在游戏中编辑代码	
93	
8.2.1 TinyPad	

93	
8.2.2 Omnibus	
94	
8.2.3 WowLua	
94	
8.3 编写自定义print()函数	
94	
8.4 插件和脚本的局限性	
95	
8.5 用Blizzard插件编写资源	
95	
8.5.1 用户界面自定义工具	
95	
8.5.2 用户界面和宏论坛	
97	
8.6 论坛资源	
97	
8.6.1 Wowwiki	
97	
8.6.2 魔兽版本监视器	
97	
8.7 小结	
97	
第9章 插件解析	
98	
9.1 开发属于您自己的插件文件夹	
98	
9.1.1 暴雪的插件	
98	
9.1.2 自定义插件	
99	
9.2 插件组件和文件	
99	
9.2.1 内容表格文件(.toc)	
99	
9.2.2 Lua脚本文件	
104	
9.2.3 XML文件	
104	
9.2.4 媒体文件	
104	
9.3 本地化您的插件	
106	
9.3.1 可用的本地语言	
106	
9.3.2 本地化的原因	
106	
9.3.3 鼓励玩家建议	
106	

9.3.4 实现本地化	107
9.4 创建插件框架	108
9.4.1 为插件命名(创建目录)	108
9.4.2 创建.toc文件	109
9.4.3 创建框架.xml文件	109
9.5 使用外部库	109
9.6 小结	109
第10章 在XML中创建窗体	110
10.1 了解UI对象	110
10.1.1 为对象进行命名	110
10.1.2 指定父对象	111
10.1.3 设置对象尺寸	112
10.1.4 锚定对象	113
10.1.5 将窗体和图形元素分层	115
10.1.6 一般属性	117
10.2 创建纹理	118
10.2.1 添加颜色	118
10.2.2 添加渐变效果	118
10.2.3 添加图形元素	120
10.3 使用字体字符串创建文本	120
10.3.1 使用模板	121
10.3.2 使用的例子	121
10.3.3 进一步自定义	122
10.4 探讨窗体类型	123
10.4.1 按钮(Button)	

123	
10.4.2 复选按钮(Check Button)	123
10.4.3 颜色选择(ColorSelect)	124
10.4.4 编辑框(Editbox)	124
10.4.5 游戏工具提示(Game Tooltip)	124
10.4.6 消息窗体(Message Frame)	125
10.4.7 小地图(Minimap)	125
10.4.8 模型(Model)	125
10.4.9 滚动信息窗体(Scrolling Message Frame)	126
10.4.10 滚动窗体(Scroll Frame)	126
10.4.11 简单的HTML窗体	126
10.4.12 滑动器(Slider)	126
10.4.13 状态栏(Status Bar)	127
10.4.14 飞行路线窗体	127
10.5 小结	128
第11章 向XML窗体中添加行为	129
11.1 理解事件和脚本	129
11.1.1 窗体脚本	129
11.1.2 游戏事件	129
11.2 使用脚本响应窗体事件	130
11.2.1 <OnEnter>和<OnLeave>	130
11.2.2 <OnLoad>	131
11.2.3 <OnEvent>	131
11.2.4 <OnClick>	132
11.2.5 <OnUpdate>	133

11.3 可用的窗体脚本	133
11.4 使用窗体方法改变窗体	135
11.4.1 常用方法	135
11.4.2 指定类型的函数	135
11.5 创建并使用模板	136
11.5.1 模板有用的原因	136
11.5.2 新建XML模板	136
11.5.3 使用XML模板	137
11.6 使用默认UI工具集模板	138
11.7 小结	139
第12章 创建您的第一个插件：CombatTracker	140
12.1 定义规格说明	140
12.1.1 CombatTracker 用户要求	140
12.1.2 找准游戏事件	140
12.2 创建插件架构	142
12.3 定义CombatTracker的XML窗体	143
12.4 测试CombatTrackerFrame	144
12.5 向CombatTrackerFrame添加操作脚本	144
12.5.1 <OnLoad>	145
12.5.2 <OnEvent>	145
12.5.3 <OnClick>	145
12.5.4 <OnDragStart>	145
12.5.5 <OnDragStop>	145
12.6 为CombatTracker.lua添加函数	146
12.6.1 CombatTracker_OnLoad(frame)	

146	
12.6.2	CombatTracker_OnEvent(frame, event, ...)
147	
12.6.3	CombatTracker_ReportDPS()
148	
12.7	测试CombatTracker
149	
12.7.1	窗体拖拽
149	
12.7.2	右击报告：第I部分
150	
12.7.3	测试战斗跟踪
150	
12.7.4	右击报告：第 II 部分
150	
12.8	小结
151	
	第13章 运用魔兽世界API
152	
13.1	理解魔兽世界API
152	
13.1.1	常规API
152	
13.1.2	类库API
153	
13.1.3	FrameXML函数
154	
13.1.4	受保护函数
154	
13.1.5	单位函数的使用与关闭
155	
13.2	创建简单的单位窗体
156	
13.2.1	创建窗体
157	
13.2.2	添加数据域
157	
13.2.3	设定窗体事件处理程序
160	
13.3	使用API
164	
13.3.1	显示和隐藏窗体
164	
13.3.2	实现简单的更新函数
165	
13.3.3	显示生命和法力值
167	
13.3.4	更新敌对信息
169	

13.4 小结

173

第14章 使用AddOn Studio创建基本插件

174

14.1 开始使用AddOn Studio

174

14.2 可视化设计您的界面

175

14.2.1 使用Toolbox

176

14.2.2 设置属性

176

14.3 编写代码

177

14.3.1 监听事件

177

14.3.2 自动添加事件处理程序

178

14.3.3 验证单位是否存在

178

14.3.4 添加“关闭”按钮的代码

180

14.4 部署插件

180

14.5 小结

181

第 部分 高级插件技术

第15章 利用编程模板

183

15.1 向TargetText添加增益

184

15.1.1 创立基本模板

185

15.1.2 定义基本的行为

186

15.2 创建增益图标

188

15.2.1 创建栅格

189

15.2.2 设定锚点

189

15.2.3 创建特定的wrapper函数

190

15.3 更新增益

191

15.3.1 对增益进行遍历

191

15.3.2 对更新操作进行封装

193

15.4 小结	194
第16章 脱离XML编写插件	195
16.1 探索基本的窗体创建	195
16.1.1 使用CreateFontString()函数	196
16.1.2 使用CreateTexture()函数	196
16.2 创建更好的玩家窗体	197
16.2.1 设置Lua	197
16.2.2 创建窗体	198
16.2.3 使它工作	199
16.3 比较LUA窗体和XML窗体	200
16.4 小结	203
第17章 使用安全模板	204
17.1 了解污染(Taint)	204
17.1.1 启用污染日志	205
17.1.2 执行污染	205
17.1.3 变量污染	207
17.1.4 蔓延的污染	208
17.1.5 理解被保护的窗体	209
17.1.6 无污染地设置窗体属性	209
17.2 使用安全模板	210
17.2.1 为动作按钮定义行为	210
17.2.2 做一些简单的选择	214
17.3 使您的单位窗体充满生机	218
17.3.1 移除禁止代码	218
17.3.2 接受单击	

219	
17.3.3	显示和隐藏窗体
219	
17.3.4	设置动作
219	
17.4	小结
220	
	第18章 创建命令
221	
18.1	检测基本的命令
221	
18.2	小提示、技巧以及好主意
222	
18.2.1	象征输入
222	
18.2.2	将确认用于模式匹配
223	
18.2.3	假跳转
224	
18.2.4	好主意
225	
18.3	小结
226	
	第19章 用函数钩改变已有行为
227	
19.1	什么是函数钩
227	
19.1.1	修改返回值
228	
19.1.2	函数钩工具脚本
228	
19.1.3	安全地钩一个函数
230	
19.1.4	安全地钩住脚本
230	
19.2	决定什么时候使用函数钩
231	
19.2.1	理解钩链(Hook Chain)
231	
19.2.2	寻找替代
232	
19.3	设计插件：MapZoomOut
233	
19.3.1	创建定时器
233	
19.3.2	创建MapZoomOut
234	
19.4	小结
236	

第20章 创建自定义图形界面

238

20.1 创建图形的普遍规则

238

20.2 GIMP

238

20.2.1 新建图片

239

20.2.2 添加图形组件

239

20.2.3 保存纹理

240

20.3 Adobe Photoshop

240

20.3.1 新建图像

241

20.3.2 添加图形组件

241

20.3.3 创建透明度通道

241

20.3.4 保存图像

242

20.4 Paint Shop Pro

243

20.4.1 新建图像

243

20.4.2 添加图像组件

244

20.4.3 创建透明度通道

244

20.4.4 保存图像

244

20.5 测试纹理

245

20.5.1 没有按钮出现

246

20.5.2 出现绿色方块

246

20.6 小结

247

第21章 使窗体移动

248

21.1 拖动和放置

248

21.2 标题区域

248

21.3 使用OnMouseDown和OnMouseUp

249

21.4 使用OnDragStart/OnDragStop

249
21.5 给窗体赋予生命
250
21.6 渐隐
251
21.7 小结
252
第22章 创建滚动窗体
253
22.1 使用滚动结构
255
22.1.1 定义滚动窗体
255
22.1.2 添加子滚动条
255
22.1.3 创建滚动条
257
22.2 创建伪滚动窗体
258
22.2.1 添加滚动条
261
22.2.2 利用鼠标滚轮进行滚动
261
22.3 小结
262
第23章 创建下拉菜单
263
23.1 创建简单的下拉菜单
263
23.1.1 添加切换按钮
264
23.1.2 创建下拉窗体
264
23.1.3 初始化下拉菜单
265
23.1.4 设置下拉菜单
267
23.1.5 测试下拉菜单
267
23.2 创建多级下拉菜单
268
23.3 创建高级下拉菜单
269
23.3.1 功能菜单项
270
23.3.2 CheckButton菜单项
270
23.4 调色板菜单项
271

23.5 自动创建菜单	272
23.6 小结	274
第24章 工具提示	275
24.1 操作工具提示	275
24.1.1 使用“:Set...”函数	275
24.1.2 包含自定义信息的工具提示	276
24.1.3 创建自定义工具提示	277
24.2 解析工具提示数据	278
24.2.1 基础结构剖析	278
24.2.2 基本扫描循环	278
24.3 小结	279
第25章 使用状态标题	280
25.1 创建第一个状态标题	280
25.1.1 重设鼠标单击	282
25.1.2 通过单击修改状态	284
25.2 控制视觉属性	285
25.2.1 显示和隐藏子标题	286
25.2.2 重置子标题的位置	286
25.2.3 重新定义子标题和标题的大小	287
25.2.4 重置标题位置	287
25.3 配置其他的属性	290
25.3.1 改变标题的单位	290
25.3.2 覆盖绑定键	291
25.3.3 在栈中存储状态	292
25.4 创建宏序列	

293	
25.4.1	定义序列
294	
25.4.2	设定基本的序列
295	
25.4.3	重置您的序列
297	
25.5	创建Pie按钮
299	
25.5.1	设定Pie按钮
299	
25.5.2	构建窗体
300	
25.5.3	完成行为
302	
25.6	小结
303	
第26章	驱动状态改变
304	
26.1	使用状态驱动
304	
26.1.1	延时驱动状态改变
305	
26.1.2	理解宏选项
306	
26.1.3	合并情景类型
310	
26.1.4	在标题中共享状态
311	
26.2	使用定位模板
311	
26.2.1	触发定位改变
312	
26.2.2	重新映射定位转换
312	
26.3	创建动作弹出菜单
313	
26.3.1	创建窗口
313	
26.3.2	添加按钮
314	
26.3.3	定义基本行为
316	
26.3.4	应用锚点
320	
26.4	小结
322	
第27章	利用组模板创建单位窗体
323	

27.1 模板的工作原理	323
27.2 配置SecureGroupHeader	324
27.3 SecureGroupPetHeader模板属性	326
27.4 新建SecureGroupHeader	326
27.5 定义XML模板	327
27.6 添加行为脚本	328
27.6.1 初始化窗体	329
27.6.2 更新单位信息	329
27.6.3 更新最大生命值	330
27.6.4 添加拖拽支持	331
27.7 测试 BasicUnitFrames	331
27.8 小结	332
第 部分 参考文献	
第28章 API参考文献	334
28.1 API参考约定	334
28.1.1 函数签名	334
28.1.2 参数和返回列表	335
28.2 API 伪类型	335
28.3 按字母排序的API列表	337
第29章 API目录	682
29.1 动作	682
29.2 动作栏	683
29.3 插件	683
29.4 竞技场	683
29.5 拍卖	684

29.6 银行	685
29.7 战场	685
29.8 绑定	687
29.9 增益	687
29.10 摄像机	688
29.11 频道	688
29.12 聊天	690
29.13 仅在中国使用的函数	690
29.14 客户端	691
29.15 战斗	691
29.16 申诉	691
29.17 容器	691
29.18 手工艺	692
29.19 光标	693
29.20 配置变量	694
29.21 调试	694
29.22 阵营	695
29.23 GM调查	695
29.24 GM 请求	695
29.25 公会	695
29.26 公会银行	697
29.27 查看	698
29.28 副本	698
29.29 物品栏	699
29.30 物品	

699
29.31 物品链接
701
29.32 物品文本
701
29.33 知识库
702
29.34 键盘
702
29.35 寻求组队/寻求成员
703
29.36 拾取
704
29.37 Mac
704
29.38 Macro
705
29.39 邮件
706
29.40 地图
707
29.41 商人
707
29.42 其他
708
29.43 不同的单击
709
29.44 金钱
709
29.45 移动
709
29.46 NPC
710
29.47 小队
710
29.48 宠物
711
29.49 请求
713
29.50 玩家
713
29.51 配置
715
29.52 PvP
715
29.53 任务
716
29.54 团队
718

29.55 技能	719
29.56 社交	719
29.57 凹槽	719
29.58 声音	720
29.59 法术	720
29.60 形态	721
29.61 统计信息	722
29.62 天赋	723
29.63 目标	723
29.64 飞行器	724
29.65 跟踪	724
29.66 交易	724
29.67 交易技能	725
29.68 训练师	726
29.69 UI	727
29.70 单位	728
29.71 实用程序	730
29.72 视频	731
29.73 语音	732
29.74 区域	733
第30章 事件参考	734
第31章 工具集参考	794
31.1 工具集API	794
31.1.1 UIObject	794
31.1.2 FontInstance	

794	
31.1.3 Region	
795	
31.1.4 VisibleRegion	
796	
31.1.5 LayeredRegion	
796	
31.1.6 Font	
797	
31.1.7 Texture	
797	
31.1.8 FontString	
798	
31.1.9 Frame	
798	
31.1.10 GameTooltip	
801	
31.1.11 EditBox	
805	
31.1.12 Slider	
807	
31.1.13 Model	
807	
31.1.14 MessageFrame	
808	
31.1.15 Cooldown	
809	
31.1.16 ScrollFrame	
809	
31.1.17 ColorSelect	
810	
31.1.18 Minimap	
811	
31.1.19 SimpleHTML	
811	
31.1.20 ScrollingMessageFrame	
813	
31.1.21 StatusBar	
814	
31.1.22 Button	
814	
31.1.23 PlayerModel	
816	
31.1.24 CheckButton	
817	
31.1.25 TabardModel	
817	
31.1.26 DressUpModel	
818	

31.2 处理程序列表

818

第 部分 附录

附录A 发布插件

831

附录B 利用版本控制系统跟踪

历史记录

838

附录C 最好的代码风格

841

附录D 避免常见的错误

858

附录E 使用插件库

860

附录F 作者和插件社区

866

附录G 词汇表

868

编辑推荐

阅读完《魔兽世界编程宝典：World of Warcraft Addons完全参考手册》，您将学会如何对魔兽世界界面进行自定义修改。如果您初编程，将从《魔兽世界编程宝典：World of Warcraft Addons完全参考手册》学习到Lua和XML的基础知识，以及如何开始编写自己的第一个插件。如果您已经熟悉一些原始插件，那么可以直接跳到《魔兽世界编程宝典：World of Warcraft Addons完全参考手册》的第1和第 部分，使用模板、函数钩自定义图形、状态标题等。《魔兽世界编程宝典：World of Warcraft Addons完全参考手册》有助于您发布自己的插件，使WOW更加有趣。

《魔兽世界编程宝典：World of Warcraft Addons完全参考手册》内容

- 如何响应事件，创建窗体和使WOW API与游戏交互
- 通过一系列交互式的示例、值和变量，以及自定义函数和控制结构来学习Lua编程语言
- 学习基本和高级的函数和控制结构
- 编写结构良好的XML和有效地使用模式
- 实现已保存的变量、slash命令、安全模板、动态窗体、解析工具提示等
- 创建自定义图形、滚动窗体和下拉菜单

轻松地在综合性的API参考文献、API目录、事件和工具集中找出特定的函数

精彩短评

- 1、第一印象是好厚的一本书啊，有一定编程基础的朋友看起来倒是很好理解，不知道初涉此行的是什么感觉.
- 2、牛B
- 3、开始买的时候不知道是这么厚德一本，我实在是没时间去研究，估计要真的是看完，肯定要一年~
- 4、一本 program fans 的开胃手册。
- 5、讲的比较基础，非常适合入门。帮朋友带了几本。
- 6、好东西，现在终于用上了自己开发的插件
- 7、很厚，要看完，的确要花很多时间
- 8、里面的例子全是和魔兽世界有关的，wower应该会喜欢，目前正在看。不过我觉得如果你想通过这个学编程的话，这本书不推荐，里面的例子通用性不是很强。
- 9、可惜国内没第二版的翻译，不过它有它的参考价值，作为程序员当作旁门左道的东西学学lua也不错哈~

最后希望国服早日开C.T.M

- 10、不错，值得推荐！~~~~~写插件的好助手！~~~~~
- 11、很实在也很详细的介绍了魔兽宏 编程等等的一系列东西，个人觉得很值！
- 12、都过两年了，游戏更新快，确实有点落伍
- 13、书是正版，就是没有快递要自己去邮局拿
- 14、其实想买这书很久了，一直担心自己基础不够，看不懂。后来想既然喜欢就慢慢学么~~~建议有一些基础的玩家看，确实能学到不好实用的东东~~~
- 15、买了，但是没仔细看，而且本人c++学的好差，心塞
- 16、很多内容还不懂，待努力
- 17、高中时玩毁灭术不会写宏命令，以为这本书会教，结果翻开是教你用lua.....
- 18、虽然在实例上基本考虑的是wow的事例，但是作为入门级别的教材，还是挺适合lua入门的
- 19、很棒！老公喜欢！！
- 20、没有买到Lua编程大全，先用这本书看看，还能学学魔兽的插件编程
- 21、不错，感觉不难理解
- 22、书不错，图太少了。

《魔兽世界编程宝典》

精彩书评

1、1.这是一本Lua的教程。2.它能让你从头到尾保持对学习Lua的兴趣。3.初学者看起来不太费力，但是有其他语言编程经验才可通吃。4.这还不够？

《魔兽世界编程宝典》

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu000.com