If you have to define a class Derived which derives from Base class, if B has a data member function mf() which is no virtual then we should never have the same function mf() defined in Derived class because

1. Everything that applies to B objects also applies to D objects, because every D object is-a B object;
2. Classes derived from B must inherit both the interface *and* the implementation of mf, because mf is non-virtual in B.

Now, if D redefines mf, there is a contradiction in your design . Its is not a complete “is-a” relationship then