In Java, there are several types of inheritance, each serving a specific purpose in class hierarchy design. The main types of inheritance are:

1. **Single Inheritance**: A subclass inherits from a single superclass. This is the simplest form of inheritance.

   ```java
   class Superclass {
       void display() {
           System.out.println("This is the superclass.");
       }
   }

   class Subclass extends Superclass {
       // Subclass inherits from Superclass
   }
   ```

2. **Multilevel Inheritance**: In this type of inheritance, a class is derived from another class, which is also a subclass of some other class.

   ```java
   class Grandparent {
       void display() {
           System.out.println("This is the grandparent.");
       }
   }

   class Parent extends Grandparent {
       // Parent inherits from Grandparent
   }

   class Child extends Parent {
       // Child inherits from Parent
   }
   ```

3. **Hierarchical Inheritance**: In this type, multiple classes inherit from a single superclass.

   ```java
   class Superclass {
       void display() {
           System.out.println("This is the superclass.");
       }
   ```

```
}

class Subclass1 extends Superclass {
    // Subclass1 inherits from Superclass
}

class Subclass2 extends Superclass {
    // Subclass2 inherits from Superclass
}
```

4. **Multiple Inheritance (through Interfaces)**: Java supports multiple inheritance of interfaces. A class can implement multiple interfaces, effectively inheriting method signatures from multiple sources.

```java
interface Interface1 {
    void method1();
}

interface Interface2 {
    void method2();
}

class MyClass implements Interface1, Interface2 {
    public void method1() {
        // Implement method1
    }

    public void method2() {
        // Implement method2
    }
}
```

5. **Hybrid Inheritance (Combination of Multiple Types)**: Hybrid inheritance is a combination of any of the above types. It's essential to design carefully when using hybrid inheritance to avoid complications.

Remember that Java doesn't support true multiple inheritance for classes (i.e., inheriting from multiple classes). If you want to achieve multiple inheritance-like behavior, you can use interfaces, as shown in the "Multiple Inheritance (through Interfaces)" example.

Keep in mind that while inheritance is a powerful concept, it should be used judiciously to maintain code clarity and avoid creating complex and tightly coupled class hierarchies.