

УДК 004.43Elixir
ББК 32.972
Ю70

Юрич С.

Ю70 Elixir в действии / пер. с англ. Н. А. Нестеровой. – М.: ДМК Пресс, 2020. – 376 с.: ил.

ISBN 978-5-97060-773-2

В книге рассматриваются базовые характеристики и примеры использования языка Elixir, предназначенного для создания масштабируемых, распределенных и отказоустойчивых систем, работающих на основе виртуальной машины Erlang. Сегодня ее используют в самых разных областях: для создания инструментов совместной работы, систем открытых торгов в режиме реального времени, серверов баз данных, многопользовательских онлайн-игр и др.

Приведенные в книге примеры подчеркивают преимущества функционального и конкурентного программирования на языке Elixir, которые могут обеспечить бесперебойную работу систем, обслуживающих огромное количество пользователей со всего мира.

Издание предназначено для профессиональных разработчиков, имеющих опыт программирования на Java, C#, Ruby, C++ или другом языке общего назначения.

УДК 004.43Elixir
ББК 32.972

Original English language edition published by Manning Publications USA, USA. Copyright © 2019 by Manning Publications Co. Russian-language edition copyright © 2020 by DMK Press. All rights reserved.

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

ISBN 978-1-617-29502-7 (анг.)
ISBN 978-5-97060-773-2 (рус.)

Copyright © 2019 by Manning Publications Co.
© Оформление, издание, перевод, ДМК Пресс, 2020

Содержание

Отзывы о первом издании	10
Предисловие	11
Благодарности	12
О книге	13
Об авторе	17
Иллюстрация на обложке	18
Глава 1. Первые шаги	19
1.1. Общие сведения об Erlang	19
1.1.1. Высокая доступность	20
1.1.2. Конкурентная модель Erlang	21
1.1.3. Системы на стороне сервера	23
1.1.4. Платформа разработки	25
1.2. Общие сведения об Elixir	26
1.2.1. Упрощенный код	27
1.2.2. Композиция функций	30
1.2.3. Выводы	31
1.3. Недостатки	32
1.3.1. Скорость	32
1.3.2. Экосистема	32
Выводы	33
Глава 2. Основы языка	34
2.1. Интерактивная оболочка	35
2.2. Работа с переменными	36
2.3. Организация кода	37
2.3.1. Модули	37
2.3.2. Функции	39
2.3.3. Арность функций	42
2.3.4. Видимость функций	43
2.3.5. Импорты и псевдонимы	44
2.3.6. Атрибуты модулей	45
2.3.7. Комментарии	47
2.4. Понятие системы типов	48
2.4.1. Числа	48
2.4.2. Атомы	49
2.4.3. Кортежи	51
2.4.4. Списки	52
2.4.5. Иммутабельность	56
2.4.6. Словари	59
2.4.7. Бинарные данные и битовые строки	62
2.4.8. Строки	63
2.4.9. Функции первого класса	65
2.4.10. Прочие встроенные типы	67
2.4.11. Типы данных более высокого уровня	68
2.4.12. Списки ввода-вывода	72
2.5. Операторы	73
2.6. Макросы	74

6 ❖ Содержание

2.7. Среда выполнения.....	75
2.7.1. Модули и функции в среде выполнения	75
2.7.2. Запуск среды выполнения.....	78
Выводы	80
Глава 3. Поток управления.....	81
3.1. Сопоставление с образцом	81
3.1.1. Оператор сопоставления.....	82
3.1.2. Сопоставление кортежей	82
3.1.3. Сопоставление с константой	83
3.1.4. Переменные в качестве образцов	84
3.1.5. Сопоставление списков.....	85
3.1.6. Сопоставление словарей	86
3.1.7. Сопоставление с битовыми строками и бинарными данными.....	86
3.1.8. Сложные сопоставления	88
3.1.9. Обобщенное поведение	90
3.2. Сопоставление с образцом в функциях.....	90
3.2.1. Функции с несколькими предложениями	91
3.2.2. Ограничители	94
3.2.3. Анонимные функции с несколькими предложениями.....	96
3.3. Условные конструкции	97
3.3.1. Ветвление с помощью функций с несколькими предложениями	97
3.3.2. Классические конструкции ветвления.....	99
3.3.3. Специальная форма with	101
3.4. Циклы и итерации	104
3.4.1. Итерация на основе рекурсии	105
3.4.2. Хвостовые вызовы функций.....	106
3.4.3. Функции высшего порядка	109
3.4.4. Генераторы.....	114
3.4.5. Поток	116
Выводы	119
Глава 4. Абстракции данных	121
4.1. Создание абстракций с помощью модулей.....	122
4.1.1. Создание простой абстракции.....	123
4.1.2. Сложные абстракции	125
4.1.3. Структурирование данных с помощью словарей.....	126
4.1.4. Абстракции на основе структур.....	127
4.1.5. Прозрачность данных.....	131
4.2. Работа с иерархическими данными	133
4.2.1. Генерация идентификаторов.....	134
4.2.2. Обновление записей.....	136
4.2.3. Обновление неизменяемых иерархических данных	138
4.2.4. Итеративное обновление	140
4.2.5. Практика: импорт из файла.....	141
4.3. Полиморфизм с помощью протоколов	143
4.3.1. Общие сведения о протоколах.....	143
4.3.2. Реализация протокола.....	144
4.3.3. Встроенные протоколы	145
Выводы	147
Глава 5. Основы конкурентности	148
5.1. Конкурентность в BEAM	148
5.2. Работа с процессами	151

5.2.1. Создание процессов	152
5.2.2. Обмен сообщениями	154
5.3. Серверные процессы с сохранением состояния	159
5.3.1. Серверные процессы	159
5.3.2. Сохранение состояния процесса	163
5.3.3. Изменяемое состояние	165
5.3.4. Сложные состояния	168
5.3.5. Регистрация процессов	172
5.4. Особенности времени выполнения	173
5.4.1. Последовательность выполнений действий в процессах	173
5.4.2. Бездонные почтовые ящики процессов	175
5.4.3. Конкурентность без разделения ресурсов	176
5.4.4. Внутреннее устройство планировщиков	177
Выводы	178
Глава 6. Обобщенные серверные процессы	179
6.1. Создание обобщенного серверного процесса	179
6.1.1. Подключение к обобщенному коду с помощью модулей	180
6.1.2. Реализация обобщенного кода	181
6.1.3. Использование обобщенной абстракции	182
6.1.4. Поддержка асинхронных запросов	184
6.1.5. Упражнение: реорганизация сервера для списка дел	185
6.2. Использование GenServer	186
6.2.1. Поведения OTP	187
6.2.2. Подключение к GenServer	187
6.2.3. Обработка запросов	188
6.2.4. Обработка простых сообщений	190
6.2.5. Прочие особенности GenServer	191
6.2.6. Жизненный цикл процесса	194
6.2.7. Совместимые с OTP процессы	195
6.2.8. Упражнение: создание сервера для списка дел на основе GenServer	196
Выводы	196
Глава 7. Создание конкурентной системы	198
7.1. Работа с проектом mix	198
7.2. Управление несколькими списками дел	200
7.2.1. Создание кеш-процесса	201
7.2.2. Создание тестов	203
7.2.3. Анализ зависимостей процесса	206
7.3. Сохранение данных	208
7.3.1. Кодирование и сохранение	208
7.3.2. Использование базы данных	210
7.3.3. Анализ системы	213
7.3.4. Устранение узкого места процесса	214
7.3.5. Упражнение: пул процессов и синхронизация	217
7.4. Логика работы процессов	218
Выводы	219
Глава 8. Основы отказоустойчивости	220
8.1. Ошибки времени выполнения	221
8.1.1. Типы ошибок	221
8.1.2. Обработка ошибок	222
8.2. Ошибки в конкурентных системах	226
8.2.1. Установка связей между процессами	227
8.2.2. Мониторы	229

8.3. Супервизоры	230
8.3.1. Подготовка существующего кода	232
8.3.2. Запуск процесса-супервизора	232
8.3.3. Спецификации потомков	235
8.3.4. Обертка супервизора	237
8.3.5. Использование модуля обратного вызова	237
8.3.6. Связывание всех процессов	238
8.3.7. Частота перезапусков	241
Выводы	242
Глава 9. Изолирование последствий ошибок	243
9.1. Деревья супервизоров	244
9.1.1. Разделение слабо связанных частей	244
9.1.2. Усовершенствованное обнаружение процессов	247
9.1.3. Via-кортежи	249
9.1.4. Регистрация рабочих процессов базы данных	251
9.1.5. Наблюдение за рабочими процессами	253
9.1.6. Построение дерева супервизоров	256
9.2. Динамический запуск рабочих процессов	259
9.2.1. Регистрация серверных процессов	260
9.2.2. Динамические супервизоры	260
9.2.3. Обнаружение серверных процессов	262
9.2.4. Использование временных рабочих процессов	263
9.2.5. Тестирование системы	264
9.3. Let it crash	265
9.3.1. Процессы, отказа которых допускать нельзя	266
9.3.2. Обработка ожидаемых ошибок	267
9.3.3. Сохранение состояния	268
Выводы	269
Глава 10. За пределами GenServer	270
10.1. Задачи	270
10.1.1. Задачи с ожиданием ответа	271
10.1.2. Задачи без ожидания ответа	273
10.2. Агенты	275
10.2.1. Использование агентов	275
10.2.2. Агенты и конкурентность	276
10.2.3. Сервер списка дел на основе модуля Agent	277
10.2.4. Пределы возможностей агентов	279
10.3. Таблицы ETS	281
10.3.1. Основные операции	284
10.3.2. Хранилище ключ/значение на основе таблицы ETS	287
10.3.3. Прочие операции ETS	290
10.3.4. Упражнение: реестр процессов	293
Выводы	295
Глава 11. Работа с компонентами	296
11.1. OTP-приложения	296
11.1.1. Создание приложений с помощью инструмента mix	296
11.1.2. Поведение приложения	298
11.1.3. Запуск приложения	299
11.1.4. Библиотечные приложения	300
11.1.5. Создание приложения текущей системы	300
11.1.6. Структура каталогов приложения	302
11.2. Работа с зависимостями	304

11.2.1. Добавление зависимости	305
11.2.2. Реорганизация пула процессов.....	305
11.2.3. Визуализация системы.....	308
11.3. Создание веб-сервера	309
11.3.1. Выбор зависимостей.....	309
11.3.2. Запуск сервера	310
11.3.3. Обработка запросов.....	312
11.3.4. Логика работы системы.....	315
11.4. Настройка приложений	319
11.4.1. Окружение приложения	319
11.4.2. Изменяемость настроек	320
11.4.3. Особенности скриптов конфигурации.....	321
Выводы	322
Глава 12. Создание распределенной системы.....	323
12.1. Примитивы распределенных вычислений.....	325
12.1.1. Запуск кластера.....	325
12.1.2. Взаимодействие узлов.....	326
12.1.3. Обнаружение процессов.....	329
12.1.4. Ссылки и мониторы.....	332
12.1.5. Прочие сервисы распределения	333
12.2. Создание отказоустойчивого кластера.....	335
12.2.1. Устройство кластера	336
12.2.2. Распределенный кеш.....	336
12.2.3. Создание репликационной базы данных	341
12.2.4. Тестирование системы	344
12.2.5. Обнаружение потери связности сети.....	346
12.2.6. Высокодоступные системы	347
12.3. Особенности сетевого соединения	348
12.3.1. Имена узлов	348
12.3.2. Файлы cookie	349
12.3.3. Скрытые узлы	350
12.3.4. Фаерволы.....	350
Выводы	352
Глава 13. Запуск системы	353
13.1. Запуск системы с помощью инструментов Elixir	353
13.1.1. Использование команд mix и elixir	354
13.1.2. Выполнение скриптов.....	355
13.1.3. Компиляция для промышленной эксплуатации.....	356
13.2. OTP-релизы	358
13.2.1. Создание релиза с помощью distillery	358
13.2.2. Использование релиза.....	360
13.2.3. Структура релиза	361
13.3. Анализ поведения системы.....	365
13.3.1. Отладка.....	365
13.3.2. Журналирование.....	367
13.3.3. Взаимодействие с системой.....	367
13.3.4. Трассировка.....	368
Выводы	371