

# Traits - implementation of pattern

<http://www.agiledeveloper.com> (<http://www.agiledeveloper.com>)

In [1]: *//Implement a method in interface by using default method*

In [2]: *//when interface implements then dif between abstract base class n interf*  
*//abstract base class can have state, feels, but interface cant*  
*//factory pattern*

In [6]: **trait** Friend{ *//automatically create an abstract class n an interface in*  
     **val** name:String *//not initializing it as it is in trait*  
     **def** listen = **println**(name+" listening")  
 }  
**println**("Ok")

Ok

defined trait Friend

In [7]: **class** Human(**val** name:String) **extends** Friend  
     **val** sam= **new** Human("Sam")  
     sam.listen

Sam listening

defined class Human

sam: \$user.Human = cmd6\$\$user\$Human@3e4f6846

In [8]: **def** help(friend:Friend) = friend.listen  
     help(sam)

Sam listening

defined function help

In [ ]: