I'm not robot

reCAPTCHA

Continue

Demonic gorilla osrs plugin

2018 Demon Gorilla Plug-in '...@@@@@@@@@@@@@@@@@@@@113,6+113,12@@@@@@@@@@@@-113,6+113,6+LOOKING_INTO 113,6+113,@@@@832; public static final indig= 830; final VENGEANCE_OTHER 4. Public static final int DEMONIC_GORILLA_MAGIC_ATTACK = 7225; public static final int DEMONIC_GORILLA_MELEE_ATTACK = 7226; public static final int DEMONIC_GORILLA_RANGED_ATTACK = 7227; public static final int DEMONIC_GORILLA_AOE_ATTACK = 7228; public static final int DEMONIC_GORILLA_PRAYER_SWITCH = 7228; public static final int DEMONIC_GORILLA_DEFEND = 7224; NPC animation release static final TZTOK_JAD_MAGIC_ATTACK = 2656; .... WINTERTODT_SNOW_FALL_AOE... Public static final int DEMONIC_GORILLA_RANGED = 1302; Public static final int DEMONIC_GORILLA_MAGIC = 1304; Public static final int DEMONIC_GORILLA_BOULDER = 856; /** * Missing: Marble gargoyle, excellent dark beast */

package demon.client.clients. Get Java.util.Array. Get Java.util.list; Imported Lombok. Getter; Imported Lombok. Setter; import net.runelite.api.actor; import net.runelite.api.NPC; import net.runelite.api.NPCComposition; Earnings net.runelite.api.coords.WorldArea; Public class DemonicGorilla {static final int MAX_ATTACK_RANGE = 10; Need >= 10= tiles= to reach= target= static= final= int= attack_rate=5, 5= ticks= between= each= attack= static= final= int= attacks_per_switch=3= 3= unsuccessful= attacks= per= style= switch= static= int= projectile_magic_speed=8; travels= 8 tiles= per= static tick= final= final= final= projectile_ranged_speed= projectile_ranged_speed travels= 6= tiles= per= tick= static= final= int= projectile_magic_delay=12; requires= an= extra= 12= tiles= static= final= int= projectile_ranged_delay=9; requires= an= extra= 9= tiles= public= static= all_regular_attack_styles==attackstyle.melee,= attackstyle.ranged,= attackstyle.magic= ==enum= attackstyle= {=magic,= ranged,= melee,= boulder= @getter= @getter= private= npc=npc;= @getter= @setter= private=>lt;==<AttackStyle>next PosibleAttackStyles; @Getter @Setter personal attack toswitch; @Getter @Setter personal int nextAttackTick; @Getter @Setter personal int last TickAnimation; @Getter @Setter personal world area last TickAnimation; @Getter @Setter personal boolean changeAttackStyleThisTick; @Getter @Setter personal boolean changeAttackStyleLastTick; @Getter @Setter personal boolean battle begins; @Getter @Setter personal actors last tick interaction; @Getter @Setter personal boolean shooting damage; @Getter @Setter personal @Getter @Setter @Getter @Setter personal int last tick overhead icon; @Getter @Setter personal int disabled MeleeMovementForTicks; Public Demon Gorilla (NPC NPC NPC) { this.npc = NPC; this.nextPosibleAttackStyles = Array.asList (ALL_REGULAR_ATTACK_STYLES); this.nextAttackTick = -100; this.attacksUntilSwitch = ATTACKS_PER_SWITCH; this.last-lastTickOverhead = -1; this.lastTickOverheadIcon = -1; ≤Public int getoverid (NPC configuration= npc= this.npc.getComposition) (ifstyle]} Return -1; } ≤ No renewal at the end of the file

package net.runelite.api.Player; import net.runelite.api.Point; Earnings net.runelite.api.Skill; import net.runelite.api.coords.LocalPoint; Get net.runelite.client.game.SkillIconManager. import net.runelite.client.ui.overlay.overlay. net.runelite.client.ui.overlay.Overlay.Overlay.Overlay. import net.runelite.client.ui.overlay.Overlay. Public class DemonicGorilla Overlay { Personal Static Final Color COLOR_ICON_BACKGROUND = New Colors (0, 0, 0, 128); Personal Static Final Colors (0, 0, 0, 255); Personal Static Final Color COLOR_ICON_BORDER = New Colors (0, 0, 0, 255); Personal Static Final Color = COLOR_ICON_BORDER_FILL = New Colors (219, 175, 0, 0, 0, 255); Personal Static Final int = OVERLAY_ICON_DISTANCE 50; Personal Static End int OVERLAY_ICON_MARGIN = 8; Private Client Client; Personal Demogora Plug-in; @Inject Private SkillIconManager Icon Manager; @Inject Public Demogora Overlay (Client, Demon Gorilla Plug-in) {set (overlay position.dynamic); setLayer (OverlayLayer.ABOVE_SCENE; client; this.= plug-in= plug-in= personal bufferimage(geticon;{ Case Proximity: Return iconManager.getSkillImage (Skill.ATTACK); Case RANGED: Return iconManager.getSkillImage (Skill.RANGED); Case Magic: Return iconManager.getSkillImage (Skill.MAGIC); } null return; @Override public dimension rendering (Graphics2D graphics) {player = client.getLocalPlayer (); for (demon gorilla: plugin.getGorillas())) {gorilla.getNpc().getInteracting ()==null) {continue; local point lp = gorilla.getNpc ().getLocalLocation(); (lp!= null) { Point = Perspective.worldToCanvas (client, lp.getX (), lp.getY (), client.getPlane (), gorilla.getNpc ().getLogicalHeight () +16); if (point != null) { list <DemonicGorilla.AttackStyle> Attack Style = Gorilla.getNextPosibleAttackStyles (); List<BufferedImage> Icon = New ArrayList >lt;> (); int totalWidth = (attackStyles.size)- 1) * OVERLAY_ICON_MARGIN; (demonicGorilla.AttackStyle Attack Style: Attack Style) { Buffering image icon = getIcon (attackStyle); icons.add (icon); full width += icon.getWidth (); } int bgpadding = 4; int currentPosX = 0; (Buffer image icon: icon) | graphics.setStroke (new basic strokes (2);graphics.setColor (COLOR_ICON_BACKGROUND); graphics.fillOval (point.getX) - Total width / 2 + current PosX - bgPadding, point.getY () - icon.getHeight () / 2 - OVERLAY_ICON_DISTANCE - bgPadding, icon.getWidth () + bgPheight *2, icon.getHeight () + bgPadding *2); graphics.setColor (COLOR_ICON_BORDER); graphics.drawOval (point.getX)- Total width / 2 + current posX - bgPadding, point.getY () - icon.getHeight () / 2 - OVERLAY_ICON_DISTANCE - bgPadding* 2.bg.adding.getP+) graphics.drawImage (icon, point.getX ( ) - totalWidth / 2 + currentPosX, point.getY () - icon.getHeight () / 2 - OVERLAY_ICON_DISTANCE - bgPadding, icon.getWidth (COLOR_ICON_BORDER_FILL OVERLAY_ICON_DISTANCE) Arc2D.Double arc = new Arc2D.Double (pointX) - Total width / 2 + currentPosX - bgPadding, point.getY () - icon.getHeight () / 2 - OVERLAY_ICON_DISTANCE - bgPadding, icon.getWidth () + bgPadding * 2, icon.getHeight () + bgPadding * 2, 90.0, -360.0 * (DemonicGorilla.ATTACKS_PER_SWITCH - gorilla.getAttacksUpSwitch (switch)) / DemonicGorilla.ATTACKS_PER_SWITCH, Arc2D.OPEN); Graphics.draw (arc); CurrentPosX += icon.getWidth () + OVERLAY_ICON_MARGIN; } } return null; } } No renewal at the end of the file

package demon.client.clients. Get Java.util.ArrayList; Get Java.util.list; Imported Lombok. Getter; Imported Lombok. Setter; Earnings net.runelite.api.Hitsplat; get net.runelite.api.Player; Earnings net.runelite.api.coords.WorldArea; Public Class Memorization Player (Player) {e.player = Player; This.Recent Hit Flat = New ArrayList >lt;> ()} } } New line at the end of the file

package net.runelite.clients.demonic; Imported Lombok. Getter; get net.runelite.api.Player; Public class hold gorilla attack { @Getter personal demon gorilla attacker; @Getter private demon gorilla. AttackStyle>/https:>/Hitsplat>target sit-@Getter individual players; @Getter personal int completeOnTick; Openly hold gorilla attack (evil gorilla attacker, demon gorilla. AttackStyle Attack Style, Player Target, Int CompleteOnTick) { This.Attacker = Attacker; E.attackStyle = AttackStyle; This.Target = Target; This.finishesOnTick = CompleteOnTick; } } \ There is no latest news at the end of the file file.