



Forum: WD 9

Topic: recherche toutes infos pour création planning quotidien pour RDV

Subject: Re: recherche toutes infos pour création planning quotidien pour RDV

Posté par: JYM

Contribution le : 19/5/2006 12:08:07

Bonjour,

J'ai développé un planning horaire ,sur la base de l'exemple windev

Pour y arriver il faut juste se plonger dedans un ou deux

jour, après on fait a peut près ce que l'on veut quand on a compris le principe

(drag&drop,déplacer,alarme...)

ce qui ma mis sur la voie c'est ce post trouvé sur le newsgroup pcSoft.

(merci à Christian)

Cordialement,

JYM

Citation :

Salut

Je te conseille d'analyser l'exemple fourni par windev, il est très bien fait.

Pour tracer les heures dans la journée:

Dans la classe :TraceCadres

A la fin de ta classe remplace l'existant par :

```
// Séparateur demi-journées ou les heures dans la journée
SI :AfficheMidi ALORS
SI :bNoirEtBlanc ALORS dStylo(iNoir, 2) SINON dStylo(:CoulHeures, 2)
// selon (DateVersEntier(au)-DateVersEntier(du))
SELON :NombreColonnes
CAS <=:NombreJourAfficheHeures
//on divise la journée en heures
POUR i=0 A :NombreColonnes-1
// On trace un séparateur en pointillés à moitié de chaque jour
// x = (:LargeurMarge+(i * :LargeurColonne)) + (:LargeurColonne/2)
POUR j=0 A (Val(FinJournée) - Val(DebutJournée))-1
x=:LargeurMarge + :LargeurColonne*i+
(:LargeurColonne*j)/(Val(FinJournée)-Val(DebutJournée))
SI j=0 ALORS
SI i<>0 ALORS
//la première ligne de séparation n'est pas affichée
SI :bNoirEtBlanc ALORS dStylo(iNoir, 2) SINON
dStylo(:CoulHeures, 0)
dLigne( x, (:HauteurEntête*2)/3+1, x, :HauteurChamp )
FIN
SINON
```

```
SI :bNoirEtBlanc ALORS dStylo(iNoir, 2) SINON dStylo(:CoulHeures,
2)
dLigne( x, (:HauteurEntête*2)/3+1, x, :HauteurChamp )
FIN
```

FIN

FIN

AUTRE CAS

POUR i=0 A :NombreColonnes-1

// On trace un séparateur en pointillés à moitié de chaque jour

// x = (:LargeurMarge+(i * :LargeurColonne)) + (:LargeurColonne/2)

x = (:LargeurMarge+(i * :LargeurColonne)) +

(:LargeurColonne*(12-Val(DebutJournée))/(Val(FinJournée)-Val(DebutJournée)))

dLigne(x, :HauteurEntête+1, x, :HauteurChamp)

FIN

FIN

FIN

Explication:

:NombreJourAfficheHeures est un entier. Le principe est que si le nombre de jours affichés est inférieur à une certaine valeur, on affiche la séparation des heures

Pour tes tests, tu peux remplacer :NombreJourAfficheHeures avec 3

DébutJournée et FinJournée sont des entiers.

Par exemple DébutJournée peut être à 7 et FinJournée peut être à 20

Ainsi il va afficher les heures entre 7h00 et 20h00

```
////////////////////////////////////
```

```
////
```

Pour afficher le libellé des heures

C'est la classe TraceJours

Je te l'ai mis en entier, tu peux donc comparer ce qui existe avec ce qui est nouveau

PROCEDURE TraceJours()

// Pour parcourir chaque colonne

i est un entier

sHeureJournée est un entier

sHeureAffichée est un entier

sChaineHeure est une chaîne

// Pour le calcul de la position horizontale

xJour, xMois sont des entiers

yJour, yHeures sont des entiers

```

xHeure est un entier
// Date à afficher
IDate est un entier
// Pour construire la chaîne affichée
sChaineJour est une chaîne
// Numéro de jour du mois
sJour est une chaîne

//selon (DateVersEntier(au)-DateVersEntier(du))
SELON :NombreColonnes
CAS <=:NombreJourAfficheHeures
yJour= :HauteurEntête-2-( :HauteurEntête/3)
yHeures=:HauteurEntête-2
AUTRE CAS
yJour= :HauteurEntête-2
yHeures=-1
FIN

// On dessine le jour pour chaque colonne
POUR i = 0 A :NombreColonnes
IDate = :DateCourante + i

// Lettre du jour
sChaineJour = Gauche(EntierVersJourEnLettre(IDate),1)

// Numéro de jour
sJour = Droite(EntierVersDate(IDate),2)
sChaineJour += " "+sJour

// Position horizontale en prenant en compte la marge
xJour = :MargeEnteteX + :LargeurMarge+(i * :LargeurColonne)

// On sélectionne la police verticale
dPolice(:NomPoliceJour, :HauteurPoliceJour, :AttrPoliceJour, 90)
SI EntierVersJour(IDate)=6 OU EntierVersJour(IDate)=7 ALORS
// Samedi-Dimanche
SI :bNoirEtBlanc ALORS dTexte( xJour, yJour, sChaineJour, iNoir )
SINON dTexte( xJour, yJour, sChaineJour, :CoulNumJour )
SINON
SI EstFerie(IDate) ALORS
// Jour férié
SI :bNoirEtBlanc ALORS dTexte( xJour, yJour, sChaineJour, iNoir )
SINON dTexte( xJour, yJour, sChaineJour, :CoulNumJour )
SINON
// Autres jour : couleur par défaut
SI :bNoirEtBlanc ALORS dTexte( xJour, yJour, sChaineJour, iNoir )
SINON dTexte( xJour, yJour, sChaineJour, :CoulNumJour )
FIN

```

FIN

```
SI yHeures<>-1 ALORS
// On sélectionne la police verticale
dPolice(:NomPoliceJour, :HauteurPoliceJour, :AttrPoliceJour, 90)
//on écrit les heures
xHeure=xJour
POUR sHeureJournée=0 A (Val(FinJournée)-Val(DebutJournée))-1
sHeureAffichée=sHeureJournée+Val(DebutJournée)
sChaineHeure=Droite("0"+sHeureAffichée+":00",5)
xHeure=xJour+( :LargeurColonne)/(Val(FinJournée)-Val(DebutJournée))*sHeureJ
ournée
dTexte( xHeure, yHeures, sChaineHeure, :CoulNumJour )
FIN
FIN
```

```
// On change de mois alors on trace une séparation et on affiche le
nom du mois
SI sJour = "01" ALORS
xMois = :LargeurMarge + (i * :LargeurColonne)
SI :bNoirEtBlanc ALORS dLigne( xMois, 0, xMois, :PositionJours, iNoir
) SINON dLigne( xMois, 0, xMois, :PositionJours, :CoulCadres )
// On sélectionne la police horizontale
dPolice(:NomPoliceJour, :HauteurPoliceJour+2, :AttrPoliceJour, 0)
SI :bNoirEtBlanc ALORS dTexte( xMois+:MargeEnteteX, :MargeEnteteY,
EntierVersMoisEnLettre(IDate), iNoir ) SINON dTexte(
xMois+:MargeEnteteX, :MargeEnteteY, EntierVersMoisEnLettre(IDate),
:CoulMois )
FIN
FIN
```

```
// On sélectionne la police horizontale
dPolice(:NomPoliceJour, :HauteurPoliceJour+2, :AttrPoliceJour, 0)
// On affiche le mois et l'année
SI :bNoirEtBlanc ALORS
dTexte( :MargeEnteteX, :MargeEnteteY,
Gauche(EntierVersDate(:DateCourante),4), iNoir )
dTexte( :MargeEnteteX+:LargeurMarge, :MargeEnteteY ,
EntierVersMoisEnLettre(:DateCourante), iNoir )
SINON
dTexte( :MargeEnteteX, :MargeEnteteY,
Gauche(EntierVersDate(:DateCourante),4), :CoulMois )
dTexte( :MargeEnteteX+:LargeurMarge, :MargeEnteteY ,
EntierVersMoisEnLettre(:DateCourante), :CoulMois )
FIN
```

Voilà, j'espère que ca va t'aider.

A+

--

Bon développement à tous
Christian