Advantages and Disadvantages of Object-Oriented Programming (OOP)

This reading discusses advantages and disadvantages of object-oriented programming, which is a well-adopted programming style that uses interacting objects to model and solve complex programming tasks. Two examples of popular object-oriented programming languages are Java and C++. Some other well-known object-oriented programming languages include Objective C, Perl, Python, Javascript, Simula, Modula, Ada, Smalltalk, and the Common Lisp Object Standard.

Some of the advantages of object-oriented programming include:

1. Improved software-development productivity: Object-oriented programming is modular, as it provides separation of duties in object-based program development. It is also extensible, as objects can be extended to include new attributes and behaviors. Objects can also be reused within an across applications. Because of these three factors – modularity, extensibility, and reusability – object-oriented programming provides improved software-development productivity over traditional procedure-based programming techniques.

2. Improved software maintainability: For the reasons mentioned above, object-oriented software is also easier to maintain. Since the design is modular, part of the system can be updated in case of issues without a need to make large-scale changes.

3. Faster development: Reuse enables faster development. Object-oriented programming languages come with rich libraries of objects, and code developed during projects is also reusable in future projects.

4. Lower cost of development: The reuse of software also lowers the cost of development. Typically, more effort is put into the object-oriented analysis and design, which lowers the overall cost of development.

5. Higher-quality software: Faster development of software and lower cost of development allows more time and resources to be used in the verification of the software. Although quality is dependent upon the experience of the teams, object-oriented programming tends to result in higher-quality software.

Some of the disadvantages of object-oriented programming include:

1. Steep learning curve: The thought process involved in object-oriented programming may not be natural for some people, and it can take time to get used to it. It is complex to create programs based on interaction of objects. Some of the key programming techniques, such as inheritance and polymorphism, can be challenging to comprehend initially.

2. Larger program size: Object-oriented programs typically involve more lines of code than procedural programs.

3. Slower programs: Object-oriented programs are typically slower than procedure-based programs, as they typically require more instructions to be executed.

4. Not suitable for all types of problems: There are problems that lend themselves well to functional-programming style, logic-programming style, or procedure-based programming style, and applying object-oriented programming in those situations will not result in efficient programs.