

《Erlang趣学指南》

图书基本信息

书名：《Erlang趣学指南》

13位ISBN编号：9787115431906

出版时间：2016-9-7

页数：506

译者：邓辉,孙鸣

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.tushu111.com

《Erlang趣学指南》

内容概要

这是一本讲解Erlang编程语言的入门指南，内容通俗易懂，插图生动幽默，示例短小清晰，结构安排合理。书中从Erlang的基础知识讲起，融汇所有的基本概念和语法。内容涉及模块、函数、类型、递归、错误和异常、常用数据结构、并行编程、多处理、OTP、事件处理，以及所有Erlang的重要特性和强大功能。

作者简介

作者介绍

Fred Hébert是一位自学成才的程序员，具有Web前端、Web 服务开发经验以及全面的后端多语言编程经验。他撰写的Erlang在线教程Learn You Some Erlang for Great Good! 被公认为好的Erlang学习教程。在Erlang Solutions工作期间，他撰写过Erlang培训资料，并在西方国家到处教授Erlang编程。目前，他在用Erlang开发一款实时竞价平台（AdGear），他被提名为2012年度Erlang User。

译者介绍

邓辉 独立咨询顾问，捷中科技首席架构师，有10年Erlang编程经验。《敏捷软件开发：原则、模式与实践》一书的译者，Erlang领域两本重要著作《面对软件错误构建可靠的分布式系统》《硝烟中的Erlang》中文译本的审校者和译者。对指示语义、类型理论、程序语言理论和分布式系统设计有浓厚的兴趣。目前在研究学习TLA+。

孙鸣 中兴通讯杰出架构师，具有丰富的电信软件架构设计经验。《敏捷软件开发：原则、模式与实践》（C#版）、《硝烟中的Erlang》等书的译者。对函数式编程有浓厚的兴趣，平时喜欢设计各种各样的小语言。喜欢的编程语言是Scheme和Erlang，并用Erlang语言重写了Structure and Interpretation of Computer Programs一书中的全部代码。

书籍目录

第1章 启程 1

- 1.1 使用Erlang shell 1
 - 1.1.1 输入shell命令 1
 - 1.1.2 退出shell 2
- 1.2 Erlang基础知识 3
 - 1.2.1 数值类型 3
 - 1.2.2 不变的变量 4
 - 1.2.3 原子 5
 - 1.2.4 布尔代数和比较操作符 6
 - 1.2.5 元组 8
 - 1.2.6 列表 10
 - 1.2.7 列表推导式 13
- 1.3 处理二进制数据 14
 - 1.3.1 位语法 15
 - 1.3.2 二进制数的按位操作 17
 - 1.3.3 二进制字符串 18
 - 1.3.4 二进制推导式 19

第2章 模块 20

- 2.1 什么是模块 20
- 2.2 创建模块 21
- 2.3 编译代码 23
- 2.4 定义宏 26
- 2.5 模块的其他内容 27
 - 2.5.1 元数据 27
 - 2.5.2 环形依赖 28

第3章 函数 29

- 3.1 模式匹配 29
 - 3.1.1 模式进阶 31
 - 3.1.2 绑定中的变量 31
- 3.2 卫语句 33
- 3.3 if表达式 34
- 3.4 case . . . of表达式 37
- 3.5 如何选择 38

第4章 类型 39

- 4.1 动态强类型 39
- 4.2 类型转换 40
- 4.3 数据类型检测函数 41
- 4.4 致静态类型爱好者 43

第5章 递归 44

- 5.1 递归的工作原理 44
 - 5.1.1 列表的长度 45
 - 5.1.2 列表长度的尾递归实现 46
- 5.2 更多递归函数 47
 - 5.2.1 duplicate函数 47
 - 5.2.2 reverse函数 48
 - 5.2.3 sublist函数 49
 - 5.2.4 zip函数 50

- 5.2.5 快速排序 51
- 5.3 不仅仅是列表 53
- 5.4 递归思维 55
- 第6章 高阶函数 58
 - 6.1 一切都是函数 58
 - 6.2 匿名函数 60
 - 6.2.1 匿名函数的其他用途 60
 - 6.2.2 函数的作用域和闭包 61
 - 6.3 映射、过滤器、折叠以及其他 63
 - 6.3.1 过滤器 63
 - 6.3.2 折叠一切 64
 - 6.3.3 其他抽象 66
- 第7章 错误和异常 67
 - 7.1 错误编译 67
 - 7.1.1 编译期错误 67
 - 7.1.2 逻辑错误 69
 - 7.1.3 运行时错误 69
 - 7.2 引发异常 71
 - 7.2.1 出错异常 71
 - 7.2.2 退出异常 72
 - 7.2.3 抛出异常 73
 - 7.3 处理异常 74
 - 7.3.1 处理不同类型的异常 74
 - 7.3.2 catch后的after语句 76
 - 7.3.3 尝试多个表达式 76
 - 7.3.4 更多选择 78
 - 7.4 在树中使用try语句 80
- 第8章 用函数式思维解决问题 82
 - 8.1 逆波兰式计算器 82
 - 8.1.1 RPN计算器的工作原理 82
 - 8.1.2 实现RPN计算器 84
 - 8.1.3 代码测试 85
 - 8.2 从希思罗到伦敦 87
 - 8.2.1 递归地解决问题 87
 - 8.2.2 编写代码 89
 - 8.2.3 不使用Erlang shell运行程序 93
- 第9章 常用数据结构简介 95
 - 9.1 记录 95
 - 9.1.1 定义记录 95
 - 9.1.2 读取记录字段值 97
 - 9.1.3 更新记录 99
 - 9.1.4 共享记录定义 99
 - 9.2 键/值存储 100
 - 9.2.1 小数据量存储 100
 - 9.2.2 大数据量存储：字典和通用平衡树 101
 - 9.3 集合 102
 - 9.4 有向图 104
 - 9.5 队列 104
 - 9.6 小结 105

第10章 并发编程漫游指南	106
10.1 不必惊慌	106
10.2 并发概念	107
10.2.1 伸缩性	108
10.2.2 容错	108
10.2.3 并发实现	109
10.3 并非完全不能线性伸缩	110
10.4 再见，谢谢你的鱼	111
10.4.1 创建进程	112
10.4.2 发送消息	113
10.4.3 接收消息	114
第11章 深入多重处理	118
11.1 定义进程状态	118
11.2 隐藏消息实现	120
11.3 超时	121
11.4 选择性接收	123
11.4.1 选择性接收的风险	124
11.4.2 邮箱使用的其他风险	126
第12章 错误与进程	127
12.1 链接	127
12.1.1 捕获退出信号	129
12.1.2 老异常，新概念	130
12.2 监控器	133
12.3 命名进程	134
第13章 并发应用设计	139
13.1 理解问题	139
13.2 设计协议	141
13.3 目录结构	143
13.4 事件模块	143
13.4.1 事件和循环	144
13.4.2 增加接口	146
13.5 事件服务器	148
13.5.1 处理消息	149
13.5.2 代码热升级	152
13.5.3 隐藏消息细节	153
13.6 测试	155
13.7 增加监督功能	156
13.8 命名空间	157
第14章 OTP简介	158
14.1 提炼通用进程	158
14.2 基础服务器	159
14.2.1 kitty服务器	159
14.2.2 通用化同步调用	161
14.2.3 通用化服务器循环	162
14.2.4 启动函数	164
14.2.5 通用化kitty服务器	165
14.3 专用与通用	166
14.4 面向未来的回调	167
14.4.1 init函数	167

- 14.4.2 handle_call函数 168
- 14.4.3 handle_cast函数 169
- 14.4.4 handle_info函数 169
- 14.4.5 terminate函数 169
- 14.4.6 code_change函数 169
- 14.5 gen_server实践 170
- 第15章 令人愤怒的有限状态机 174
 - 15.1 什么是有限状态机 174
 - 15.2 通用有限状态机 178
 - 15.2.1 init函数 178
 - 15.2.2 StateName函数 178
 - 15.2.3 handle_event函数 179
 - 15.2.4 handle_syn_event函数 179
 - 15.2.5 code_change和terminate函数 179
 - 15.3 交易系统规格说明 179
 - 15.3.1 操作定义 180
 - 15.3.2 定义状态图和状态迁移 181
 - 15.4 游戏交易 186
 - 15.4.1 公共接口 186
 - 15.4.2 FSM到FSM的函数 187
 - 15.4.3 gen_fsm回调函数 189
 - 15.5 为自己骄傲 196
 - 15.6 适用于真实世界吗 197
- 第16章 事件处理器 198
 - 16.1 处理它！*泵式散弹枪* 198
 - 16.2 通用事件处理器 199
 - 16.2.1 init和terminate函数 200
 - 16.2.2 handle_event函数 200
 - 16.2.3 handle_call函数 201
 - 16.2.4 handle_info函数 201
 - 16.2.5 code_change函数 201
 - 16.3 现在是冰壶比赛时间 201
 - 16.3.1 记分牌 202
 - 16.3.2 比赛事件 203
 - 16.3.3 通知新闻界 206
- 第17章 谁来监督监督者 211
 - 17.1 监督者中的概念 211
 - 17.2 使用监督者 213
 - 17.2.1 重启策略 213
 - 17.2.2 重启限制 215
 - 17.2.3 子进程规格说明 215
 - 17.3 乐队排练 217
 - 17.3.1 音乐人 217
 - 17.3.2 乐队监督者 220
 - 17.4 动态监督 223
 - 17.4.1 动态使用标准监督者 223
 - 17.4.2 使用simple_one_for_one监督者 224
- 第18章 构建应用 226
 - 18.1 进程池 226

- 18.1.1 洋葱理论 227
- 18.1.2 进程池监督树 228
- 18.2 实现监督者 230
- 18.3 进程池服务器 233
- 18.4 实现工作者 239
- 18.5 运行进程池 240
- 18.6 小结 242
- 第19章 构建OTP应用 243
 - 19.1 我还有辆车是一个游泳池 243
 - 19.2 应用资源文件 244
 - 19.3 把ppool转换成OTP应用 246
 - 19.4 application行为 247
 - 19.5 从混乱到应用 249
 - 19.6 库应用 251
- 第20章 深入OTP应用 253
 - 20.1 从OTP应用到真实的应用 253
 - 20.1.1 应用文件 254
 - 20.1.2 应用回调模块和监督者 255
 - 20.1.3 分派器 256
 - 20.1.4 计数模块 264
 - 20.2 运行应用 265
 - 20.3 包含应用 267
 - 20.4 复杂的终止 267
- 第21章 发布 268
 - 21.1 修理漏水的管道 268
 - 21.1.1 终止VM 268
 - 21.1.2 更新应用文件 269
 - 21.1.3 编译应用 270
 - 21.2 使用systools构建发布 270
 - 21.2.1 创建启动文件 271
 - 21.2.2 发布打包 272
 - 21.3 使用Reltool构建发布 273
 - 21.4 Reltool技巧集 279
 - 21.5 基于release文件发布 282
- 第22章 升级Process Quest 283
 - 22.1 升级面临的问题 283
 - 22.2 Erlang学习的第9级 285
 - 22.3 Process Quest 286
 - 22.3.1 regis-1.0.0应用 287
 - 22.3.2 processquest-1.0.0应用 287
 - 22.3.3 sockserv-1.0.0应用 288
 - 22.3.4 发布 289
 - 22.4 改进Process Quest 291
 - 22.4.1 更改code_change函数 291
 - 22.4.2 增加appup文件 293
 - 22.4.3 发布升级 296
 - 22.5 Relup回顾 299
- 第23章 套接字编程 301
 - 23.1 IO列表 301

- 23.2 UDP和TCP：伙伴协议 302
 - 23.2.1 UDP套接字 303
 - 23.2.2 TCP套接字 306
- 23.3 使用Inet进行更多的控制 308
- 23.4 重新审视sockserv 310
- 23.5 下一步的工作 319
- 第24章 EUnit：单元测试框架 320
 - 24.1 什么是EUnit 320
 - 24.2 测试生成器 324
 - 24.3 测试夹具 326
 - 24.3.1 其他测试控制方法 328
 - 24.3.2 测试文档 329
 - 24.4 测试regis 329
 - 24.5 EUnit小结 338
- 第25章 ETS：免费的内存NoSQL数据库 339
 - 25.1 为什么需要ETS 339
 - 25.2 ETS的概念 340
 - 25.3 ETS的基本操作 342
 - 25.3.1 表的创建和删除 342
 - 25.3.2 数据的插入和查询 343
 - 25.4 匹配操作 345
 - 25.5 选择操作 347
 - 25.6 DETS 351
 - 25.7 少说一点，多做一点 351
 - 25.7.1 接口 352
 - 25.7.2 实现细节 352
- 第26章 分布式编程 357
 - 26.1 这是我的火枪 358
 - 26.2 分布式计算中的谬误 359
 - 26.2.1 网络是可靠的 360
 - 26.2.2 网络没有延迟 360
 - 26.2.3 带宽是无限的 361
 - 26.2.4 网络是安全的 361
 - 26.2.5 网络拓扑不会变化 362
 - 26.2.6 只有一个管理员 363
 - 26.2.7 传输成本是零 363
 - 26.2.8 网络是同质的 363
 - 26.2.9 谬误小结 364
 - 26.3 死亡还是失去联系 364
 - 26.4 CAP定理 365
 - 26.4.1 一致性 366
 - 26.4.2 可用性 366
 - 26.4.3 分区容忍 366
 - 26.4.4 僵尸幸存者和CAP 367
 - 26.5 搭建Erlang集群 369
 - 26.5.1 节点命名 369
 - 26.5.2 连接节点 370
 - 26.5.3 更多工具 371
 - 26.6 cookie 373

- 26.7 远程shell 374
- 26.8 隐藏节点 375
- 26.9 防火墙问题 376
- 26.10 高级调用 376
 - 26.10.1 net_kernel模块 376
 - 26.10.2 global模块 377
 - 26.10.3 rpc模块 378
- 26.11 小结 380
- 第27章 分布式OTP应用 381
 - 27.1 更多OTP内容 381
 - 27.2 接管和故障切换 382
 - 27.3 神奇8号球 383
 - 27.3.1 构建应用 384
 - 27.3.2 变身分布式应用 387
- 第28章 不寻常的Common Test 391
 - 28.1 什么是Common Test 391
 - 28.2 Common Test的组织结构 392
 - 28.3 创建一个简单的测试套件 393
 - 28.4 带状态的测试 396
 - 28.5 测试组 397
 - 28.5.1 定义测试组 398
 - 28.5.2 测试组属性 399
 - 28.5.3 会议室 400
 - 28.6 再谈测试套件 404
 - 28.7 测试规格说明 404
 - 28.7.1 规格说明文件内容 405
 - 28.7.2 创建规格说明文件 406
 - 28.7.3 通过规格说明文件运行测试 407
 - 28.8 大规模测试 407
 - 28.8.1 创建分布式测试规格说明文件 409
 - 28.8.2 运行分布式测试 410
 - 28.9 集成Common Test和EUnit 411
 - 28.10 还有其他内容吗 411
- 第29章 Mnesia—记忆的艺术 412
 - 29.1 Mnesia是什么 412
 - 29.2 应该存储什么呢 413
 - 29.2.1 需要存储的数据 413
 - 29.2.2 表结构 414
 - 29.3 从记录到表 415
 - 29.4 Mnesia数据模式和表 416
 - 29.5 创建表 418
 - 29.5.1 安装数据库 418
 - 29.5.2 启动应用 421
 - 29.6 数据表存取上下文 422
 - 29.7 读、写以及其他操作 423
 - 29.8 实现第一个请求 424
 - 29.8.1 测试增加服务 424
 - 29.8.2 测试查询 427
 - 29.8.3 账目和新的需求 430

29.9	满足老板	432
29.10	删除操作示例	434
29.11	列表推导式查询	437
29.12	记住Mnesia	438
第30章	类型规格说明与Dialyzer	439
30.1	PLT是最好的三明治	439
30.2	成功类型化	440
30.3	类型推导和错误	442
30.4	类型的种类	445
30.4.1	单例类型	445
30.4.2	联合类型和内置类型	445
30.4.3	定义类型	448
30.4.4	记录类型	449
30.5	为函数增加类型声明	450
30.6	类型定义实践	453
30.7	类型导出	457
30.8	OTP行为类型	459
30.9	多态类型	460
30.9.1	我们买了一个动物园	460
30.9.2	注意事项	462
30.10	Dialyzer, 我的好朋友	463
30.11	朋友们, 就这么多	463
后记		464
附录	Erlang语法	467

精彩短评

1、书籍不错，比Joe大神些的还要详细一点儿，适合初级入门。

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.tushu111.com