using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Drawing;

namespace RefrigtzDLL

{

[Serializable]

public class ThingsConverter

{

//Initiate Global Variables.

public static bool LoadConvertTable = false;

public static int[,] TableConverted = null;

bool ArrangmentsChanged = false;

public static bool ClickOcurred = false;

public static bool ActOfClickEqualTow = false;

public bool Convert = false;

public bool ConvertedToMinister = false;

public bool ConvertedToCastle = false;

public bool ConvertedToElefant = false;

public bool ConvertedToHourse = false;

public int Max;

public int Row, Column;

Color color;

int Order;

int Current = 0;

//AllDraw. THIS;

public ThingsConverter()

{ }

//Constructor

public ThingsConverter(bool Arrangments, int i, int j, Color a, int[,] Tab, int Ord, bool TB, int Cur//,ref AllDraw. THI

)

{

//Initite Global Variables with Local Parameter.

//THIS = THI;

ArrangmentsChanged = Arrangments;

Row = i;

Column = j;

color = a;

Order = Ord;

Current = Cur;

}

//Convert Operation of Randomly All State Method.

public bool ConvertOperation(int i, int j, Color a, int[,] Tab, int Ord, bool TB, int Cur)

{

//Initiate Global variables with Local One.

Row = i;

Column = j;

color = a;

Order = Ord;

Current = Cur;

//If Convert is Act and click tow time occured

if (!Convert && ActOfClickEqualTow)

{

Object O = new Object();

lock (O)

{

AllDraw.ConvertWait = true;

}

Object O1 = new Object();

lock (O1)

{

ClickOcurred = true;

}

//Set tow time click unclicked.

Object O2 = new Object();

lock (O2)

{

ActOfClickEqualTow = false;

}

if (!ArrangmentsChanged)

{

//Convert State Boolean Variable Consideration.

if (Order == 1 && Column == 7)

Convert = true;

if (Order == -1 && Column == 0)

Convert = true;

}

else

{

//Convert State Boolean Variable Consideration.

if (Order == 1 && Column == 0)

Convert = true;

if (Order == -1 && Column == 7)

Convert = true;

}

//If Converted is Occured the Operation od Set and table reference content occured.

if (Convert)

{

if (!AllDraw.Blitz)

{

AllDraw.ConvertedKind = -1;

AllDraw.SodierConversionOcuured = true;

//Randomly Number of 4 kind Object.

int Rand = -1;

if (AllDraw.Person && AllDraw.StateCP && AllDraw.THISSecradioButtonGrayOrderChecked)

{

if (AllDraw.OrderPlate == 1)

{

while (AllDraw.ConvertedKind == -1) { System.Threading.Thread.Sleep(100); }

Rand = AllDraw.ConvertedKind;

AllDraw.ConvertedKind = -2;

}

}

else

if (AllDraw.Person && AllDraw.StateCP && AllDraw.THISSecradioButtonBrownOrderChecked)

{

if (AllDraw.OrderPlate == -1)

{

//(new FormُSelectItems()).ShowDialog();

while (AllDraw.ConvertedKind == -1) { System.Threading.Thread.Sleep(100); }

Rand = AllDraw.ConvertedKind;

AllDraw.ConvertedKind = -2;

}

}

else

Rand = (new Random()).Next(0, 4);

//If Rand is Equaled the Operation will cuased automaticcally Base on Color..

if (Rand == 0)

{

if (Order == 1)

{

// MinisterMidle++;

//MinisterHigh++;

Tab[Row, Column] = 5;

}

else if (Order == -1)

{

//MinisterHigh++;

Tab[Row, Column] = -5;

}

ConvertedToMinister = true;

}

else if (Rand == 1)

{

if (Order == 1)

{

//CastleMidle++;

//CastleHigh++;

Tab[Row, Column] = 4;

}

else if (Order == -1)

{

//CastleHigh++;

Tab[Row, Column] = -4;

}

ConvertedToCastle = true;

}

else if (Rand == 2)

{

if (Order == 1)

{

//HourseMidle++;

//HourseHight++;

Tab[Row, Column] = 3;

}

else if (Order == -1)

{

//HourseHight++;

Tab[Row, Column] = -3;

}

ConvertedToHourse = true;

}

else

{

if (Order == 1)

{

//ElefantMidle++;

//ElefantHigh++;

Tab[Row, Column] = 2;

}

else if (Order == -1)

{

//ElefantHigh++;

Tab[Row, Column] = -2;

}

ConvertedToElefant = true;

}

}

else

{

if (AllDraw.OrderPlate == 1)

{

AllDraw.ConvertedKind = -1;

AllDraw.SodierConversionOcuured = true;

//Randomly Number of 4 kind Object.

int Rand = -1;

if (AllDraw.Person && AllDraw.StateCP && AllDraw.THISSecradioButtonGrayOrderChecked)

{

if (AllDraw.OrderPlate == 1)

{

while (AllDraw.ConvertedKind == -1) { System.Threading.Thread.Sleep(100); }

Rand = AllDraw.ConvertedKind;

AllDraw.ConvertedKind = -2;

}

}

else

if (AllDraw.Person && AllDraw.StateCP && AllDraw.THISSecradioButtonBrownOrderChecked)

{

if (AllDraw.OrderPlate == -1)

{

while (AllDraw.ConvertedKind == -1) { System.Threading.Thread.Sleep(100); }

Rand = AllDraw.ConvertedKind;

AllDraw.ConvertedKind = -2;

}

}

else

Rand = (new Random()).Next(0, 4);

//If Rand is Equaled the Operation will cuased automaticcally Base on Color..

if (Rand == 0)

{

if (Order == 1)

{

// MinisterMidle++;

//MinisterHigh++;

Tab[Row, Column] = 5;

}

else if (Order == -1)

{

//MinisterHigh++;

Tab[Row, Column] = -5;

}

ConvertedToMinister = true;

}

else if (Rand == 1)

{

if (Order == 1)

{

//CastleMidle++;

//CastleHigh++;

Tab[Row, Column] = 4;

}

else if (Order == -1)

{

//CastleHigh++;

Tab[Row, Column] = -4;

}

ConvertedToCastle = true;

}

else if (Rand == 2)

{

if (Order == 1)

{

//HourseMidle++;

//HourseHight++;

Tab[Row, Column] = 3;

}

else if (Order == -1)

{

//HourseHight++;

Tab[Row, Column] = -3;

}

ConvertedToHourse = true;

}

else

{

if (Order == 1)

{

//ElefantMidle++;

//ElefantHigh++;

Tab[Row, Column] = 2;

}

else if (Order == -1)

{

//ElefantHigh++;

Tab[Row, Column] = -2;

}

ConvertedToElefant = true;

}

}

else

{

AllDraw.ConvertedKind = -1;

AllDraw.SodierConversionOcuured = true;

//Randomly Number of 4 kind Object.

int Rand = -1;

if (AllDraw.Person && AllDraw.StateCP &&AllDraw.THISSecradioButtonGrayOrderChecked)

{

if (AllDraw.OrderPlate == 1)

{

while (AllDraw.ConvertedKind == -1) { System.Threading.Thread.Sleep(100); }

Rand = AllDraw.ConvertedKind;

AllDraw.ConvertedKind = -2;

}

}

else

if (AllDraw.Person && AllDraw.StateCP && AllDraw.THISSecradioButtonBrownOrderChecked)

{

if (AllDraw.OrderPlate == -1)

{

while (AllDraw.ConvertedKind == -1) { System.Threading.Thread.Sleep(100); }

Rand = AllDraw.ConvertedKind;

AllDraw.ConvertedKind = -2;

}

}

else

Rand = (new Random()).Next(0, 4);

//If Rand is Equaled the Operation will cuased automaticcally Base on Color..

if (Rand == 0)

{

if (Order == 1)

{

//MinisterMidle++;

//MinisterHigh++;

Tab[Row, Column] = 5;

}

else if (Order == -1)

{

//MinisterHigh++;

Tab[Row, Column] = -5;

}

ConvertedToMinister = true;

}

else if (Rand == 1)

{

if (Order == 1)

{

//CastleMidle++;

//CastleHigh++;

Tab[Row, Column] = 4;

}

else if (Order == -1)

{

//CastleHigh++;

Tab[Row, Column] = -4;

}

ConvertedToCastle = true;

}

else if (Rand == 2)

{

if (Order == 1)

{

//HourseMidle++;

//HourseHight++;

Tab[Row, Column] = 3;

}

else if (Order == -1)

{

//HourseHight++;

Tab[Row, Column] = -3;

}

ConvertedToHourse = true;

}

else

{

if (Order == 1)

{

//ElefantMidle++;

//ElefantHigh++;

Tab[Row, Column] = 2;

}

else if (Order == -1)

{

//ElefantHigh++;

Tab[Row, Column] = -2;

}

ConvertedToElefant = true;

}

}

}

AllDraw.ConvertWait = false;

}

}

AllDraw.ConvertWait = false;

if (Convert)

{

Object O = new Object();

lock (O)

{

TableConverted = new int[8, 8];

for (int iii = 0; iii < 8; iii++)

for (int jjj = 0; jjj < 8; jjj++)

TableConverted[iii, jjj] = Tab[iii, jjj];

}

Object O1 = new Object();

lock (O1)

{

LoadConvertTable = true;

}

}

//System.Threading.Thread.Sleep(100);

//return Convert State.

return Convert;

}

}

}

//End of Documeatation.