

1. Principle Violated: Interface Segregation Principle
2. Reason: Interface `IInsect` contains 5 methods. By implementing this interface, class `FlyingInsect` had to implement the method `Swim()` which it doesn't need. Similarly, class `AquaticInsect` had to implement the method `Fly()` even though it is not required for that class. To fix this violation, I have created three specific interfaces instead of one generic interface. Interface `IFlyInsect` contains methods which are specific to flying insects. Interface `IAquaInsect` contains methods which are specific to aquatic insects. Interface `IInsect` contains methods which are common to both types. Class `FlyingInsect` implements interfaces `IFlyInsect` and `IInsect` while class `AquaticInsect` implements interfaces `IAquaInsect` and `IInsect`.
3. The following code fixes the violation of Interface Segregation Principle.  
`IFlyInsect.java`  
`IAquaInsect.java`  
`IInsect.java`  
`FlyingInsect.java`  
`AquaticInsect.java`