```
2c2
< // Swift source code of UserInterface.swift Version 9.0.9 at greenchess.com //
---
> // Swift source code of UserInterface.swift Version 9.0.8 at greenchess.com //
7a8
> import AVFoundation
39,41d39
<    // the placeholder text //
<    var Placeholder:String = ""
<
59,61d56
<
<    // the date picker function for a view //
<    var DateFunc:(String, Date) -> () = {(Key:String, GameDate:Date) in ()}
65,67d59
<
<    // the entered text field function //
<    var EnteredFunc:(String, String) -> () = {(Key:String, Entered:String) in ()}
110,119d101
<
<    // a screen view initializer for an entered text box control view //
<    init(_ Label:String, Placeholder:String, Key:String, XP:Double, YP:Double, Width:Double, Height:
<         Double, Window:NSWindow, EnteredFunc:@escaping(String, String) -> ())
<    {
<       // initialize the screen view values for an entered text box control view //
<       (self.Label, self.Placeholder, self.Key, self.XP, self.YP, self.Width, self.Height, self
<       .Window, self.EnteredFunc) = (Label, Placeholder, Key, XP, YP, Width, Height, Window,
<       EnteredFunc)
<    }
121,129d102
<    // a screen view initializer for date picker control view //
<    init(_ Label:String, Key:String, XP:Double, YP:Double, Width:Double, Height:Double, Window:
<         NSWindow, DateFunc:@escaping(String, Date) -> ())
<    {
<       // initialize the screen view values for a date picker control view //
<       (self.Label, self.Key, self.XP, self.YP, self.Width, self.Height, self.Window, self
<       .DateFunc) = (Label, Key, XP, YP, Width, Height, Window, DateFunc)
<    }
<
140,141c113,114
<    init(_ Window:NSWindow, ToolbarButtons:[(String, Color, String, String)], ToolbarFunc:
<         @escaping(Int) -> ())
---
>    init(_ XP:Double, YP:Double, Width:Double, Height:Double, Window:NSWindow, ToolbarButtons:
>         [(String, Color, String, String)], ToolbarFunc:@escaping(Int) -> ())
144c117,118
<       (self.Window, self.ToolbarButtons, self.ToolbarFunc) = (Window, ToolbarButtons, ToolbarFunc)
---
>       (self.XP, self.YP, self.Width, self.Height, self.Window, self.ToolbarButtons, self
>       .ToolbarFunc) = (XP, YP, Width, Height, Window, ToolbarButtons, ToolbarFunc)
195,197d168
<    // add padding to the top and bottom //
<    .padding([.bottom, .top], (ScreenObj.Inf.ToolbarButtons.count == 1) ? 7.5 : 0)
<
202,254d172
< // the date picker structure //
< struct DateStruct:View
< {
<    // the observed screen view object //
<    @ObservedObject var ScreenObj:Obj = Obj()
<
<    // the observed screen view initializer //
<    init(_ ScreenObj:Obj) {self.ScreenObj = ScreenObj}
<
<    // the main body //
<    var body:some View
<    {
<       // the date picker //
<       DatePicker("", selection:Binding(get:{ScreenObj.GameDate}, set:{GameDate in ScreenObj
<                 .GameDate = GameDate; ScreenObj.Inf.DateFunc(ScreenObj.Inf.Key, ScreenObj
<                 .GameDate)}), displayedComponents:[.date])
<
<       // set the view attributes //
<       .font(.system(size:15)).border(.gray, width:1).labelsHidden().frame(width:ScreenObj.Inf.Width,
<        height:ScreenObj.Inf.Height).position(x:ScreenObj.Inf.XP, y:ScreenObj.Inf.YP)
<    }
< }
<
< // the entered text field structure //
< struct EnteredStruct:View
< {
<    // the observed screen view object //
<    @ObservedObject var ScreenObj:Obj = Obj()
<
<    // the observed screen view initializer //
<    init(_ ScreenObj:Obj) {self.ScreenObj = ScreenObj}
<
<    // the main body //
<    var body:some View {HStack()
<    {
<       // the text field label //
<       Text(ScreenObj.Inf.Label)
<
<       // set the view attributes //
<       .font(.system(size:15)).frame(width:200, height:ScreenObj.Inf.Height, alignment:.trailing)
<
<       // the text field enter box //
<       TextField(ScreenObj.Inf.Placeholder, text:Binding(get:{ScreenObj.Entered}, set:{Entered in
<                ScreenObj.Entered = Entered; ScreenObj.Inf.EnteredFunc(ScreenObj.Inf.Key, ScreenObj
<                .Entered)})).textFieldStyle(.roundedBorder)
<
<       // set the view attributes //
<       .overlay(RoundedRectangle(cornerRadius:7, style:.continuous).strokeBorder(Color.gray,
<        lineWidth:1)).font(.system(size:14)).frame(width:ScreenObj.Inf.Width, height:ScreenObj.Inf
<        .Height)
<    } .position(x:ScreenObj.Inf.XP, y:ScreenObj.Inf.YP)}
< }
<
275,276c193
<        ScreenObj.Inf.XP, y:ScreenObj.Inf.YP).shadow(color:Color(red:0.25, green:0.25, blue:0.25),
<        radius:0.5, x:0, y:0)
---
>        ScreenObj.Inf.XP, y:ScreenObj.Inf.YP)
296c213
<       .font(.system(size:15)).frame(width:ScreenObj.Inf.Width, height:ScreenObj.Inf.Height,
---
>       .font(.system(size:15)).bold().frame(width:ScreenObj.Inf.Width, height:ScreenObj.Inf.Height,
344,346d260
<
<           // set the view attributes ///
<           .frame(width:ScreenObj.Inf.Width, height:ScreenObj.Inf.Height, alignment:.leading)
349,352c263,266
<      // set the view attributues //
<      .toggleStyle(SwitchToggleStyle()).controlSize(ControlSize.regular).font(.system(size:14))
<       .position(x:ScreenObj.Inf.XP, y:ScreenObj.Inf.YP)
<
---
>
>       // set the view attributes //
>       .toggleStyle(CheckboxToggleStyle()).controlSize(ControlSize.regular).font(.system(size:15))
>        .frame(width:ScreenObj.Inf.Width, height:ScreenObj.Inf.Height, alignment:.leading)
>        .position(x:ScreenObj.Inf.XP, y:ScreenObj.Inf.YP)
373,374c287,288
<                .MinV...ScreenObj.MaxV).shadow(color:Color(red:0.25, green:0.25, blue:0.25), radius:
<                0.5, x:0, y:0).controlSize(ControlSize.regular)
---
>                .MinV...ScreenObj.MaxV).shadow(color:.gray, radius:0, x:0, y:0)
>                .controlSize(ControlSize.small)
406,411d319
<
<    // the published entered text string //
<    @Published var Entered:String = ""
<
<    // the published date picker //
<    @Published var GameDate:Date = Date()
432,433c340,341
<       (Val, MinV, MaxV, Inf, ObjView) = (Double(InfVals.VI), Double(InfVals.VMinInt), Double(InfVals
<       .VMaxInt), InfSlide, NSHostingView(rootView:SliderStruct(self)))
---
>       (Val, MinV, MaxV, self.Inf, ObjView) = (Double(InfVals.VI), Double(InfVals.VMinInt),
>       Double(InfVals.VMaxInt), InfSlide, NSHostingView(rootView:SliderStruct(self)))
440c348
<       (self.Selection, Inf, ObjView) = (Selection, InfBar, NSHostingView(rootView:
---
>       (self.Selection, self.Inf, ObjView) = (Selection, InfBar, NSHostingView(rootView:
450c358
<       Inf.Window?.addTitlebarAccessoryViewController(TitleBar)
---
>       self.Inf.Window?.addTitlebarAccessoryViewController(TitleBar)
457c365
<       (self.Selection, Inf, ObjView) = (Selection, InfPick, NSHostingView(rootView:
---
>       (self.Selection, self.Inf, ObjView) = (Selection, InfPick, NSHostingView(rootView:
459,461d366
<
<       // add the object view to the window if the subwindow is empty //
<       if Inf.SubWindow.isEmpty {Inf.Window.contentView!.addSubview(ObjView)}
463,482d367
<
<    // the observable object initializer for an enter text view //
<    init(_ Entered:String, InfEnt:ScreenInfo)
<    {
<       // initialize the observable object enter text view attributes //
<       (self.Entered, Inf, ObjView) = (Entered, InfEnt, NSHostingView(rootView:EnteredStruct(self)))
<
<       // add the object view to the window //
<       Inf.Window.contentView!.addSubview(ObjView)
<    }
<
<    // the observable object initializer for a date picker view //
<    init(_ GameDate:Date, InfDte:ScreenInfo)
<    {
<       // initialize the observable object date picker view attributes //
<       (self.GameDate, Inf, ObjView) = (GameDate, InfDte, NSHostingView(rootView:DateStruct(self)))
<
<       // add the object view to the window //
<       Inf.Window.contentView!.addSubview(ObjView)
<    }
488c373,374
<       (self.Status, Inf, ObjView) = (Status, InfTogg, NSHostingView(rootView:ToggleStruct(self)))
---
>       (self.Status, self.Inf, ObjView) = (Status, InfTogg, NSHostingView(rootView:
>       ToggleStruct(self)))
495c381
<       (Inf, ObjView) = (InfImage, NSHostingView(rootView:ImageStruct(self)))
---
>       (self.Inf, ObjView) = (InfImage, NSHostingView(rootView:ImageStruct(self)))
502,505c388
<       (Inf, ObjView) = (InfText, NSHostingView(rootView:TextStruct(self)))
<
<       // add the object view to the window if the subwindow is empty //
<       if Inf.SubWindow.isEmpty {Inf.Window.contentView!.addSubview(ObjView)}
---
>       (self.Inf, ObjView) = (InfText, NSHostingView(rootView:TextStruct(self)))
512,513c395,396
<       (self.ColorView, ColorView.frame.origin, Inf, ObjView) = (ColorView, CGPoint(x:InfCol.OX, y:
<       InfCol.OY), InfCol, ColorView)
---
>       (self.ColorView, ColorView.frame.origin, self.Inf, ObjView) = (ColorView, CGPoint(x:InfCol.OX,
>       y:InfCol.OY), InfCol, ColorView)
520,521c403,404
<       (self.ClockView, ClockView.frame.origin, Inf, ObjView) = (ClockView, CGPoint(x:InfClk.OX, y:
<       InfClk.OY), InfClk, ClockView)
---
>       (self.ClockView, ClockView.frame.origin, self.Inf, ObjView) = (ClockView, CGPoint(x:InfClk.OX,
>       y:InfClk.OY), InfClk, ClockView)
551c434,437
<
---
>
>    // the pgn tags controller //
>    var PGNTags:PGNTagsController!
>
554,556d439
<
<    // the enter game info pgn tags toolbar information //
<    var TagsBar:[(String, Color, String, String)] = [("xmark.circle.fill", Color.gray, "", "")]
594c477
<
---
>
596c479
<    func AddNSViews(_ Board:GreenChess)
---
>    func AddNSViews(_ Board:GreenChess, PGNTags:PGNTagsController)
598,599c481,482
<       // set the board //
<       self.Board = Board
---
>       // set the board and pgn tags data //
>       (self.Board, self.PGNTags) = (Board, PGNTags)
621c504
<       // add the clock surface color view to the screen dictionary //
---
>       // all the clock surface color view to the screen dictionary //
670,671c553,554
<             ScreenInfo("Multithread Stockfish Engine", Key:Key, SubWindow:"PrefsStockfish", XP:155,
<             YP:-420, Width:210, Height:30, Window:Board.PreferencesWindow, ToggleFunc:Board
---
>             ScreenInfo("Multithread Stockfish Engine", Key:Key, SubWindow:"PrefsStockfish", XP:180,
>             YP:-420, Width:300, Height:30, Window:Board.PreferencesWindow, ToggleFunc:Board
677c560
<             155, YP:-345, Width:210, Height:30, Window:Board.PreferencesWindow, ToggleFunc:Board
---
>             180, YP:-345, Width:300, Height:30, Window:Board.PreferencesWindow, ToggleFunc:Board
693c576
<             ScreenInfo(Key, SubWindow:"PrefsBoard", XP:503, YP:-550, Width:165, Height:30, Window:
---
>             ScreenInfo(Key, SubWindow:"PrefsBoard", XP:515, YP:-550, Width:165, Height:30, Window:
699c582
<             SubWindow:"PrefsBoard", XP:503, YP:-595, Width:165, Height:30, Window:Board
---
>             SubWindow:"PrefsBoard", XP:515, YP:-595, Width:165, Height:30, Window:Board
734c617
<       Key = "BoardPreview"; Objs.updateValue(Obj(InfImage:ScreenInfo("PrefsBoard", XP:218, YP:-405,
---
>       Key = "BoardPreview"; Objs.updateValue(Obj(InfImage:ScreenInfo("PrefsBoard", XP:225, YP:-405,
739c622
<             ScreenInfo("Autosave Game", Key:Key, SubWindow:"PrefsBoard", XP:536, YP:-505, Width:174,
---
>             ScreenInfo("Autosave Game", Key:Key, SubWindow:"PrefsBoard", XP:555, YP:-505, Width:225,
745c628
<             Key:Key, SubWindow:"PrefsBoard", XP:536, YP:-470, Width:174, Height:30, Window:Board
---
>             Key:Key, SubWindow:"PrefsBoard", XP:555, YP:-470, Width:225, Height:30, Window:Board
750c633
<             ScreenInfo("Opening Book", Key:Key, SubWindow:"PrefsBoard", XP:536, YP:-435, Width:174,
---
>             ScreenInfo("Opening Book", Key:Key, SubWindow:"PrefsBoard", XP:555, YP:-435, Width:225,
756,757c639,640
<             ScreenInfo("Highlight Last Move", Key:Key, SubWindow:"PrefsBoard", XP:536, YP:-400,
<             Width:174, Height:30, Window:Board.PreferencesWindow, ToggleFunc:BPComp
---
>             ScreenInfo("Highlight Last Move", Key:Key, SubWindow:"PrefsBoard", XP:555, YP:-400,
>             Width:225, Height:30, Window:Board.PreferencesWindow, ToggleFunc:BPComp
762,763c645,646
<             ScreenInfo("Highlight Captures", Key:Key, SubWindow:"PrefsBoard", XP:536, YP:-365,
<             Width:174, Height:30, Window:Board.PreferencesWindow, ToggleFunc:BPComp
---
>             ScreenInfo("Highlight Captures", Key:Key, SubWindow:"PrefsBoard", XP:555, YP:-365,
>             Width:225, Height:30, Window:Board.PreferencesWindow, ToggleFunc:BPComp
768,769c651,652
<             ScreenInfo("Highlight Threats", Key:Key, SubWindow:"PrefsBoard", XP:536, YP:-330,
<             Width:174, Height:30, Window:Board.PreferencesWindow, ToggleFunc:BPComp
---
>             ScreenInfo("Highlight Threats", Key:Key, SubWindow:"PrefsBoard", XP:555, YP:-330,
>             Width:225, Height:30, Window:Board.PreferencesWindow, ToggleFunc:BPComp
774,775c657,658
<             ScreenInfo("Move by Selecting Square", Key:Key, SubWindow:"PrefsBoard", XP:536, YP:
<             -295, Width:174, Height:30, Window:Board.PreferencesWindow, ToggleFunc:BPComp
---
>             ScreenInfo("Move by Selecting Squares", Key:Key, SubWindow:"PrefsBoard", XP:555, YP:
>             -295, Width:225, Height:30, Window:Board.PreferencesWindow, ToggleFunc:BPComp
780,781c663,664
<             ScreenInfo("Display Captured Pieces", Key:Key, SubWindow:"PrefsBoard", XP:536, YP:-260,
<             Width:174, Height:30, Window:Board.PreferencesWindow, ToggleFunc:BPComp
---
>             ScreenInfo("Display Captured Pieces", Key:Key, SubWindow:"PrefsBoard", XP:555, YP:-260,
>             Width:225, Height:30, Window:Board.PreferencesWindow, ToggleFunc:BPComp
786c669
<             ScreenInfo("Rotate Board", Key:Key, SubWindow:"PrefsBoard", XP:536, YP:-225, Width:174,
---
>             ScreenInfo("Rotate Board", Key:Key, SubWindow:"PrefsBoard", XP:555, YP:-225, Width:225,
811,812c694,695
<             ScreenInfo("Display Diagrams in Game Moves", Key:Key, SubWindow:"PrefsDiagram", XP:162,
<             YP:-480, Width:225, Height:30, Window:Board.PreferencesWindow, ToggleFunc:Board
---
>             ScreenInfo("Display Diagrams in Game Moves", Key:Key, SubWindow:"PrefsDiagram", XP:180,
>             YP:-480, Width:300, Height:30, Window:Board.PreferencesWindow, ToggleFunc:Board
822,823c705,706
<             ScreenInfo("Display Game Clock", Key:Key, SubWindow:"PrefsClock", XP:118, YP:-555,
<             Width:135, Height:30, Window:Board.PreferencesWindow, ToggleFunc:Board
---
>             ScreenInfo("Display Game Clock", Key:Key, SubWindow:"PrefsClock", XP:180, YP:-555,
>             Width:300, Height:30, Window:Board.PreferencesWindow, ToggleFunc:Board
933,970d815
<
<       // add the event entered text field box to the screen dictionary //
<       Key = "EventTitle"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("Event", Placeholder:"", Key:Key, XP:270, YP:-610, Width:500, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the site location text field box to the screen dictionary //
<       Key = "SiteLocation"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("Site", Placeholder:"", Key:Key, XP:270, YP:-575, Width:500, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the date game played text to the screen dictionary //
<       Key = "DateGamePlayedText"; Objs.updateValue(Obj(InfText:ScreenInfo("Date", SubWindow:"",
<             XP:99, YP:-540, Width:50, Height:30, Window:Board.EnterInfoWindow)), forKey:Key)
<
<       // add the date game played date picker to the screen dictionary //
<       Key = "DateGamePlayed"; Objs.updateValue(Obj(Attributes.GetGameDate(Key), InfDte:
<             ScreenInfo("Date", Key:Key, XP:174, YP:-540, Width:100, Height:30, Window:Board
<             .EnterInfoWindow, DateFunc:Attributes.UpdateGameDate)), forKey:Key)
<
<       // add the round number text field box to the screen dictionary //
<       Key = "RoundNumber"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("Round", Placeholder:"", Key:Key, XP:70, YP:-505, Width:100, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the white name text field box to the screen dictionary //
<       Key = "WhiteName"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("White", Placeholder:"", Key:Key, XP:270, YP:-470, Width:500, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the black name text field box to the screen dictionary //
<       Key = "BlackName"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("Black", Placeholder:"", Key:Key, XP:270, YP:-435, Width:500, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the end result text to the screen dictionary //
<       Key = "EndResultText"; Objs.updateValue(Obj(InfText:ScreenInfo("Result", SubWindow:"", XP:95,
<             YP:-400, Width:50, Height:30, Window:Board.EnterInfoWindow)), forKey:Key)
972,1027d816
<       // add the end result picker view to the screen dictionary //
<       Key = "EndResult"; Objs.updateValue(Obj(Attributes.GetEndResult(Key), InfPick:ScreenInfo(Key,
<             SubWindow:"", XP:179, YP:-400, Width:125, Height:30, Window:Board.EnterInfoWindow,
<             Items:Attributes.EndResults, PickerFunc:Attributes.UpdateEndResult)), forKey:Key)
<
<       // add the time control text field box to the screen dictionary //
<       Key = "TimeControl"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("Time Control", Placeholder:"", Key:Key, XP:270, YP:-365, Width:500, Height:
<             30, Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:
<             Key)
<
<       // add the white elo text field box to the screen dictionary //
<       Key = "WhiteElo"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("White Elo", Placeholder:"", Key:Key, XP:70, YP:-330, Width:100, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the black elo text field box to the screen dictionary //
<       Key = "BlackElo"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("Black Elo", Placeholder:"", Key:Key, XP:70, YP:-295, Width:100, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the white uscf text field box to the screen dictionary //
<       Key = "WhiteUSCF"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("White USCF", Placeholder:"", Key:Key, XP:70, YP:-260, Width:100, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the black uscf text field box to the screen dictionary //
<       Key = "BlackUSCF"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("Black USCF", Placeholder:"", Key:Key, XP:70, YP:-225, Width:100, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the annotator text field box to the screen dictionary //
<       Key = "Annotator"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("Annotator", Placeholder:"", Key:Key, XP:270, YP:-190, Width:500, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the eco text field box to the screen dictionary //
<       Key = "ECO"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:ScreenInfo("ECO",
<             Placeholder:"", Key:Key, XP:70, YP:-155, Width:100, Height:30, Window:Board
<             .EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the opening text field box to the screen dictionary //
<       Key = "Opening"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("Opening", Placeholder:"", Key:Key, XP:270, YP:-120, Width:500, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the section text field box to the screen dictionary //
<       Key = "Section"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("Section", Placeholder:"", Key:Key, XP:270, YP:-85, Width:500, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
<       // add the board number text field box to the screen dictionary //
<       Key = "BoardNumber"; Objs.updateValue(Obj(Attributes.Inf.Vals(Key).VS, InfEnt:
<             ScreenInfo("Board", Placeholder:"", Key:Key, XP:70, YP:-50, Width:100, Height:30,
<             Window:Board.EnterInfoWindow, EnteredFunc:Attributes.UpdateEnteredText)), forKey:Key)
<
1030,1035c819,820
<             ScreenInfo(Board.PreferencesWindow, ToolbarButtons:PrefsBar, ToolbarFunc:Board
<             .PreferencesAction)), forKey:Key)
<
<       // add the enter game info pgn tabs toolbar to the screen dictionary //
<       Key = "TagsToolbar"; Objs.updateValue(Obj(1, InfBar:ScreenInfo(Board.EnterInfoWindow,
<             ToolbarButtons:TagsBar, ToolbarFunc:Board.ExitInfoWindow)), forKey:Key)
---
>             ScreenInfo(280, YP:-580, Width:100, Height:30, Window:Board.PreferencesWindow,
>             ToolbarButtons:PrefsBar, ToolbarFunc:Board.PreferencesAction)), forKey:Key)
```