

# Erlang

Erlang – функциональный язык программирования с сильной динамической типизацией, предназначенный для создания распределенных масштабируемых отказоустойчивых систем и управления потоками данных. Обладает рядом особенностей, которые позволяют в короткие сроки создавать надежные функциональные приложения. Язык не предназначен для использования в высокопроизводительных вычислениях, однако прекрасно подходит для быстрой распределенной обработки данных, обработки потоков и создания параллельных приложений. Среди основных особенностей можно перечислить следующие:

- функциональная природа и отсутствие изменяемого состояния, что означает запрет изменения объектов программы (за исключением специальных таблиц);
- гибкий механизм pattern-matching-a, который позволяет компактно представлять сложные условия и проверку соответствия для объектов произвольных типов и структуры;
- удобная система встроенных типов данных (включая атомы, кортежи, битовые строки и бинарные данные, списки, функциональные объекты, ассоциативные массивы), поддержанных на уровне синтаксиса, удобный функционал для работы с этими типами;
- возможность создания анонимных функций и создания функций с использованием замыканий, возможность использования функций как данных, функции высшего порядка, динамическое создание функций;
- мощный механизм обработки потоков данных с помощью проверки соответствия шаблону;
- реализация легковесных процессов, позволяющая создавать сотни тысяч процессов на одном вычислителе, и обеспечение эффективной работы с ними;
- удобная система межпроцессной коммуникации с помощью сообщений, невозможность возникновения конфликтов при параллельной работе процессов;
- уникальная платформа OTP (Open Telecom Platform), позволяющая оперировать такими сущностями, как сервера, супервизоры, приложения и другие, с помощью которых как из строительных кирпичиков можно собирать надежные отказоустойчивые масштабируемые системы.

## Примеры использования:

```
-sh-4.2$ erl
Eshell V5.10.4 (abort with ^G)
1> A = lists:seq(1, 10, 1).
[1,2,3,4,5,6,7,8,9,10]
2> A2 = lists:map(fun(X) -> X * X + 1 end, A).
[2,5,10,17,26,37,50,65,82,101]
3> A3 = [X div 5 + 1 || X <- A2, X > 60].
[14,17,21]
4> A4 = lists:zip(A, A3 ++ lists:duplicate(7, 2)).
[{1,14},{2,17},
 {3,21},
 {4,2},
 {5,2},
 {6,2},
 {7,2},
 {8,2},
 {9,2},
 {10,2}]
5> lists:foldl(fun({X, Y}, Acc) -> Acc + X * Y end, 0, A4).
209
```

## Книги:

- Armstrong J. *Programming Erlang. Software for a Concurrent World.* // 2nd edition, Pragmatic Programmers, 2013.
- Armdtrong J. *Programming Erlang. Software for a Concurrent World.* // Pragmatic Programmers, 2007.
- Hebert F. *Learn You Some Erlang for Great Good! A Beginner' Guide.* // No Star Press, 2013.
- Cesarini F., Thompson S. *Erlang Programming. A Concurrent Approach to Software Development.* // O'Reilly, 2009.
- Laurent S. St. *Introducing Erlang. Getting Started in Functional Programming.* // O'Reilly, 2017.
- Sher G. I. *Handbook of Neuroevolution Through Erlang.* // Springer, 2013.

## Сетевые ресурсы:

- *Erlang Programming Language.* <https://www.erlang.org>
- *Github repository erlang/otp.* <https://github.com/erlang/otp/releases>
- *Learn You Some Erlang for Great Good!* <https://learnyousomeerlang.com/content>

## Консультация по вопросам использования:

[vtasks@jscs.ru](mailto:vtasks@jscs.ru)