ADVANTAGES AND DISADVANTAGES OF INHERITANCE

Advantages

1. Reduce code redundancy.
2. Efficient code re-usability.
    a. As the existing code is reused, it leads to less development and maintenance costs.
    b. When a class inheritance or derives another class, it can access all the functionality from the inherited class.
    c. Reusability enhanced reliability. The base class code will be already tested and debugged.
3. Reduces source code size and improves code readability.
    a. Application takes less memory.
4. Supports application code extensible by overriding the base class functionality within child classes.
5. Data hiding: Base class (Abstract super classes) can decide to keep some data private so that it cannot be altered by derived classes.
6. Using inheritance, we can organize the information in the hierarchal form.

Disadvantages

1. Inherited functions work slower than normal functions as there is an indirection.
2. All unused data members are loaded into memory. It may impact application performance.
3. Improper use of inheritance may lead to wrong solutions
4. Often, data members in the base class are left unused which may lead to memory wastage.
5. Inheritance increase the coupling between base class and derived class. A change in the base class will affect all the child classes. I.e. In Inheritance, base class and child classes are tightly coupled.
    a. Refactor: If a method is deleted in the super-class or aggregate then we have to re-factor in case of using that method in all child classes.
6. Inheritance (throw classes, not through interfaces) will expose the implementation details thus violating encapsulation.
7. Inherited relationships generally cannot be altered at runtime. So, it is very hard for UNIT testing.

Scenario

Suppose that a student has attributes like name, age, and location. This class exists for a long time. After some time, we need to add one attribute pan card of the student. So, what will we have all the below options?

Option-1. The first option is that we can modify the class student.

Option-2. The second option is we need to extend student class and add an attribute in the new class.

Explanation

- The first option is the violation of principles of java 'Java classes are not open for modifications. So, we should not modify the existing one. This will increase unit testing to be done again for the existing class.
  - "**Closed for modification Open for Extension**".
- The second option is without disturbing the existing class, we can add one variable to the other class by making a subclass. So only unit testing will need for the child class, not for the parent class.