

```

Namespace MyApp
#Import "<std>"
#Import "<mojo>"
#Import "assets/"
Using std..
Using mojo..

```

' This example assumes you have a 128x128 image ready with lots of 8x8 tiles in it, or it could be anything you'll still see the effect.

```
Const w:=2732.0/2.0, h:=2048.0/2.0 ' Display resolution
```

```
Class MyApp Extends Window
```

```

    Field canvas:Canvas
    Field image:Image
    Field s:Int, wx:Int, wy:Int
    Field tilemap:Int[] = New Int[512*512] ' Create a world made of 512 x 512 tiles
    Field size:Int=32 ' Tilesize, change to anything you'd like
    Field xsprite:Int = 2000, ysprite:Int = 256, xd:Int = -4 ' An example of an enemy

```

```
    Method
```

```

New( title:String="Myapp",width:Int=w,height:Int=h,flags:WindowFlags=WindowFlags.Resizable)
    Super.New("Myapp",w,h,flags)

```

```

        ' Create a random example world of tiles
        For Local y:=0 To 511
            For Local x:=0 To 511
                tilemap[x + y * 512] = Int(Rnd(127))
            Next
        Next

```

```
        image=Image.Load("asset::128x1282.png", Null,Null)
```

```
    End
```

```
    Method OnRender( canvas:Canvas ) Override
```

```
        App.RequestRender()
```

```

        canvas.Clear(Color.Black)
        Local scrx:=wx Mod size
        Local scry:=wy Mod size
        Local mapx:=wx / size
        Local mapy:=wy / size
        Local cnty:= -scry
        For Local y:=mapy To (mapy+(h/size)+1)
            Local cntx:=-scrx
            For Local x:=mapx To (mapx+(w/size)+1)
                Local char:= tilemap[x + y * 128]
                Local tilex:= char & 7
                Local tiley:= char Shr 3
                canvas.DrawRect(cntx,cnty,size,size,image,tilex Shl 3,tiley Shl 3,8,8)
            Next
            cntx=cntx+size
            cnty=cnty+size
        Next

```

' Hardcoded to get its sprites from a 128x128px tilesheet made up of 8x8px tiles (Shl 3 = \* 8)

```

        ' Do something like checking for exit and create some movement.

```

```

If Keyboard.KeyReleased(Key.Escape) Then App.Terminate()

' Put a slow automatic scrolling to the right of the world
wx=wx+1

' Add manual scrolling using cursor keys
Local s:=4 ' Manual scrollspped
wx = wx - s * Int(Keyboard.KeyDown(Key.Left)) + s *
Int(Keyboard.KeyDown(Key.Right))
wy = wy - s * Int(Keyboard.KeyDown(Key.Up)) + s *
Int(Keyboard.KeyDown(Key.Down))
wx=Max(0,Min(10000,wx)) ; wy=Max(0,Min(10000,wy)) ' Limit the world
woordinates to stay within 0-10000 (change this to what you like)

' Create a simple back and forth enemy-pattern to have something to look at
xsprite= xsprite + xd
If xsprite > 2900 Then xd = - xd
If xsprite < 700 Then xd=-xd

' Put enemies and objects into the world using this sx sy style of coordinates
Local sx:=xsprite-wx
Local sy:=ysprite-wy
canvas.DrawRect(sx,sy,size,size,image,2 Shl 3, 2 Shl 3,8,8)

canvas.Flush()

End Method

End

Function Main()
    New AppInstance
    New Myapp
    App.Run()
End

```