#===============================================================================

# Skill Self Effects

# By Jet10985 (Jet)

# Requested by Touchfuzzy

#===============================================================================

# This script will allow you to specify a skill's effects tot arget the user

# instead of the target, on skills that don't already effect the user.

# This script has: 0 customization options.

#===============================================================================

# Overwritten Methods:

# None

#-------------------------------------------------------------------------------

# Aliased methods:

# Game\_Battler: item\_effect\_apply

#===============================================================================

=begin

To specify the effects, use this notetag in the skill's notebox:

<self effect: 1>

or

<self effect: 1, 2, 3> to specify more than 1 effect

--------------------------------------------------------------------------------

Use of this script could be as the following.

You make a "Berserk Strike" skill, which does 200% damage to an enemy.

You make the 2nd effect "Defense Down 50%" and want it applied to the user.

You'd use this notetag: <self effect: 2>

=end

class RPG::Skill

def self\_effects

if @self\_effects.nil?

@self\_effects = []

self.note.each\_line {|a|

scan = a.scan(/<self[ ]\*effect[ ]\*\:[ ]\*(\d+(?:[ ]\*,[ ]\*\d+)\*)>/i)

begin

scan[0][0].scan(/\d+/).each {|b|

@self\_effects.push(@effects[b.to\_i - 1]) if b.to\_i >= 1

}

rescue Exception => e

end

}

end

@self\_effects

end

end

class Game\_Battler

alias jet3745\_item\_effect\_apply item\_effect\_apply

def item\_effect\_apply(user, item, effect, reffed = false)

if item.is\_a?(RPG::Skill) && item.self\_effects.include?(effect) && !reffed

user.item\_effect\_apply(user, item, effect, true)

return

end

jet3745\_item\_effect\_apply(user, item, effect)

end

end