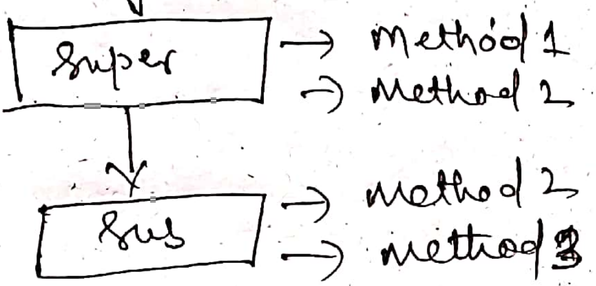
Dynamic Method Dispatch:

*) It is Runtime polymorphisim.

*) It is the mechanism by which a call to an overriden method is Resolved at Run time.

*) when an overriden Method is called thorough a superclass reference, java determines which version (super/sub class) of that method is to be executed based upon the type of the object being Reffered to at the time of call.



Super obj = new Sub() Allowed

obj. method2() → Allowed

obj. method3() – not allowed.

e.g.

```
class xyz
{  void method1()
     {  s.o.p("method1");
     }
   void method2()
     {  s.o.p("method2 of Base");
     }
}
class ABC extends xyz
{  void method2()
     {  s.o.p("method 2 of child");
     }
   void method3()
     {  s.o.p("method 3");
     }
p.s.v.m()
{  xyz ob = new ABC();
     ob.method1(); → Base class
     ob.method2(); → child class
     ob.method3(); → X Not allowed
}  }
```

O/P

```
Method 1.
method 2 of child
```

*) Dynamic method dispatch is creating the object of derived class with the Reference of Base class.