Cloud Haskell in Cloud Computing

1. Introduction

1. Haskell is a purely functional programming language.
2. Cloud Haskell is a framework built on Haskell for distributed computing.
3. It is inspired by the Erlang model of concurrency and distribution.

Distributed Computing

- **Definition:** Distributed computing means dividing a large task into smaller tasks and running them on multiple computers (nodes) that work together as one system.
- Each computer shares resources (CPU, memory, storage) to complete the job faster.
- It is used in cloud computing, big data, and high-performance systems.
  - Example: Google Search uses thousands of computers together to give results quickly.

What is Haskell?

- **Haskell** is a **functional programming language**.
- It is named after the mathematician **Haskell Curry**.
- It is based on **mathematical functions** instead of step-by-step instructions.
- It is known for:
  1. **Purity** – No side effects, functions always give the same output for the same input.
  2. **Lazy evaluation** – Code is only executed when needed.
  3. **Strong type system** – Catches errors at compile time.
  4. **Concurrency and parallelism** – Can run multiple tasks at the same time.
- Used in **academics, research, compilers, financial systems, and distributed computing**.

2. Why Cloud Haskell in Cloud Computing?

4. **Scalability** – Programs can run across many cloud machines.
5. **Reliability** – Even if one process fails, others continue running.
6. **Concurrency** – Supports multiple tasks at the same time.
7. **Performance** – Efficient handling of parallel operations.

3. Features of Cloud Haskell

8. Provides lightweight processes that consume fewer resources.
9. Uses message passing for process communication.
10. Avoids shared memory, reducing errors and race conditions.
11. Supports distributed execution across cloud servers.
12. Offers failure detection to identify system crashes.
13. Provides fault tolerance to keep the system running after errors.
14. Ensures location transparency (process can run anywhere).
15. Abstracts low-level networking details from developers.
16. Designed for high-performance cloud applications.

4.Applications in Cloud Computing

17. Used in distributed databases for safe concurrent updates.
18. Helps build scalable web applications.
19. Supports IoT systems managing multiple devices.
20. Useful for big data analytics and processing.
21. Builds reliable cloud services.
22. Suitable for banking systems needing high fault tolerance.
23. Used in telecom systems to handle millions of users.
24. Good for real-time messaging systems.
25. Supports parallel scientific computations.

5. Advantages of Cloud Haskell

26. Simplifies parallel and distributed programming.
27. Provides safe communication between processes.
28. Increases reliability of cloud applications.
29. Makes systems fault-tolerant.
30. Supports both small and large-scale applications.
31. Reduces complexity of distributed computing.
32. Encourages modular program design.
33. Provides high concurrency without heavy resource use.
34. Enables developers to focus on logic instead of networking.

35. Scales easily with cloud infrastructure.