

Forum:[Hack Forums](#)

Thread: [Gmod TTT score board ranks.](#)

scavskinner4233 - November 9, 2014, 7:16 pm

Hey I tried using this old tutorial <https://hackforums.net/showthread.php?tid=3223233>

But I got this error

Code [ERROR] gamemodes/terrortown/gamemode/vgui/sb_team.lua:97: attempt to index local 'row' (a nil value)

1. AddPlayerRow - gamemodes/terrortown/gamemode/vgui/sb_team.lua:97
2. UpdateScoreboard - gamemodes/terrortown/gamemode/vgui/sb_main.lua:278
3. Init - gamemodes/terrortown/gamemode/vgui/sb_main.lua:134
4. Create - lua/includes/extensions/client/panel/scriptedpanels.lua:153
5. ScoreboardCreate - gamemodes/terrortown/gamemode/cl_scoreboard.lua:27
6. unknown - gamemodes/terrortown/gamemode/cl_scoreboard.lua:34

```
Code self.cols[5] = vgui.Create("DLabel", self)
self.cols[5]:SetText("Rank")
```

```
if KARMA.IsEnabled() then
    self.cols[4] = vgui.Create("DLabel", self)
    self.cols[4]:SetText(GetTranslation("sb_karma"))
end
```

```
if ply:IsUserGroup("superadmin") then
self.cols[5]:SetText("Root Admin")
self.cols[5]:SetTextColor(Color(0,222,11,190))
```

```
else if ply:IsUserGroup("admin") then
self.cols[5]:SetText("Admin")
self.cols[5]:SetTextColor(Color(0,255,0,0))
```

```
else if ply:IsUserGroup("senioradmin") then
self.cols[5]:SetText("Senior Admin")
self.cols[5]:SetTextColor(Color(0,137,0,222))
```

```
else if ply:IsUserGroup("realsuper") then
self.cols[5]:SetText("Super Admin")
self.cols[5]:SetTextColor(Color(0,0,255,255))
```

```
else if ply:IsUserGroup("user") then
self.cols[5]:SetText("Guest")
self.cols[5]:SetTextColor(Color(0,150,150,150))
```

```
end
```

^ That is what I added to the sb_row file. Can anyone tell me what I've done wrong?

xESWx DEATH - November 10, 2014, 7:13 am

you need to post more of the file because it says that the local row variable on line 97 in sb_team.lua

is a nil value or non-existent without more code i can't tell you whats wrong because there has nothing to do with the code you posted.

scavskinner4233 - November 10, 2014, 2:14 pm

Code ---- Scoreboard player score row, based on sandbox version

```
include("sb_info.lua")

local GetTranslation = LANG.GetTranslation
local GetPTranslation = LANG.GetParamTranslation

SB_ROW_HEIGHT = 24 --16

local PANEL = {}

function PANEL:Init()
    -- cannot create info card until player state is known
    self.info = nil

    self.open = false

    self.cols = {}
    self:AddColumn( GetTranslation("sb_ping"), function(ply) return ply:Ping() end )
    self:AddColumn( GetTranslation("sb_deaths"), function(ply) return ply:Deaths() end )
    self:AddColumn( GetTranslation("sb_score"), function(ply) return ply:Frgs() end )

    if KARMA.IsEnabled() then
        self:AddColumn( GetTranslation("sb_karma"), function(ply) return
math.Round(ply:GetBaseKarma()) end )
    end

    -- Let hooks add their custom columns
    hook.Call("TTTScoreboardColumns", nil, self)

    for _, c in ipairs(self.cols) do
        c:SetMouseInputEnabled(false)
    end

    self.tag = vgui.Create("DLabel", self)
    self.tag:SetText("")
    self.tag:SetMouseInputEnabled(false)

    self.sresult = vgui.Create("DImage", self)
    self.sresult:SetSize(16,16)
    self.sresult:SetMouseInputEnabled(false)

    self.avatar = vgui.Create( "AvatarImage", self )
    self.avatar:SetSize(SB_ROW_HEIGHT, SB_ROW_HEIGHT)
    self.avatar:SetMouseInputEnabled(false)

    self.nick = vgui.Create("DLabel", self)
    self.nick:SetMouseInputEnabled(false)
```

```

self.voice = vgui.Create("DImageButton", self)
self.voice:SetSize(16,16)

self:SetCursor( "hand" )
end

function PANEL:AddColumn( label, func, width )
    local lbl = vgui.Create( "DLabel", self )
    lbl.GetPlayerText = func
    lbl.IsHeading = false
    lbl.Width = width or 50 -- Retain compatibility with existing code

    table.insert( self.cols, lbl )
    return lbl
end

local namecolor = {
    default = COLOR_WHITE,
    admin = Color(220, 180, 0, 255),
    dev = Color(100, 240, 105, 255)
};

function GM:TTTScoreboardColorForPlayer(ply)
    if not IsValid(ply) then return namecolor.default end

    if ply:SteamID() == "STEAM_0:0:1963640" then
        return namecolor.dev
    elseif ply:IsAdmin() and GetGlobalBool("tts_highlight_admins", true) then
        return namecolor.admin
    end
    return namecolor.default
end

local function ColorForPlayer(ply)
    if IsValid(ply) then
        local c = hook.Call("TTTScoreboardColorForPlayer", GAMEMODE, ply)

        -- verify that we got a proper color
        if c and type(c) == "table" and c.r and c.b and c.g and c.a then
            return c
        else
            ErrorNoHalt("TTTScoreboardColorForPlayer hook returned something that isn't a color!\n")
        end
    end
    return namecolor.default
end

function PANEL:Paint()
    if not IsValid(self.Player) then return end

    -- if ( self.Player:GetFriendStatus() == "friend" ) then
    --     color = Color( 236, 181, 113, 255 )
    -- end

```

```

local ply = self.Player

if ply:IsTraitor() then
    surface.SetDrawColor(255, 0, 0, 30)
    surface.DrawRect(0, 0, self:GetWide(), SB_ROW_HEIGHT)
elseif ply:IsDetective() then
    surface.SetDrawColor(0, 0, 255, 30)
    surface.DrawRect(0, 0, self:GetWide(), SB_ROW_HEIGHT)
end

if ply == LocalPlayer() then
    surface.SetDrawColor( 200, 200, 200, math.Clamp(math.sin(RealTime() * 2) * 50, 0, 100))
    surface.DrawRect(0, 0, self:GetWide(), SB_ROW_HEIGHT )
end

return true
end

function PANEL:SetPlayer(ply)
    self.Player = ply
    self.avatar:SetPlayer(ply)

    if not self.info then
        local g = ScoreGroup(ply)
        if g == GROUP_TERROR and ply != LocalPlayer() then
            self.info = vgui.Create("TTTScorePlayerInfoTags", self)
            self.info:SetPlayer(ply)

            self:InvalidateLayout()
            elseif g == GROUP_FOUND or g == GROUP_NOTFOUND then
                self.info = vgui.Create("TTTScorePlayerInfoSearch", self)
                self.info:SetPlayer(ply)
                self:InvalidateLayout()
            end
        else
            self.info:SetPlayer(ply)

            self:InvalidateLayout()
        end

        self.voice.DoClick = function()
            if IsValid(ply) and ply != LocalPlayer() then
                ply:SetMuted(not ply:IsMuted())
            end
        end

        self:UpdatePlayerData()
    end

    self.cols[5] = vgui.Create("DLabel", self)
    self.cols[5]:SetText("Rank")

```

```

if KARMA.IsEnabled() then
    self.cols[4] = vgui.Create("DLabel", self)
    self.cols[4]:SetText(GetTranslation("sb_karma"))
end

if ply:IsUserGroup("superadmin") then
self.cols[5]:SetText("Root Admin")
self.cols[5]:SetTextColor(Color(0,222,11,190))

else if ply:IsUserGroup("admin") then
self.cols[5]:SetText("Admin")
self.cols[5]:SetTextColor(Color(0,255,0,0))

else if ply:IsUserGroup("senioradmin") then
self.cols[5]:SetText("Senior Admin")
self.cols[5]:SetTextColor(Color(0,137,0,222))

else if ply:IsUserGroup("realsuper") then
self.cols[5]:SetText("Super Admin")
self.cols[5]:SetTextColor(Color(0,0,255,255))

else if ply:IsUserGroup("user") then
self.cols[5]:SetText("Guest")
self.cols[5]:SetTextColor(Color(0,150,150,150))

end

if self.cols[4] then
    self.cols[4]:SetText(math.Round(ply:GetBaseKarma()))
end

function PANEL:GetPlayer() return self.Player end

function PANEL:UpdatePlayerData()
    if not IsValid(self.Player) then return end

    local ply = self.Player
    for i=1,#self.cols do
        -- Set text from function, passing the label along so stuff like text
        -- color can be changed
        self.cols[i]:SetText( self.cols[i].GetPlayerText(ply, self.cols[i]) )
    end

    self.nick:SetText(ply:Nick())
    self.nick:SizeToContents()
    self.nick:SetTextColor(ColorForPlayer(ply))

    local ptag = ply.sb_tag
    if ScoreGroup(ply) != GROUP_TERROR then
        ptag = nil
    end

    self.tag:SetText(ptag and GetTranslation(ptag.txt) or "")

```

```

self.tag:SetTextColor(ptag and ptag.color or COLOR_WHITE)

self.sresult:SetVisible(ply.search_result != nil)

-- more blue if a detective searched them
if ply.search_result and (LocalPlayer():IsDetective() or (not ply.search_result.show)) then
    self.sresult:SetImageColor(Color(200, 200, 255))
end

-- cols are likely to need re-centering
self:LayoutColumns()

if self.info then
    self.info:UpdatePlayerData()
end

if self.Player != LocalPlayer() then
    local muted = self.Player:IsMuted()
    self.voice:SetImage(muted and "icon16/sound_mute.png" or "icon16/sound.png")
else
    self.voice:Hide()
end
end

function PANEL:ApplySchemeSettings()
    for k,v in pairs(self.cols) do
        v:SetFont("treb_small")
        v:SetTextColor(COLOR_WHITE)
    end

    self.nick:SetFont("treb_small")
    self.nick:SetTextColor(ColorForPlayer(self.Player))

    local ptag = self.Player and self.Player.sb_tag
    self.tag:SetTextColor(ptag and ptag.color or COLOR_WHITE)
    self.tag:SetFont("treb_small")

    self.sresult:SetImage("icon16/magnifier.png")
    self.sresult:SetImageColor(Color(170, 170, 170, 150))
end

function PANEL:LayoutColumns()
    local cx = self:GetWide()
    for k,v in ipairs(self.cols) do
        v:SizeToContents()
        cx = cx - v.Width
        v:SetPos(cx - v:GetWide()/2, (SB_ROW_HEIGHT - v:GetTall()) / 2)
    end

    self.tag:SizeToContents()
    cx = cx - 90
    self.tag:SetPos(cx - self.tag:GetWide()/2, (SB_ROW_HEIGHT - self.tag:GetTall()) / 2)

```

```

    self.sresult:SetPos(cx - 8, (SB_ROW_HEIGHT - 16) / 2)
end

function PANEL:PerformLayout()
    self.avatar:SetPos(0,0)
    self.avatar:SetSize(SB_ROW_HEIGHT,SB_ROW_HEIGHT)

    local fw = sboard_panel.ply_frame:GetWide()
    self:SetWide( sboard_panel.ply_frame.scroll.Enabled and fw-16 or fw )

    if not self.open then
        self:SetSize(self:GetWide(), SB_ROW_HEIGHT)

        if self.info then self.info:SetVisible(false) end
    elseif self.info then
        self:SetSize(self:GetWide(), 100 + SB_ROW_HEIGHT)

        self.info:SetVisible(true)
        self.info:SetPos(5, SB_ROW_HEIGHT + 5)
        self.info:SetSize(self:GetWide(), 100)
        self.info:PerformLayout()

        self:SetSize(self:GetWide(), SB_ROW_HEIGHT + self.info:GetTall())
    end

    self.nick:SizeToContents()

    self.nick:SetPos(SB_ROW_HEIGHT + 10, (SB_ROW_HEIGHT - self.nick:GetTall()) / 2)

    self:LayoutColumns()

    self.voice:SetVisible(not self.open)
    self.voice:SetSize(16, 16)
    self.voice:DockMargin(4, 4, 4, 4)
    self.voice:Dock(RIGHT)
end

function PANEL:DoClick(x, y)
    self:SetOpen(not self.open)
end

function PANEL:SetOpen(o)
    if self.open then
        surface.PlaySound("ui/buttonclickrelease.wav")
    else
        surface.PlaySound("ui/buttonclick.wav")
    end

    self.open = o

    self:PerformLayout()
    self:GetParent():PerformLayout()
    sboard_panel:PerformLayout()

```

end

```
function PANEL:DoRightClick()  
    local menu = DermaMenu()  
    menu.Player = self:GetPlayer()  
  
    local close = hook.Call( "TTTScoreboardMenu", nil, menu )  
    if close then menu:Remove() return end  
  
    menu:Open()  
end
```

```
vgui.Register( "TTTScorePlayerRow", PANEL, "Button" )  
^ thats my entire sb_row file.
```

xESWx DEATH - *November 10, 2014, 8:10 pm*

The issue doesn't look to be that file it seems to be in sb_team.lua

Azzurez - *January 15, 2015, 6:35 pm*

It's the shading column causing the trouble. 'Width is a nil value'.

Removing line 81 and 82 fixes the script errors although your Scoreboard looks little bit untidy after that.