

[Top 22 Erlang Interview Questions & Answers](#)

1) What is Erlang?

Erlang is a general purpose programming language and a runtime environment. It has built in support for concurrency, fault tolerance and distribution. Erlang is available as an open source.

2) Explain what is OTP (Open Telecom Platform)?

OTP is an open source platform, and it is a large set of libraries for Erlang to do all from compiling ASN.1 to providing a WWW server.

3) What are the key features of Erlang?

Key features for Erlang is that it has

- Declarative Syntax: It has got declarative syntax
- Concurrent: It has process based model of concurrency
- Real time: It programs real time system
- Continuous Operation: It allows continuous operation
- Robust: For detecting runtime errors it has three constructs in the language
- Memory Management: Memory is allocated automatically and de-allocated when not required
- Distribution: It has got no shared memory, distributed systems can be easily be built in erlang
- Integration: Easy integration of program written in other language

4) Name the datatypes that erlang provides?

Erlang provides two datatypes

- Constant data types: These types of data types cannot be split into more primitive subtypes. It consists of Numbers and Atoms.
- Compound data types: These types of data types are used to group together other types of data types. It consists of tuples and lists.

5) What is the command to disconnect node in Erlang?

By calling erlang: disconnect_node (Node) will force disconnection of the node.



6) Explain what is modules in Erlang?

Erlang enables to organize code into modules. A module consists of functions; it introduces a local scope of functions (Public and Private).

7) Mention how module is stored in Erlang?

A module name is stored in a file named .erl. File basename and module name must be the same.

8) Explain what is Bit strings in Erlang?

To store an area of untyped memory, a bit string is used and are expressed using the bit syntax.

9) Explain what is Pattern Matching?

A pattern matching occurs when evaluating a function call. Through pattern matching mechanism, variables are bound to the values.

10) Explain what is PID datatype?

PID mean Process Identifier and is the datatype used for the unique process identifiers which are assigned to every process.

11) To include a file in Erlang what is the command used?

In two ways, a file can be included in Erlang

- include (File).
- include_lib (File).

12) How run-time error is handled in Erlang?

It is possible to prevent run-time errors and other exceptions from causing the process to terminate by using Catch or Try. Catch Expr returns the value of expression unless an exception occurs during the evaluation. Try Exprs is the enhancement of catch with ability to identify the exception class and choose to handle the desired one.

13) In Erlang how "process" is created?

The process is created by calling spawn, and spawn forms a new process and returns the pid.

- Spawn (Module, Name, Args) -> pid ()

14) Define guard sequence and what are the valid guard expressions?

Erlang function clauses can be protected with guards; a clause can be protected only if the guard holds. It is a sequence of guards, separated by semicolon (;), and a guard is a sequence of guard expressions separated by comma (.). The guard sequence is true if at least one of the guards is true.

15) Give the list of valid guard expressions?

Valid guard expressions are

- Atom true,
- Other constants
- Calls to the BIFs specified
- Term Comparisons
- Arithmetic Expressions
- Boolean Expressions
- Short-circuit Expressions

16) Explain how message is send and receive in Erlang?

For sending message Erlang uses the exclamation mark (!) as the operator for sending a message. The syntax used for sending a message is Pid ! Message.

For receiving message Erlang uses Pattern Matching, to deliver a message from the message queue receive statement is used.

17) Explain what is Spawn/ 113 and Spawn_link 113?

Spawn/ 113: It creates a new process and return its pid. In system scheduler queue, the new process is placed so that it runs some time later.

Spawn_link/113: It provides the same functionality as spawn/113 but with the addition of a link that is atomically created between the newly spawned process and the caller.

18) Mention what is the syntax to write Macros in Erlang?

The syntax used to write Macros in Erlang is

- define (Const, Replacement).
- define (Fun (Var1 , Var2, ..., Var) , Replacement)

19) List out the pre-defined Macros in Erlang?

The pre-defined Macros in Erlang is

- ? Module: It returns the name of the current module
- ? Module_String: It returns the name of the current module, as a string
- ? File: It returns the current file name
- ? Line: It returns the current line number
- ? Machine: It returns the current machine name, 'BEAM.'

20) Explain what is a record and how you can define records in Erlang?

For storing a fixed number of elements, a data structure is used which is referred as record. Expressions are translated into tuple expressions during compilation record.

Record in Erlang is defined by the name of the record, followed by the field names of the record. Record and field names must be atoms.

- record (Name, { Field1 [= Value] , ... FieldN [= ValueN] })

21) Mention what is the command used for accessing record field in Erlang?

Expr#Name.Field, this command will return the value of the mentioned field while for returning the position of the specified field #Name.Field is used.

22) Explain what is epmd?

epmd is a small name server which is used by Erlang programs when establishing distributed Erlang communications. It is also referred as an Erlang Port Mapper Daemon. It is responsible for mapping the symbolic node names to machine addresses it is used.

[Guru99](#) Provides [FREE ONLINE TUTORIAL](#) on Various courses like

Java	MIS	MongoDB	BigData	Cassandra
Web Services	SQLite	JSP	Informatica	Accounting
SAP Training	Python	Excel	ASP Net	HBase
Project Management	Test Management	Business Analyst	Ethical Hacking	PMP
Live Project	SoapUI	Photoshop	Manual Testing	Mobile Testing
Data Warehouse	R Tutorial	Tableau	DevOps	AWS
Jenkins	Agile Testing	RPA	JUnit	Software Engineering
Selenium	CCNA	AngularJS	NodeJS	PLSQL

**Stay updated with new
courses at Guru99
Join our Newsletter**