function buff()

{

Orion.InfoBuff();

}

function Seed()

{

Orion.ClearJournal();

var seeds = Orion.FindType(0x1727,'-1',backpack);

for (var c = 0; c < seeds.length; c++)

var seed = seeds[c]

var L = 1;

while(L == 1)

{

Orion.Wait(150)

if (Player.Hits() < 35 && Orion.ObjectExists(seeds))

{

Orion.Wait(150)

Orion.UseType(0x1727,'-1',backpack);

}

var Seeding = Orion.InJournal('The bitter seed instantly restores some of your health!');

if(Seeding)

{

Orion.Wait(60000);

Orion.ClearJournal();}

}}

function StrengthAgility()

{

var str = Orion.FindType('0x0f09','0x0000',backpack)

var agil = Orion.FindType('0x0f08','any',backpack)

while (!Player.Dead()){

Orion.Wait(450)

while (Orion.ObjAtLayer('LeftHand'))

Orion.Wait(450)

while (Orion.ObjAtLayer('LeftHand') && Orion.ObjAtLayer('RightHand'))

Orion.Wait(450)

if (!Orion.BuffExists('0x7567'))

{

Orion.Wait(650)

Orion.UseItemOnMobile(str[0],str[0]);

Orion.Wait(150)

}

if (!Orion.BuffExists('0x753C'))

{

Orion.Wait(650)

Orion.UseItemOnMobile(agil[0],agil[0])

Orion.Wait(150)

Orion.CancelWaitTarget()}

}

}

function StrengthAgility2()

{

var str = Orion.FindType('0x0f09','0x0000',backpack)

var agil = Orion.FindType('0x0f08','any',backpack)

while (!Player.Dead()){

Orion.Wait(450)

while (Orion.ObjAtLayer('RightHand'))

Orion.Wait(450)

while (Orion.ObjAtLayer('LeftHand') && Orion.ObjAtLayer('RightHand'))

Orion.Wait(450)

if (!Orion.BuffExists('0x7567'))

{

Orion.Wait(650)

Orion.UseItemOnMobile(str[0],str[0]);

Orion.Wait(150)

}

if (!Orion.BuffExists('0x753C'))

{

Orion.Wait(650)

Orion.UseItemOnMobile(agil[0],agil[0])

Orion.Wait(150)

Orion.CancelWaitTarget()}

}

}

function Petals()

{

var petal = Orion.FindType('0x1021', 'any', backpack)

Orion.UseItemOnMobile(petal[0],petal[0])

while (true){

Orion.Wait(750)

while ((!Orion.WaitWhileTargeting()) && (!Player.Frozen()))

Orion.Wait(350)

if (!Orion.BuffExists('0x75FC'))

{

Orion.Wait(7000)

Orion.UseItemOnMobile(petal[0],petal[0])

Orion.Wait(2000);}

}

}

function TrapBox()

{

Orion.ClearJournal();

var boxes = Orion.FindType('0x09A9', 'any', 'backpack');

for (var c = 0; c < boxes.length; c++)

var box = boxes[c]

var L = 1;

while(L == 1)

{

Orion.Wait(100)

if (Orion.BuffExists('Paralyzed') || Orion.BuffExists('Paralyze'))

{

Orion.Wait(10);

Orion.UseItemOnMobile(box, box);

Orion.Wait(300);

Orion.Hide(Orion.GetSerial(box))

Orion.Wait(150)

Orion.Unhide(Orion.GetSerial(box))

}

var NerveStriked = Orion.InJournal('Your attacker dealt a crippling nerve strike!');

if(NerveStriked)

{

Orion.Wait(500);

Orion.UseItemOnMobile(box, box);

Orion.Wait(500);

Orion.Hide(Orion.GetSerial(box));

Orion.Unhide(Orion.GetSerial(box))

Orion.ClearJournal();}

}

}

function PopHorse()

{

Orion.UseType('0x2124');

Orion.Wait(100)

if (Orion.WaitForGump(1000))

{

var gump0 = Orion.GetGump('last');

if ((gump0 !== null) && (!gump0.Replayed()) && (gump0.ID() === '0x00002340'))

{

Orion.Wait(200)

gump0.Select(Orion.CreateGumpHook(7));

Orion.Wait(200) ;

}

}

Orion.WarMode(0);

var Timer =Orion.Now();

while (Orion.ObjAtLayer('Mount') == null){

Orion.Say('all follow me');

Orion.Wait(100);

Orion.UseFromGroundList('Horses','1','nothuman|near|live|');

if(Orion.Now()>=Timer + 2000)

break; }

}

function TeleTenTile2()

{

var value = 10;

var empty = 0;

var offset =

[

[empty, -value], //direction = 0

[value, -value], //direction = 1

[value, empty], //direction = 2

[value, value], //direction = 3

[empty, value], //direction = 4

[-value, value], //direction = 5

[-value, empty], //direction = 6

[-value, -value] //direction = 7

];

var xy = offset[Player.Direction() & 7];

Orion.WaitTargetTileRelative('0', xy[0], xy[1], Player.Z());

Orion.UseType('TeleScroll')

}

function TeleTenTile()

{

var value = 10;

var empty = 0;

var offset =

[

[empty, -value], //direction = 0

[value, -value], //direction = 1

[value, empty], //direction = 2

[value, value], //direction = 3

[empty, value], //direction = 4

[-value, value], //direction = 5

[-value, empty], //direction = 6

[-value, -value] //direction = 7

];

var xy = offset[Player.Direction() & 7];

Orion.TargetTileRelative('0', xy[0], xy[1], Player.Z());

Orion.WaitTargetTileRelative('0', xy[0], xy[1], Player.Z());

Orion.UseType('TeleScroll')

Orion.Cast('Teleport');

}

function TargetBehindMe()

{

var value = -1;

var empty = 0;

var offset =

[

[empty, -value], //direction = 0

[value, -value], //direction = 1

[value, empty], //direction = 2

[value, value], //direction = 3

[empty, value], //direction = 4

[-value, value], //direction = 5

[-value, empty], //direction = 6

[-value, -value] //direction = 7

];

var xy = offset[Player.Direction() & 7];

Orion.TargetTileRelative('0', xy[0], xy[1], Player.Z());

}

function AddMount()

{

Orion.AddObject('myMount');

Orion.Print('-1', 'Target your mount')

}

function MountAndDismount()

{

if(!Orion.ObjAtLayer('Mount'))

{

if(!Orion.FindObject('myMount'))

AddMount();

else

Orion.UseObject('myMount');

}

else

Orion.UseObject('self')

}

function FindNearestEnemy()

{

Orion.Ignore('self');

var friends = Orion.GetFriendList();

for(var i = 0; i < friends.length; i++) {

Orion.Ignore(friends[i]);

}

var enemy = Orion.FindType('0xFFFFFF', '-1', ground, 'near|mobile|live', '18', 'orange');

if ( !enemy.length ) {

// Orion.Ignore('self');

// var friends = Orion.GetFriendList();

// for(var i = 0; i < friends.length; i++) {

// Orion.Ignore(friends[i]);

// }

// enemy = Orion.FindType('0xFFFFFF', '-1', ground, 'near|mobile', '18', 'gray|criminal|red|orange|blue');

// if ( !enemy.length ) {

Orion.Print( '-1', 'no enemy found, resetting ignore list');

Orion.IgnoreReset();

return '';

// }

}

return enemy[0];

}

function TargetNextEnemy()

{

var serial = FindNearestEnemy();

Orion.GetEnemiesStatus();

if(!serial.length) return;

var enemy = Orion.FindObject(serial);

var notoColor;

switch (enemy.Notoriety() ) {

case 1: notoColor = 2119;

break;

case 6: notoColor = 33;

break;

default: notoColor = 48;

}

Orion.CharPrint(self, notoColor, enemy.Name() );

Orion.RemoveHighlightCharacter(lasttarget, true);

//Orion.CharPrint(enemy.Serial(), '48', enemy.Name());

Orion.AddHighlightCharacter(enemy.Serial(), '1152', true);

Orion.ClientLastTarget(serial);

Orion.TargetSystemSerial(serial);

Orion.Ignore(serial);

}

function pullbar()

{

if (lasttarget)

Orion.ShowStatusbar(lasttarget, 980, 720);

}

function closeStatusbar()

{

Orion.CloseStatusbar(Orion.ClientLastTarget());

}

function closeStatusbar()

{

Orion.CloseStatusbar(Orion.ClientLastTarget());

}

function SecondaryAbility()

{

Orion.UseAbility('Secondary', 'state');

Orion.Attack(lasttarget)

}

function PrimaryAbility()

{

Orion.UseAbility('Primary', 'state');

}

function OpenBackpack()

{

Orion.OpenContainer('backpack');

}

function OpenPaperdoll()

{

Orion.OpenPaperdoll('self');

}

function OpenJournal()

{

Orion.ShowJournal();

}

function FastRecall()

{

if (Orion.GetSerial('runebook') == '0x00000000')

{

Orion.AddObject('runebook');

Orion.Print('0x0123', 'Choose your runebook/rune for fast recall.');

Orion.Wait(200);

while (Orion.HaveTarget())

Orion.Wait(1000);

Orion.Print('0x0123', 'Bind is added to Lists->Objects.');

}

Orion.Cast('Recall');

Orion.WaitTargetObject('runebook');

}

function Apple()

{

var apple = Orion.FindType('0x2fd8', 'any', backpack)

Orion.UseItemOnMobile(apple[0], apple[0])

Orion.AddDisplayTimer('Apple', 30000, 'Right', 'Circle|Bar', 'Apple', 0, 0, '33', 0xff, '0xFFFFFF')

}

function TargetLowHitFrendly(){

Orion.IgnoreReset();

Orion.Ignore(self);

var mobileID = Orion.FindType("-1", "-1", "ground", "human|live|", 23, "|friendly|");

// Orion.Print("Count:" + mobileID.length);

var minEnemyHits = 999;

var minEnemyObj;

var minEnemynotoColor;

var minEnemyID;

for(var i = 0; i < mobileID.length ; i ++){

Orion.AddObject("enemyObj",mobileID[i]);

var obj = Orion.FindObject("enemyObj");

if (obj != null) {

var enemyObj = Orion.FindObject(mobileID[i]);

var notoColor;

//Orion.Print(obj.Name());

if (enemyObj)

{

switch (enemyObj.Notoriety())

{

case 1:

notoColor = 2119;

break;

case 3:

notoColor = 906;

break;

case 6:

notoColor = 33;

break;

default:

notoColor = 69;

}

}

var curEnemyHits = enemyObj.Hits("%") ;

if(minEnemyHits > curEnemyHits || minEnemyHits == 999){

minEnemyHits = curEnemyHits;

minEnemyObj = enemyObj;

minEnemynotoColor = notoColor;

minEnemyID = mobileID[i];

}

}

}

if(minEnemyObj != null)

{

Orion.CharPrint(self, minEnemynotoColor, "" + Orion.RequestName(minEnemyID) + " " + minEnemyObj.Hits("%") + "% HP");

Orion.CharPrint(minEnemyID, minEnemynotoColor, " Target: " + Orion.RequestName(minEnemyID) + " ");

Orion.RemoveHighlightCharacter(lasttarget);

//Orion.CharPrint(enemy.Serial(), '48', enemy.Name());

Orion.AddHighlightCharacter(minEnemyID, '69' );

Orion.ClientLastTarget(minEnemyID);

Orion.TargetSystemSerial(minEnemyID);

Orion.Ignore(minEnemyID);

}

}

function EvadeTimer() {

if (!Orion.BuffExists('0x75f8') && !Orion.DisplayTimerExists('evadetimer')){

Orion.Cast('Evasion')

Orion.Wait(750)}

if (Orion.BuffExists('0x75f8') && !Orion.DisplayTimerExists('evadetimer'))

{Orion.AddDisplayTimer('evadetimer', 25000, 'UnderChar', 'Circle|Bar', 'Evade', 0, 0, '33', 0xff, '0xFFFFFF')}

}

function TeleRoof()

{

Orion.TargetTile('any', Player.X() ,Player.Y(), Player.Z() + 20);

}

function wallscript()

{

var myTarget = Orion.FindObject(Orion.ClientLastTarget());

if (myTarget)

{

var range = Orion.GetDistance(myTarget.Serial());

if (range <= 12)

{

var value = 3;

var empty = 0;

var offset =

[

[empty, -value], //direction = 0

[value, -value], //direction = 1

[value, empty], //direction = 2

[value, value], //direction = 3

[empty, value], //direction = 4

[-value, value], //direction = 5

[-value, empty], //direction = 6

[-value, -value] //direction = 7

];

var xy = offset[myTarget.Direction() & 7];

//var scro = Orion.FindType('0x1F42', ['0x0000'], backpack);

//if (scro.length != 0)

// Orion.UseType('0x1F42', ['0x0000'], backpack);

//else

// Orion.CharPrint(self, 33, '[No Scroll!]');

Orion.TargetTile('0xffff', myTarget.X()+ xy[0] , myTarget.Y()+ xy[1] , myTarget .Z)

}

else

{

Orion.CharPrint('self', 99 , 'too far');

}

}

else

{

Orion.CharPrint('self' , 99 , 'No Target');

}

}

function Mounting ()

{

var horsey = Orion.FindType('0x00e2|0x0076|0x0072|0x00cc|0x0079|0x0115', -1, ground, 'near|live|nothuman', 4, 'green')

var obj = Orion.FindObject('myMount')

while (Orion.ObjAtLayer('Mount') == null){

Orion.Wait(100)

Orion.UseObject(horsey)

Orion.GetStatus('myMount')

Orion.Say('All Follow Me')

if (obj.Poisoned()){

Orion.Cast('Cure')

Orion.WaitForTarget();

Orion.TargetObject('myMount')}

else

{Orion.UseObject('myMount')}

break;}

}

function ShieldArmDisarm()

{

var shield = 0x4639CC07

if (Orion.ObjAtLayer('LeftHand') !== null)

{

Orion.Unequip('LeftHand');

}

else

{

Orion.Equip(shield);

}

}

function TargetCheck()

{

var LT = lasttarget

if (Orion.GetDistance(lasttarget) <=10 && Orion.InLOS(lasttarget))

{

Orion.TargetObject(lasttarget)}

}

function useShuriTargetLast()

{

var itemID = '0x2790';

var list = Orion.FindType(itemID);

for (var i = 0; i < list.length; i++)

{

var item = list[i];

var itemProps = Orion.FindObject(item).Properties();

if (!Orion.Contains(itemProps, 'Uses Remaining: 0'))

{

Orion.UseObject(item);

Orion.WaitForTarget(1000);

Orion.TargetObject('lasttarget');

break;}

}

}

function NearestFocusAttack ()

{

var focus = FindNearestEnemy();

if (!Orion.SpellStatus('Focus Attack'))

{

Orion.Cast('Focus Attack');

Orion.Wait(100);

Orion.Attack('focus')}

}

function FocusAttack()

{

if (!Orion.SpellStatus('Focus Attack'))

{

Orion.Cast('Focus Attack');

Orion.Wait(100);

Orion.Attack(lasttarget)}

}

function autoDeathstrike()

{

while (true){

Orion.Wait(100)

if (!Orion.SpellStatus('Death Strike'))

{

Orion.Cast('Death Strike');

Orion.Wait(100);

Orion.Attack(lasttarget)}

}}

function stackshuriken (){

var BeltProps = Orion.FindObject('belt').Properties();

Orion.FindObject('belt')

Orion.RequestContextMenu('belt');

Orion.Wait(100)

Orion.WaitContextMenuID('belt', 701);

Orion.FindType('shuriken')

Orion.WaitForTarget(1000)

Orion.TargetType('shuriken')

if (Orion.Contains(BeltProps, 'Uses Remaining: 10')){

Orion.AddObject('belt')}

}

function Attacklast ()

{

Orion.RemoveHighlightCharacter(lasttarget);

Orion.CharPrint(self, 33, 'Attack ' + Orion.RequestName(lasttarget) );

Orion.ClearHighlightCharacters(any, true);

Orion.AddHighlightCharacter(lasttarget, '1152', true)

Orion.UseAbility('primary', 'state')

Orion.Attack(lasttarget)

}

function DisarmTimerShield ()

{

while (true){

Orion.Wait(250)

var Timer = Orion.Now();

if (Orion.BuffExists('0x754a')) {

Orion.AddDisplayTimer("Disarm", 6000, "AboveChar", "Circle|Bar", "DISARMED",

0,0, '33', 0xFF, '0xFFFFFF')

Orion.Wait(6100)}

while ((Orion.ObjAtLayer('RightHand') == null)){

Orion.Wait(300)

Orion.Arm('rearm')

Orion.Wait(500)}

}

}

function DisarmTimer2 ()

{

while (true){

Orion.Wait(250)

var Timer = Orion.Now();

if (Orion.BuffExists('0x754a')) {

Orion.AddDisplayTimer("Disarm", 6000, "AboveChar", "Circle|Bar", "DISARMED",

0,0, '33', 0xFF, '0xFFFFFF')

Orion.Wait(6100)}

while ((Orion.ObjAtLayer('RightHand') == null) && (Orion.ObjAtLayer('LeftHand') == null)){

Orion.Wait(300)

Orion.Arm('main')

Orion.Wait(500)}

}

}

function nearest ()

{

var near = Orion.FindType('0xFFFFFF', '-1', ground, 'near|mobile', '18', 'orange');

Orion.FindObject(near)

Orion.ClientLastTarget(near)

Orion.RemoveHighlightCharacter(lasttarget);

Orion.Attack(near)

Orion.CharPrint(self, 33, 'Near ' + Orion.RequestName(lasttarget) );

Orion.ClearHighlightCharacters(lasttarget, true);

Orion.AddHighlightCharacter(lasttarget, '1152', true)

}

function ShieldArmDisarmHeph()

{

var shield = Orion.Dress('heph')

if (Orion.ObjAtLayer('LeftHand') !== null)

{

Orion.Unequip('LeftHand');

}

else

{

Orion.Equip(shield);

}

}

function Attacklast2 ()

{

Orion.UseAbility('Secondary', 'state');

Orion.RemoveHighlightCharacter(lasttarget);

Orion.CharPrint(self, 33, 'Attack ' + Orion.RequestName(lasttarget) );

Orion.ClearHighlightCharacters(any, true);

Orion.AddHighlightCharacter(lasttarget, '1152', true)

Orion.Attack(lasttarget)

}

function MomentumAttack()

{

while (true){

Orion.Wait(500)

if (!Orion.SpellStatus('Momentum Strike'))

{

Orion.Cast('Momentum Strike');

Orion.Wait(100);

Orion.Attack(lasttarget)}

}}

function bolas ()

{

Orion.UseObject('self');

Orion.Wait(1100);

Orion.UseType('0x26AC', '0xFFFF');

Orion.Wait(1100)

while (!Orion.ObjAtLayer('LeftHand') && !Orion.ObjAtLayer('RightHand')){

Orion.Wait(150)

Orion.Dress('heph');

}}

function healpots ()

{

var heal = Orion.FindType('0x0f0c', 'any', backpack)

Orion.UseItemOnMobile(heal[0], heal[0])

}

function curepot ()

{

var cure = Orion.FindType('0x0f07', 'any', backpack)

Orion.UseItemOnMobile(cure[0], cure[0])

}

function refresh ()

{

var refresh = Orion.FindType('0x0f0B', 'any', backpack)

Orion.UseItemOnMobile(refresh[0], refresh[0])

}

function DispelFieldInFront()

{

Orion.Cast("Dispel Field");

var field, fieldObject, index, fieldIndex;

var tile =

[

[Player.X(), Player.Y() - 1],

[Player.X() + 1, Player.Y() - 1],

[Player.X() + 1, Player.Y()],

[Player.X() + 1, Player.Y() + 1],

[Player.X(), Player.Y() + 1],

[Player.X() - 1, Player.Y() + 1],

[Player.X() - 1, Player.Y()],

[Player.X() - 1, Player.Y() - 1]

];

field = Orion.FindType("0x0082|0x3946|0x3956|0x3979|0x3967", "-1", "ground", "item", 1);

if (field.length != 0)

{

for (index = 0; index < tile.length; index++)

{

if (Player.Direction() == index || Player.Direction() == (index + 128))

{

for (fieldIndex = 0; fieldIndex < field.length; fieldIndex++)

{

fieldObject = Orion.FindObject(field[fieldIndex]);

if (fieldObject.X() == tile[index][0] && fieldObject.Y() == tile[index][1])

{

Orion.WaitTargetObject(fieldObject.Serial());

return;

}}

break;

}}}}

function darttester()

{

var dist = Orion.GetDistance(lasttarget);

Orion.WaitTargetObject(lasttarget);

Orion.UseObject(dist < 6 ? Dart() : useShuriTargetLast());

Orion.Wait(500)

Orion.CancelWaitTarget(lasttarget)

}

function autoDisarm() {

while (!Player.Dead()){

var enemy = Orion.FindType("any ", -1, "ground", "human|near|live", "3", "orange");

Orion.Wait(500);

if (Orion.WaitWhileTargeting() && !Player.Frozen()

&& !Orion.BuffExists(0x9bd1) && !Player.Hidden()) {

Orion.Wait(500)

Orion.UseAbility('Primary', 'state');

if (Orion.GetDistance(enemy) < 2){

Orion.Wait(500)

Orion.Attack(enemy)}

}}

}

function AutoDisarmer3() {

while (!Player.Dead()){

Orion.Wait(500);

if (Orion.WaitWhileTargeting() && !Player.Frozen()

&& !Orion.BuffExists(0x9bd1) && !Player.Hidden()) {

Orion.Wait(500)

Orion.UseAbility('Primary', 'state');

}}

}

function Dart()

{

var itemID = '0x27AA';

var list = Orion.FindType(itemID);

for (var i = 0; i < list.length; i++)

{

var item = list[i];

var itemProps = Orion.FindObject(item).Properties();

if (!Orion.Contains(itemProps, 'Uses Remaining: 0'))

{

Orion.UseObject(item);

Orion.WaitForTarget(1000);

Orion.TargetObject('lasttarget');

break;}

}

}

function poisonshuris()

{

var itemID = '0x27AC';

var list = Orion.FindType(itemID);

for (var i = 0; i < list.length; i++)

{

var item = list[i];

var itemProps = Orion.FindObject(item).Properties();

if (!Orion.Contains(itemProps, 'Deadly Poison Charges: '));

{

Orion.UseSkill('30');

Orion.Wait(1000);

Orion.TargetType('0x0F0A', '0x0000');

Orion.Wait(1000);

Orion.TargetObject(item);

Orion.Wait(9500);

}}}

function darting()

{

Orion.PlayMacro('DartMaker')

Orion.Wait(150)

Orion.PlayMacro('UnloadFukiya')

Orion.Wait(150)

}

function poisondarts()

{

var ItemID = '0x2806'

var list = Orion.FindType(ItemID);

for (var i = 0; i < list.length; i++)

{

var item = list[i];

var itemProps = Orion.FindObject(item).Properties();

if (!Orion.Contains(itemProps, 'Deadly Poison Charges: 10'));

{

Orion.UseSkill('30');

Orion.Wait(1000);

Orion.TargetType(0x0F0A, backpack);

Orion.Wait(1000);

Orion.TargetObject(item);

Orion.Wait(9500);

}}}

function BoatRopes ()

{

var rope = Orion.FindType('0x14FA', -1, ground, "near", "11", any);

for (var c = 0; c < rope.length; c++)

{

var serial = rope[c]

Orion.UseObject(serial);}

}

function Lootsies () {

const delay = 1050; // server-imposed delay between actions

const itemList = 'CorpsesLOOT'; // OA list of items to loot

const maxRadius = 2; // maximum looting radius

const ping = 100; // round-trip latency between client and server

var itemCount = 0;

scavenge();

loot();

if (itemCount > 1)

Orion.Print('[Info] autoLoot: Acquired ' + itemCount + ' items.');

else if (itemCount == 1)

Orion.Print('[Info] autoLoot: Acquired ' + itemCount + ' item.');

function scavenge() {

var queue = Orion.FindList(itemList, ground, '', maxRadius)

if (queue.length <= 0) return;

moveToBackpack(queue);

}

function loot() {

var corpseIgnore = 'CorpsesIGNORE'; // OA list of corpses to ignore

var corpseList = 'CorpsesFIND'; // OA list of corpses to search

Orion.UseIgnoreList(corpseIgnore);

var corpses = Orion.FindList(corpseList, ground, '', maxRadius);

if (corpses.length <= 0) return;

for (var i = 0; i < corpses.length; i++) {

const corpse = corpses[i];

if (!criminalCheck(corpse)) {

var queue = Orion.FindList(itemList, corpse);

moveToBackpack(queue);

Orion.AddIgnoreListObject(corpseIgnore, corpse);

} else {

Orion.AddIgnoreListObject(corpseIgnore, corpse);

}

if (i < corpse.length - 1)

Orion.Wait(delay + ping);

}

}

function moveToBackpack(queue) {

for (var i = 0; i < queue.length; i++) {

const item = queue[i];

Orion.Wait(100)

Orion.MoveItem(item);

if (i < queue.length - 1)

Orion.Wait(delay + ping);

itemCount++;

}

}

function criminalCheck(corpse) {

Orion.ClearJournal();

Orion.OpenContainer(corpse);

Orion.Wait(delay + ping);

if (Orion.InJournal('Looting this corpse will be a criminal act!|You have disabled opening innocent corpses')) {

return true;

} else {

return false;

}

}

}

function followlt ()

{

Orion.Follow(lasttarget)

}

function FollowNextEnemy()

{

var serial = FindNearestEnemy();

Orion.GetEnemiesStatus();

if(!serial.length) return;

var enemy = Orion.FindObject(serial);

var notoColor;

switch (enemy.Notoriety() ) {

case 1: notoColor = 2119;

break;

case 6: notoColor = 33;

break;

default: notoColor = 48;

}

Orion.CharPrint(self, notoColor, enemy.Name() );

Orion.RemoveHighlightCharacter(lasttarget, true);

//Orion.CharPrint(enemy.Serial(), '48', enemy.Name());

Orion.AddHighlightCharacter(enemy.Serial(), '1152', true);

Orion.ClientLastTarget(serial);

Orion.TargetSystemSerial(serial)

Orion.Follow(serial);

Orion.Ignore(serial);

}

function InfiniteTelering() {

var rings = Orion.FindTypeEx('0x108A', '0xFFFF', backpack);

if (rings.length) {

for (var i = 0; i < rings.length; i++) {

if (!Orion.Contains(rings[i].Properties(), 'Teleport Charges: 0')) {

Orion.UseObject(rings[i].Serial());

return;

}

}

}

}

function Durability(){

mindur = 100; //Durability to alert you at.

layers = [

0,

"RightHand",

"LeftHand",

'Shoes',

"Pants",

'Shirt',

'Helmet',

'Gloves',

'Ring',

'Talisman',

'Necklace',

'Hair',

'Waist',

'InnerTorso',

'Bracelet',

'Face',

'Beard',

'MidTorso',

'Earrings',

'Arms',

'Cloak',

'Backpack',

'Robe',

'Eggs',

'Legs',

]

Orion.Print("Minimum Durability set at: " + mindur)

for(var i =1; i< layers.length; i++){

if(Orion.ObjAtLayer(i)){

var theobj = Orion.ObjAtLayer(i);

var properties = theobj.Properties();

if (!Orion.Contains(theobj.Properties(), 'Durability')){

//Orion.Print('-')

Orion.Wait(10);

}

else{

var matches = /Durability (\d+)\s\/\s(\d+)/.exec(properties);

if(matches.length > 2 && Number(matches[1]) < mindur)

{

Orion.Print(layers[theobj.Layer()] + ": " +matches[1] + " / " +matches[2])

}

}

}

}

}

function Use\_Nova(){

Orion.UseType('0x0F09', '0x000D')

if(Orion.Timer("NovaTimer") >= -1){

Orion.AddDisplayTimer("10", 120000, "Bottom", "Circle|Bar", "NOVA", 0, 0, '55', 0xFF, '0xFFFFFF');

Orion.SetTimer("NovaTimer",-120000);

}

}

function CreateTriggerBombard(){

Orion.Cast('686');

if (Orion.WaitForGump(10000))

{

var gump0 = Orion.GetGump('last');

if ((gump0 !== null) && (!gump0.Replayed()) && (gump0.ID() === '0x00002336'))

{

gump0.Select(Orion.CreateGumpHook(110));

Orion.Wait(100);

}

}

}

function UseGate()

{

Orion.CancelWaitGump();

Orion.WaitGump(Orion.CreateGumpHook(1));

if (!Orion.UseFromGround("0x0F6C|0x4BCB", "-1", 2))

{

Orion.CharPrint(self, 1153, "Gate is too far away!");

Orion.CancelWaitGump();

}

}

function bolatele()

{

targbehind();

if (Orion.WaitWhileTargeting())

Orion.Wait(100)

Orion.Cast('Teleport')

Orion.WaitForTarget(2000)

targbehind();

Orion.CancelWaitTarget();

}

function TargetPet(){

for (var ind = 0; ind < 2; ind++)

{

var petID = Orion.FindType("-1", "-1", "ground", "nothuman|near|mobile|ignorefriends|ignoreself|live", 24,'orange');

if (petID.length != 0)

{

var enemyObj = Orion.FindObject(petID[0]);

var notoColor;

if (enemyObj)

{

switch (enemyObj.Notoriety())

{

case 1:

notoColor = 2119;

break;

case 3:

notoColor = 906;

break;

case 6:

notoColor = 33;

break;

default:

notoColor = 906;

}

Orion.CharPrint(self, notoColor=1725, "[Pet] " + Orion.RequestName(petID[0]) + " " + enemyObj.Hits("%") + "% ");

Orion.CharPrint(petID[0], notoColor=1725, "\*Target\*");

}

Orion.SetGlobal("global\_enemy", petID[0]);

Orion.ClientLastTarget(petID[0]);

Orion.TargetSystemSerial(petID[0]);

Orion.Ignore(petID[0])

Orion.IgnoreReset()

Orion.CancelWaitTarget();

return;

}

else if (ind == 0)

{

Orion.CancelWaitTarget();

Orion.IgnoreReset();

}

}

Orion.Print("No Pet");

}

function elven()

{

Orion.Dress('elven')

Orion.Wait(100)

Orion.UseAbility('Secondary', 'State')

TargetPet();

Attacklast();

}

function primaryattack()

{

Orion.UseAbility('Primary', 'State')

Orion.Wait(100)

Attacklast();

}

function secondaryattack()

{

Orion.UseAbility('Secondary', 'State')

Orion.Wait(100)

Attacklast();

}

function TeleMaxNew()

{

var value = 11;

var empty = 0;

var offset =

[

[empty, -value], //direction = 0

[value, -value], //direction = 1

[value, empty], //direction = 2

[value, value], //direction = 3

[empty, value], //direction = 4

[-value, value], //direction = 5

[-value, empty], //direction = 6

[-value, -value] //direction = 7

];

//var xy = offset[Player.Direction() & 10];

var dire;

Orion.CancelWaitTarget()

if (Player.Direction()>7){

dire = Player.Direction() - 128;

}

else{

dire = Player.Direction();

}

if(!Orion.HaveTarget()){

Orion.WaitTargetTileRelative('land', offset[dire][0],offset[dire][1], Player.Z());

Orion.Cast("Teleport");

}

else{

Orion.TargetTileRelative('land', offset[dire][0],offset[dire][1], Player.Z());

}

}

function Timers()

{

var G = 1;

while(G == 1)

{

Orion.Wait(100);

if (Orion.BuffExists('0x7540') && !Orion.DisplayTimerExists('Attune'))

{

Orion.AddDisplayTimer('Attune', 255000, 'Bottom', 'Circle|Bar', 'Attune', -150, 0, '906', 0xff, '0xFFFFFF')

}

if (Orion.BuffExists('0x7554') && !Orion.DisplayTimerExists('Gift'))

{

Orion.AddDisplayTimer('Gift', 150000, 'Bottom', 'Circle|Bar', 'Gift', -220, 0, '906', 0xff, '0xFFFFFF')

}

var ManaShield = Orion.InJournal('Mana Shield has been purged.')

if(ManaShield)

{

Orion.CharPrint('self', 906 , 'Mana Shield');

Orion.ClearJournal('Mana Shield has been purged.')

}

var Gift = Orion.InJournal('Gift of Renewal has been purged.')

if(Gift)

{

Orion.CharPrint('self', 906 , 'Gift of Renewal');

Orion.ClearJournal('Gift of Renewal has been purged.')

}

var Attune = Orion.InJournal('Attune Weapon has been purged.')

if(Attune)

{

Orion.CharPrint('self', 906 , 'Attune Weapon');

Orion.ClearJournal('Attune Weapon has been purged.')

}

var Protection = Orion.InJournal('Protection has been purged.')

if(Protection)

{

Orion.CharPrint('self', 906 , 'Protection');

Orion.ClearJournal('Protection has been purged.')

}

var Disarm = Orion.InJournal('You disarm their weapon!')

if(Disarm)

{

Orion.CharPrint('self', 67 , '\*Disarm\*');

Orion.ClearJournal('You disarm their weapon!')

Orion.AddDisplayTimer("2", 18000, "UnderChar", "Line|Bar", "-", 3, 50, '906' , 0xFF , '0xEE2894');

Orion.DisplayTimerExists ('1')

Orion.DisplayTimerSetObject("2", 'lasttarget')

}

}}

function talismanward2(){

var ward = Orion.FindList('Ward')

var talis = 'ignoretali'

Orion.UseIgnoreList(talis)

for (var c = 0; c < 1; c++){

const tali = ward[c]

var Cameo= Orion.FindObject(Cameo)

if (tali !=0){

Orion.Equip(tali)

Orion.Wait(1000)

Orion.UseObject(tali)

Orion.Wait(1000)

Orion.Equip('Cameo')

Orion.AddIgnoreListObject(talis, tali)}

if (tali == null){

Orion.ClearIgnoreList('ignoretali')

Orion.CharPrint(self, 55, 'Resetting List')}

}

}

function ScanDeaths()

{

Orion.ClearJournal()

while (true)

{

Orion.Wait(100)

Orion.WaitJournal('soundecho: id=0x0423|soundecho: id=0x0315', 0, 0, 'sys')

Orion.ClearJournal()

}

}

function getgraphic()

{

Orion.GetGraphic()

}

function heightenbarrab ()

{

Orion.UseType(0x0F06, 0x04F8, backpack)

Orion.Wait(500)

Orion.Cast('Heighten Senses')

}

function Counterstrike() {

while (!Player.Dead()) {

Orion.Wait(500);

if ((Orion.WaitWhileTargeting()) && (!Player.Frozen()) && (!Orion.BuffExists(0x75f9)) && (!Player.Hidden())) {

Orion.Wait(1000);

Orion.Cast('counter attack')

Orion.Wait(1500)

}

}

}

function barrab ()

{

Orion.UseType(0x0F06, 0x04F8, backpack)

}

function changetextcolor()

{

Orion.SetFontColor(true, '0x0022')

}

function CatchSerial(){

Orion.SetGlobal("SerialCatch","true")

Orion.ClearHighlightCharacters(true);

while(Orion.GetGlobal("SerialCatch") == "true"){

var caught = Orion.WaitJournal("R;",Orion.Now(),0);

//oldcaught = Orion.GetGlobal("CaughtSerial");

//Orion.RemoveHighlightCharacter(oldcaught,true);

if (caught != null){

caught = caught.Text();

caught = caught.split(";");

caught = caught[1];

caught = Number(caught);

//caught = caught / 14;

Orion.ClientLastTarget(caught);

Orion.SetGlobal("CaughtSerial",caught);

Orion.ClearHighlightCharacters(true);

Orion.AddHighlightCharacter(caught,'1152',true);

//1161 = Blaze fire, 1153 = white,Glacial=1152

if (Orion.FindObject(caught)){

caught = Orion.FindObject(caught);

nameit = caught.Name();

}

else{

nameit = "Not Visible"

}

Orion.AddDisplayTimer(101,30000, 'Top', 'Rectangle|Bar', nameit, 0, 0, '0xFFFF', 4, 'green');

//Orion.Print(Orion.GetGlobal("CaughtSerial"));

}

Orion.Wait(500)

}

}

function manashield()

{

while (!Player.Dead() && !Player.Hidden() && !Orion.HaveTarget())

{

Orion.Wait(1000)

while (!Orion.BuffExists(0x9bb8))

{

Orion.Wait(500)

Orion.Cast('Mana Shield')

}}}

function tamerDismount()

{

Orion.UseObject('self')

Orion.Wait(100)

Orion.Say('All Kill')

}

function healPet ()

{

var pet = Orion.FindObject('myMount')

while (true){

Orion.Wait(500)

Orion.GetStatus('myMount')

if (pet.Poisoned()){

Orion.Cast('Cure')

Orion.Wait(750)

Orion.WaitForTarget();

Orion.Wait(150)

Orion.TargetObject('myMount');}

if (pet.Hits('%') < 90){

Orion.Cast('Greater Heal')

Orion.Wait(1050)

Orion.WaitForTarget();

Orion.TargetObject('myMount')

Orion.Wait(150)}}}

function attackMonster()

{

while (!Player.Dead())

{

Orion.Wait(250)

Orion.UseAbility('Secondary', 'state')

var enemy = Orion.FindType('0xFFFFFF', '-1', ground, 'near|mobile|live', '24', 'gray');

if ( enemy.length ) {

Orion.Wait(100)

Orion.Attack(enemy[0]);}}}

function attackMonster2()

{

while (!Player.Dead())

{

Orion.Wait(250)

Orion.UseAbility('Primary', 'state')

var enemy = Orion.FindType('0xFFFFFF', '-1', ground, 'near|mobile|live', '24', 'gray');

if ( enemy.length ) {

Orion.Wait(100)

Orion.Attack(enemy[0]);}}}

function Taming()

{

while (!Player.Dead())

{

var enemy = Orion.FindType('0xFFFFFF', '-1', ground, 'near|mobile|live', '18', 'gray');

var leopard = Orion.FindType('0x0040|0x0041|0x0022|0x0025|0x0024|0x0023|0x00E9|0x00E8|', '-1', ground, 'near|mobile|live', '16', 'gray');

var ally = Orion.FindType('0xFFFFFF', '-1', ground, 'near|mobile|live', '18', 'green');

Orion.Wait(250);

if (leopard.length) {

Orion.Follow(leopard);

Orion.Wait(1000)

Orion.UseSkill('Animal Taming');

Orion.Wait(1000)

Orion.TargetObject(leopard);

Orion.Wait(1000);

}

else{

Orion.WalkTo(1125, 3581)}}

}

function attackPet()

{

while (!Player.Dead())

{

Orion.Wait(150)

Orion.Ignore('self');

var friends = Orion.GetFriendList();

for(var i = 0; i < friends.length; i++) {

Orion.Ignore(friends[i]);

}

var enemy = Orion.FindType('0xFFFFFF', '-1', ground, 'near|mobile|live', '18', 'green');

if ( enemy.length ) {

Orion.Attack(enemy[0]);}}}

function getCoords(){

SelectedTile.X()

}

function pBearTame()

{

while (!Player.Dead())

{

var leopard = Orion.FindType('0x00D5', '-1', ground, 'near|mobile|live', '24', 'gray');

Orion.Wait(250);

if (leopard.length) {

Orion.Follow(leopard);

Orion.Wait(1000)

Orion.UseSkill('Animal Taming');

Orion.Wait(1000)

Orion.TargetObject(leopard);

Orion.Wait(1000);

}

else{

Orion.WalkTo(4052,563)}}

}

function leopardTame()

{

while (!Player.Dead())

{

var leopard = Orion.FindType('0x0040|0x0041|0x00DD', '-1', ground, 'near|mobile|live', '24', 'gray');

Orion.Wait(250);

if (leopard.length) {

Orion.Follow(leopard);

Orion.Wait(1000)

Orion.UseSkill('Animal Taming');

Orion.Wait(1000)

Orion.TargetObject(leopard);

Orion.Wait(1000);

}

else{

Orion.WalkTo(3930,364)}}

}

function wwolfTame()

{

while (!Player.Dead())

{

var leopard = Orion.FindType('0x0040|0x0041|0x0042|0x0022|0x0025|0x0024|0x0023', '-1', ground, 'near|mobile|live', '24', 'gray');

Orion.Wait(250);

if (leopard.length) {

Orion.Follow(leopard);

Orion.Wait(1000)

Orion.UseSkill('Animal Taming');

Orion.Wait(1000)

Orion.TargetObject(leopard);

Orion.Wait(1000);

}

else{

Orion.WalkTo(3931,364)}}

}

function fullTamer(){

var taming = Orion.SkillValue('Animal Taming')

var pbear = Orion.FindType('0x00D5', '-1', ground, 'near|mobile|live', '18', 'gray');

var leopard = Orion.FindType('0x0041|0x0042', '-1', ground, 'near|mobile|live', '24', 'gray');

var wwolf = Orion.FindType('0x0041|0x0042|0x0022|0x0025|0x0024|0x0023', '-1', ground, 'near|mobile|live', '24', 'gray');

while (true)

{

Orion.Wait(250)

if (taming <= 590) {

pBearTame();}

if ((taming > 590) && (taming < 690)){

leopardTame();}

if (taming >= 690){

wwolfTame();

}}}

function autoInsure() {

while (true){

Orion.Wait(250)

var item = Orion.FindTypeEx(any).filter(function (part)

{

return (Orion.Contains(part.Name(), "Shuriken") && part.Properties().match(/insured/i) == null)

})

if(item.length)

{

Orion.RequestContextMenu(self);

Orion.WaitContextMenuID(self, 418);

Orion.WaitForTarget(1000);

Orion.TargetObject(item[0].Serial());

Orion.Wait(300);

Orion.CancelTarget();

Orion.Wait(300);

}}}

function autoInsureDarts() {

while (true){

Orion.Wait(250)

var item = Orion.FindTypeEx(any).filter(function (part)

{

return (Orion.Contains(part.Name(), "Fukiya Dart") && part.Properties().match(/insured/i) == null)

})

if(item.length)

{

Orion.RequestContextMenu(self);

Orion.WaitContextMenuID(self, 418);

Orion.WaitForTarget(1000);

Orion.TargetObject(item[0].Serial());

Orion.Wait(300);

Orion.CancelTarget();

Orion.Wait(300);

}}}

function petBandz()

{

var pet = Orion.FindType("0xFFFFFF", "-1", ground, "next|nothuman|mobile", 3, "green");

var myPet = Orion.FindObject(pet)

var vet = Orion.SkillValue('Veterinary')

while (!Player.Dead() && vet > 30)

{

Orion.Wait(350)

Orion.GetStatus(myPet)

if (pet.length && myPet.Hits('%') < 95)

{

Orion.BandageTarget(pet) // Bandie

Orion.WaitJournal("You apply|You must|You have been cured|You did|You fail|damaged|far away|You finish",Orion.Now(),Orion.Now()+4500)

Orion.Wait(1000)

}

}

}

function petSpellz()

{

var myPet = Orion.FindType("-1", "-1", "ground", "nothuman|live", 23, "grey");

while (!Player.Dead())

{

Orion.Wait(500)

Orion.GetStatus(myPet.Serial())

Orion.Wait(250)

if (myPet.Poisoned() && myPet.Hits('%') < 91)

{

Orion.Cast('Cure')

Orion.Wait(1000)

Orion.TargetObject(myPet)

}

if (myPet.Hits('%') < 90 && !myPet.Poisoned())

{

Orion.Cast('Greater Heal')

Orion.Wait(2000)

Orion.TargetObject(myPet)

}

}

}

function lowPet(){

Orion.IgnoreReset();

Orion.Ignore(self);

var mobileID = Orion.FindType("-1", "-1", "ground", "nothuman|live", 23, "green");

// Orion.Print("Count:" + mobileID.length);

var minEnemyHits = 999;

var minEnemyObj;

var minEnemynotoColor;

var minEnemyID;

for(var i = 0; i < mobileID.length ; i ++){

Orion.AddObject("enemyObj",mobileID[i]);

var obj = Orion.FindObject("enemyObj");

if (obj != null) {

var enemyObj = Orion.FindObject(mobileID[i]);

var notoColor;

//Orion.Print(obj.Name());

if (enemyObj)

{

switch (enemyObj.Notoriety())

{

case 1:

notoColor = 2119;

break;

case 3:

notoColor = 906;

break;

case 6:

notoColor = 33;

break;

default:

notoColor = 69;

}

}

var curEnemyHits = enemyObj.Hits("%") ;

if(minEnemyHits > curEnemyHits || minEnemyHits == 900){

minEnemyHits = curEnemyHits;

minEnemyObj = enemyObj;

minEnemynotoColor = notoColor;

minEnemyID = mobileID[i];

}

}

}

if(minEnemyObj != null)

{

Orion.RemoveHighlightCharacter(lasttarget);

//Orion.CharPrint(enemy.Serial(), '48', enemy.Name());

Orion.AddHighlightCharacter(minEnemyID, '69' );

Orion.ClientLastTarget(minEnemyID);

Orion.TargetSystemSerial(minEnemyID);

Orion.Ignore(minEnemyID);

}

}

function gumps(){

Orion.InfoGump('index');

}

function petRezGump()

{

while (!Player.Dead())

{

Orion.Wait(500)

var gump0 = Orion.GetGump('last');

if ((gump0 !== null) && (!gump0.Replayed()) && (gump0.ID() === '0x000001DD'))

{

gump0.Select(Orion.CreateGumpHook(1));

Orion.Wait(100);

}

}

}

function unloadStars()

{

while (true){

Orion.Wait(100)

var itemID = '0x2790';

var list = Orion.FindType(itemID);

for (var i = 0; i < list.length; i++)

{

var item = list[i];

var itemProps = Orion.FindObject(item).Properties();

if (Orion.Contains(itemProps, 'Uses Remaining: 10'))

{

Orion.RequestContextMenu('0x45D774CC'); //Use same ID as above

Orion.WaitContextMenuID('0x45D774CC', 702);

Orion.Wait(1000)

Orion.CancelTarget();

break;}

}

}}

function fillStars2()

{

while (true){

Orion.Wait(100)

var itemID = '0x27AC';

var list = Orion.FindType(itemID);

for (var i = 0; i < list.length; i++)

{

var item = list[i];

var itemProps = Orion.FindObject(item).Properties();

if (Orion.Contains(itemProps, 'Uses Remaining: 2'))

{

Orion.RequestContextMenu('0x45D774CC'); // You have to use YOUR belt ID (Go intro macros, record,

Orion.WaitContextMenuID('0x45D774CC', 701); // click onto your belt, and click load, stop record.

Orion.Wait(1000) // Click onto 'Go to Script' and copy and paste those two lines

Orion.TargetObject(item)

Orion.Wait(1000)

break;}

}

}}

function moveShuri(){

while (true){

Orion.Wait(100)

var itemID = '0x27AC';

var list = Orion.FindType(itemID);

for (var i = 0; i < list.length; i++)

{

var item = list[i];

var itemProps = Orion.FindObject(item).Properties();

if (Orion.Contains(itemProps, 'Shuriken'))

{

Orion.DragItem(item, 0);

Orion.Wait('300');

Orion.DropDraggedItem('0x45BFEC2B');

Orion.Wait('500');

break;}

}

}}

function makingStars(){

fillStars2();

Orion.Wait(1000)

unloadStars();

Orion.Wait(1000)

moveShuri();

}

function barrabCry(){

Orion.Cast('WarCry')

Orion.Wait(500)

barrab();

}

function tester()

{

Orion.Cast('heal')

Orion.Wait(1000)

Orion.TargetObject(self)

}

function ShitTalker()

{

var phrases = [];

phrases = [

"Nice sync script %n.",

"Good lord %n, how many tele rings did you burn to still die?",

"%n, you're 10 pounds of suck in a 5 pound bag.",

'Damn %n, you died faster than it took me to type your name.',

'Should have warcried %n.',

"Sorry %n, you can't overcome the skill that is disarm AI spam",

"I didn't even bother looting your stat pot %n.",

"Losing with double numbers %n? Embarassing.",

"Man %n, that's an ambitious amount of pots for how fast you died.",

"I swear %n's horse put up a harder fight than he did."

]

while (true){

var deathFound = Orion.WaitJournal('soundecho: id=0x0423|soundecho: id=0x0315', Orion.Now(), 0, 'sys')

if(deathFound != null){

Orion.Wait(600);

var deathMsg = Orion.InJournal('a corpse of','item',0,'any',Orion.Now()-1500,Orion.Now()+1000);

if(deathMsg != null){

deathMsg = deathMsg.Text().split("a corpse of ");

deathMsg = deathMsg[1];

var name = deathMsg;

var randIndex = Math.floor(Math.random() \* phrases.length);

deathMsg = phrases[randIndex].replace(/%n/g, name);

var friendList = Orion.GetFriendList(true);

var newFriend = [];

for(var i =0; i < friendList.length;i++){

var temp = friendList[i].split(" ");

var pushName = '';

for(j =1; j< temp.length; j++){

pushName += temp[j];

if(temp.length > 2 && j != temp.length -1){

pushName += " ";

}

}

newFriend.push(pushName);

}

newFriend.push(Player.Name());

if(newFriend.indexOf(name) == -1){

Orion.Say(', ' + deathMsg);

//Orion.Print(deathMsg);

}

}

}

Orion.Wait(500);

}

}

function autoSamp()

{

while (!Player.Dead())

{

var monster = Orion.FindType('any', '-1', ground, 'near|mobile|live', '12', 'gray');

Orion.Wait(1000);

if (monster.length) {

Orion.Follow(monster);

Orion.Wait(250)

Orion.UseAbility('Primary', 'state');

Orion.Wait(200)

Orion.TargetObject(monster);

Orion.Wait(200);

Orion.Attack(monster);

}

else{

Orion.WalkTo(142,690);

Orion.StopWalking();}

}}

function caster()

{

while (true){

Orion.Wait(200)

if (!Player.Dead()){

//if (!Orion.BuffExists('0x75a7')){

//Orion.Cast('Consecrate Weapon')

//Orion.Wait(500)}

if (Player.Hits('%') < 90){

EvadeTimer();}

if (!Orion.BuffExists('0x75f8') && Player.Hits('%') < 90 && !Player.Frozen() && Orion.WaitWhileTargeting()){

Orion.Wait(400)

Orion.Cast('Confidence')

Orion.Wait(2300)}

}}}

function explosion(){

var explode = Orion.FindTypeEx('0x0f0d', '0x0000', backpack);

if (explode.length) {

for (var i = 0; i < explode.length; i++) {

if (explode[i].Count() > 1) {

Orion.UseObject(explode[i].Serial());

Orion.WaitForTarget(500);

Orion.CancelTarget();

return;

}}}}

function sash()

{

var buffed = Orion.BuffExists('0x7598')

if (!buffed){

Orion.PlayMacro('SashOn');}

else{

Orion.PlayMacro('SashOff');

}}

function autoInsure2() {

while (true){

Orion.Wait(250)

var item = Orion.FindTypeEx(any).filter(function (part)

{

return (Orion.Contains(part.Properties(), "Splintering") && part.Properties().match(/insured/i) == null)

})

if(item.length)

{

Orion.RequestContextMenu(self);

Orion.WaitContextMenuID(self, 418);

Orion.WaitForTarget(1000);

Orion.TargetObject(item[0].Serial());

Orion.Wait(300);

Orion.CancelTarget();

Orion.Wait(300);

}}}

function autoAids(){

var aid = Orion.FindType('0x0E21', any, backpack)

while (true){

Orion.Wait(500)

if (!Orion.BuffExists('0x7596') && (Player.Hits('%') < 90)){

Orion.UseObject(aid)

Orion.Wait(200)

Orion.TargetObject(self)}

}}

function SendSerial() {

var LT = Orion.ClientLastTarget();

//LT = LT \* 14;

LT = Orion.FindObject(LT);

Orion.SayGuild("R;" + LT.Serial() + ";" + LT.Name())

}

function CatchSerial(){

Orion.SetGlobal("SerialCatch","true")

while(Orion.GetGlobal("SerialCatch") == "true"){

var caught = Orion.WaitJournal("R;",Orion.Now(),0);

if (caught != null){

caught = caught.Text();

caught = caught.split(";");

caught = caught[1];

caught = Number(caught);

//caught = caught / 14;

Orion.ClientLastTarget(caught);

Orion.SetGlobal("CaughtSerial",caught);

Orion.ClearHighlightCharacters();

Orion.AddHighlightCharacter(caught,'1152');

Orion.AddDisplayTimer(101,30000, 'Top', 'Rectangle|Bar', caught, 0, 0, '0xFFFF', 4, 'green');

//Orion.Print(Orion.GetGlobal("CaughtSerial"));

}

Orion.Wait(500)

}

}

function reloader(){

reloadBelt();

Orion.Wait(50)

reloadDart();}

function reloadDart()

{

var itemID = '0x27AA';

var list = Orion.FindType(itemID);

for (var i = 0; i < list.length; i++)

{

var item = list[i];

var itemProps = Orion.FindObject(item).Properties();

if (Orion.Contains(itemProps, 'Uses Remaining: 0'))

{

Orion.RequestContextMenu(item);

Orion.WaitContextMenuID(item, 703);

if (Orion.WaitForTarget(1000))

Orion.TargetType('0x2806', '0xFFFF');}

}

}

function reloadBelt()

{

var itemID = '0x2790';

var list = Orion.FindType(itemID);

for (var i = 0; i < list.length; i++)

{

var item = list[i];

var itemProps = Orion.FindObject(item).Properties();

if (Orion.Contains(itemProps, 'Uses Remaining: 0'))

{

Orion.RequestContextMenu(item);

Orion.WaitContextMenuID(item, 701);

if (Orion.WaitForTarget(1000))

Orion.TargetType('0x27AC', '0xFFFF');

}

}}

function DartShuri(){

var belts = Orion.FindTypeEx(0x2790, any, backpack)

var blowguns = Orion.FindTypeEx(0x27AA, any, backpack)

reloader();

if (belts.length < 2)

{

Orion.Print('Stopping. Do not have at least two ninja belts.')

return

}

if (blowguns.length < 2)

{

Orion.Print('Stopping. You do not have at least two fukiyas.')

}

if (Orion.GetDistance(lasttarget) >= 5)

{

Orion.UseItemOnMobile(belts[0].Serial(), lasttarget)

Orion.Wait(1000)

}

else

{

Orion.UseItemOnMobile(blowguns[0].Serial(), lasttarget)

Orion.Wait(1000)

}

if (Orion.GetDistance(lasttarget) >= 5)

{

Orion.UseItemOnMobile(belts[1].Serial(), lasttarget)

}

else

{

Orion.UseItemOnMobile(blowguns[1].Serial(), lasttarget)

}

Orion.CancelWaitTarget()

}

function harmShuri(){

Orion.Cast('Harm')

DartShuri();

}

function moving(){

while(true){

Orion.WalkTo(2690,697)

Orion.Wait(2000)

Orion.WalkTo(2679,697)}

}

function EvadeTimer2() {

if (!Orion.BuffExists('0x75f8') && !Orion.DisplayTimerExists('evadetimer')){

Orion.Cast('Evasion')

Orion.Wait(750)}

if (Orion.BuffExists('0x75f8') && !Orion.DisplayTimerExists('evadetimer'))

{Orion.AddDisplayTimer('evadetimer', 25000, 'UnderChar', 'Circle|Bar', 'Evade', 0, 0, '33', 0xff, '0xFFFFFF')}

}

function switchweap(){

if (Orion.ObjAtLayer(1)){

Orion.Dress('hatchet')}

else{

Orion.Dress('main')}}

function donpot() {

var dickbag = false

var pot = Orion.FindType('0x0F0D', '0x0000', 'backpack');

var potc = Orion.Count('0x0F0D', '0x0000', 'backpack');

if (potc > 0) {

for (var i = 0; i < 2; ++i) {

Orion.UseItemOnMobile(pot[0], pot[0]);

if(dickbag === false){

Orion.AddDisplayTimer('dicbag', 5000, 'AboveChar', 'Circle|Bar', 'BOOM', 0,0);

dickbag = true;

}

Orion.WaitForTarget(1000)

Orion.CancelTarget()

Orion.Wait(950)

}

}

}