**Computer Programming Paradigms**

**What does the term “Computer Programming Paradigm” mean**: A computer programming that dictates how code is structured and executed. It provides a framework for organizing logic and data, guiding how programmers think about solving problems and writing software.

**List three computer programming paradigms and explain each in detail:**

1. **Procedural Programming:** This paradigm focuses on a step-by-step approach, organizing code into procedures or functions. It emphasizes sequential execution and control flow through loops and conditionals. Languages like C and Pascal follow this approach.
2. **Object-Oriented Programming (OOP)**: OOP organizes programs around objects that combine data and behavior. Key concepts include encapsulation, inheritance and polymorphism. It promotes reusability and modularity. Java, Python and C++ are common OOP languages.
3. **Functional Programming:** This paradigm treats computation as the evaluation of mathematical functions, avoiding mutable data and side effects. It emphasizes immutability and pure functions, with languages like Haskell and Lisp being key examples.

**List three advantages and three disadvantages of the Object-Oriented Programming paradigm:**

**Advantages of Object-Oriented Programming (OOP):**

* **Code Reusability:** Through inheritance and polymorphism, existing code can be reused in new applications, reducing duplication.
* **Modularity:** Encapsulation allows dividing complex programs into smaller, manageable objects, making code easier to understand and maintain.
* **Scalability**: OOP simplifies the design of large systems by allowing modular expansion and easier collaboration between teams.

**Disadvantages of Object-Oriented Programming (OOP):**

* **Increased Complexity:** OOP can introduce complexity, especially in small programs where using objects and classes might be too much for it to handle.
* **Slower Performance:** Object-oriented designs, with features like dynamic method resolution and inheritance, can lead to slower performance compared to procedural code.
* **Memory Overhead:** OOP often uses more memory due to object creation, abstraction layers and additional features like polymorphism.

**Why would a global software company, with offices in all the major cities of each continent on the planet, decide to use the OOP paradigm to develop its software:** A global software company would use OOP for its modularity, scalability and ease of collaboration. OOP allows teams in different locations to work on separate components, reuse code and maintain complex systems run efficiently making it ideal for large, distributed projects.