

Apprendre la programmation multiplateforme avec Delphi

30 avril 2020

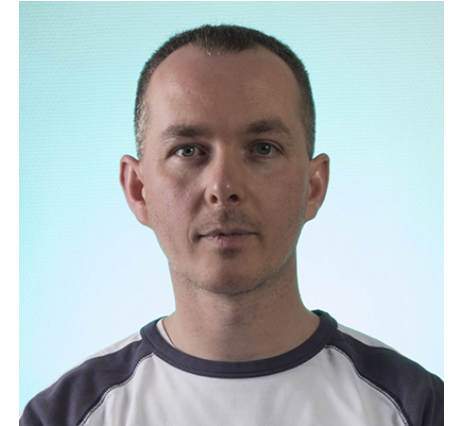
#DelphiBootCamp



Préambule



Patrick Prémartin

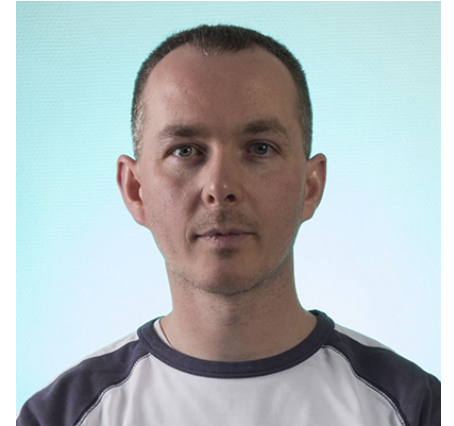


- MVP Embarcadero, prestataire informatique et formateur Delphi
- Entreprise Olf Software
(conseil, développement et assistance technique) :
<https://olfsoftware.fr>
- Formations Delphi en entreprise ou en groupe :
<https://se-former-a-delphi.fr>
- Blog sur l'actualité du développement en Pascal avec Delphi :
<https://developpeur-pascal.fr>



Pour contacter Patrick Prémartin

- Le blog :
<https://developpeur-pascal.fr/contact/>
- Twitter :
<https://twitter.com/premartinpatric>
- LinkedIn :
<https://www.linkedin.com/in/patrickpremartin/>
- Par téléphone uniquement sur rendez-vous ou (vraies) urgences de clients existants.



Fonctionnement du bootcamp

- Ce bootcamp a pour objectif de vous montrer comment faire de la programmation de logiciels et applications mobiles en utilisant le Pascal et Delphi.
- Je vous donne rendez-vous en direct du lundi au vendredi entre 10 heures et midi (heure française) pour vous faire découvrir le développement logiciel.
- Je répondrai à vos questions en fin de chaque chapitre.



Fonctionnement du bootcamp

- Pour les absents et la postérité ce bootcamp est enregistré.
- Les rediffusions accompagnées des PDF de présentation et des sources des exemples sont disponibles sur le site

<https://apprendre-delphi.fr>

et la chaîne YouTube de Barnsten

<https://vasur.fr/delphibootcamp>



Fonctionnement du bootcamp

- Pour discuter entre vous, partager vos impressions et votre progression sur les réseaux sociaux, vous pouvez utiliser #DelphiBootCamp
- Je répondrai aux questions posées avec ce hashtag sur Twitter, Facebook et LinkedIn durant toute la durée du bootcamp.
- Famille, amis et collègues sont les bienvenus s'ils sont intéressés. Les inscriptions restent possibles depuis <https://apprendre-delphi.fr>



Qui est concerné par ce bootcamp ?

- Toute personne désireuse de faire de la programmation peut reprendre ce bootcamp du début afin d'apprendre les bases du développement logiciel et le langage Pascal.
Les rediffusions des sessions précédentes sont là pour ça.
- Pour suivre la phase 2 de cette formation il faut avoir des connaissances en développement.
- Des connaissances du langage Pascal ou de la VCL peuvent être utiles mais ne sont pas obligatoires.



Découpage de cette formation

- Cette formation n'est pas le MCU mais elle est quand même découpée en deux phases :
 - La phase 1 concernait la prise en main de Delphi, du Pascal et de la programmation pour Windows avec la VCL.
 - La phase 2 concerne la programmation multiplateforme en utilisant le framework FireMonkey de Delphi.
 - La phase 3 on ne sait pas encore, ça dépend de vous...



Matériel nécessaire

- Pour suivre cette formation il ne vous faut qu'un navigateur web et une connexion Internet. Cela vous permettra de suivre les sessions en direct et de voir les rediffusions.
- Pour pratiquer, en revanche, vous devrez posséder une version de Delphi et l'exécuter sous Windows (sur un PC ou dans une machine virtuelle).
- Pour développer sur d'autres appareils (smartphones et tablettes par exemple) il vous faudra un appareil de test à connecter à votre ordinateur.



Matériel nécessaire

- Le formateur utilise une édition Entreprise de Delphi 10.3.3 Rio.
- Les exemples proposés fonctionnent dans cette version.
- Aucune garantie n'est proposée pour les versions antérieures (ni les suivantes mais en 25 ans Delphi a largement démontré sa capacité à garder une compatibilité ascendante).
- L'ordinateur utilisé est une machine virtuelle Windows gérée par VM Ware Fusion sur un iMac de 2011.



Où trouver des infos ?



Liens utiles de cette formation en ligne

- Pour voir les rediffusions de de la formation :
 - Sur la chaîne YouTube de Barnsten :
<https://vasur.fr/delphibootcamp>
 - Sur Vimeo :
<https://vimeo.com/channels/1552230>



Liens utiles de cette formation en ligne

- Téléchargement des présentations en PDF :
<https://apprendre-delphi.fr/apprendre-la-programmation-avec-delphi/>
- Sources des exemples et corrigés :
<https://github.com/DeveloppeurPascal/ApprendreLaProgrammationAvecDelphi>



Liens utiles pour la formation

- Formations individuelles chez soi :
<https://apprendre-delphi.fr>
=> pour le moment juste celle que vous suivez en ce moment
- Formations en entreprise ou en groupe :
<https://se-former-a-delphi.fr>
=> de nouvelles dates et programmes dès qu'on peut
=> des formations personnalisées sur demande, en ligne ou en présentiel



Liens utiles pour s'informer

- Développeur Pascal
<https://www.developpeur-pascal.fr>
=> actualité de Delphi, articles de fond, trucs et astuces
- Awsome Pascal
<https://github.com/Kromster80/awesome-pascal>
=> une liste d'outils, librairies, composants et logiciels liés au développement en Pascal et Delphi



Liens utiles pour s'informer

- Webinaires mensuels en français sur une thématique précise :
<https://developpeur-pascal.fr/p/6007-webinaires.html>
- D'autres sources d'infos et ressources utiles :
<https://developpeur-pascal.fr/c/4-ressources.html>



Liens utiles pour s'informer

- Agrégateur d'infos et de blogs liés au Pascal et Delphi :
<http://www.delphimagazine.com>
- Blaise Pascal Magazine (en anglais) :
<https://www.blaisepascalmagazine.eu>
=> magazine disponible en ligne dédié à la programmation en Pascal.
- Programmez! (en français) :
<https://www.programmez.com>
=> magazine mensuel disponible en kiosques et en ligne dédié au développement logiciel dans tous les langages avec des articles ponctuelles sur Delphi et la programmation en Pascal.



Liens utiles d'Embarcadero

- Embarcadero :
<https://www.embarcadero.com/fr/>
- Documentation en ligne « officielle » :
<http://docwiki.embarcadero.com>
- Actualité, blogs, forums « officiels » (en anglais) :
<https://community.idera.com/developer-tools/b/blog>



Liens utiles d'Embarcadero

- Embarcadero sur Twitter :
<https://twitter.com/EmbarcaderoTech>
- Embarcadero sur Facebook :
<https://www.facebook.com/embarcaderotech/>
- Embarcadero sur YouTube :
<https://www.youtube.com/user/EmbarcaderoTechNet/featured>



Liens utiles d'Embarcadero

- Learn Delphi :
<https://www.learndelphi.org>
- Embarcadero Academy :
<https://www.embarcaderoacademy.com>
- 25 ans de Delphi
<https://delphi.embarcadero.com>



Liens utiles d'Embarcadero

- Portail qualité :
<https://quality.embarcadero.com/>
=> signaler et consulter les anomalies recensées
=> faire des demandes de nouvelles fonctionnalités
- Portail de téléchargement et licences :
<https://my.embarcadero.com>



Liens utiles d'Embarcadero

- Delphi
<https://www.embarcadero.com/fr/products/delphi>
- Téléchargement de Delphi Trial
<https://www.embarcadero.com/fr/products/delphi/start-for-free>
- Téléchargement de Delphi Community Edition
<https://www.embarcadero.com/fr/products/delphi/starter>



Liens utiles de Barnsten

- Barnsten :
<https://www.barnsten.com/fr/>
- Barnsten sur Twitter :
<https://twitter.com/BarnstenFR>
- Barnsten sur Facebook :
<https://www.facebook.com/EmbarcaderoFrance/>



Liens utiles de Barnsten

- Barnsten sur YouTube :
<https://www.youtube.com/channel/UCSrWP5Y01uSHS3014Pfprlw/featured>
- Rediffusion des webinaires :
https://www.youtube.com/watch?v=sgPSimYtzYc&list=PLHLdMyq6m8_sTwOTGwziNlcn443POnmDd
- Rediffusion de cette formation :
https://www.youtube.com/watch?v=2bXx63WQank&list=PLHLdMyq6m8_s-OxVA_qZ4O_YvVgl_1nOp



A savoir pour coder...

- Raccourcis au clavier dans Delphi :
<https://developpeur-pascal.fr/p/005-les-combinaisons-de-touches-et-raccourcis-clavier-utiles-dans-l-edi-de-delphi.html>



Quelques banques d'images

- Icons8
<https://icons8.com>
- Material Design Icons
<http://materialdesignicons.com>
- GlyFX
<http://www.glyfx.com>



Quelques banques d'images

- Kenney Studio
<https://kenney.nl/assets>
- Adobe Stock
<https://stock.adobe.com/fr/>



Librairies open source

- Icon fonts image List
<https://github.com/EtheaDev/IconFontsImageList>
=> projet open source permettant d'utiliser des caractères d'une fonte TTF comme images dans une liste d'images compatible avec les composants VCL et FMX.
- TFrameStand et TFormStand
<https://github.com/andrea-magni/TFrameStand>
=> pour animer vos écrans FireMonkey et créer des boîtes de dialogues sous forme de cadres presque sans codage.



Librairies open source

- De nombreux exemples de projets FMX
<https://github.com/FMXExpress/Cross-Platform-Samples>



Editeurs de textes

- Notepad++
<https://notepad-plus-plus.org>
- GBEPad
<http://www.gbsoft.fr/gbepad.php>
- Sublime Text
<https://www.sublimetext.com>



Logiciels de conception de bases de données

- TMS Data Modeler
<https://www.tmssoftware.com/site/tmsdm.asp>
- Aqua Data Studio
<https://www.aquafold.com/aquadatastudio>
- E/R Studio
<https://www.idera.com/er-studio-enterprise-architecture-solutions>
- Heidi SQL
<https://www.heidisql.com>



Liens utiles pour les bases de données

- SQLite
<https://www.sqlite.org/>
- DB Browser for SQLite
<https://sqlitebrowser.org>
- Les bases de données relationnelles
<https://info.blaisepascal.fr/cpge-bases-de-donnees-relationnelles>
- Cours sur le langage SQL
https://www.w3schools.com/sql/sql_intro.asp



Dans les épisodes précédents...

Phase 2 : Pascal, Pascal Objet, FMX et FireDAC



Jour 15 - Session du 15 avril 2020

- Lors de cette session nous avons vu les points suivants :
 - Introduction à la phase 2 de cette formation
 - Licences Delphi
 - Où trouver des infos ?
 - Dans les épisodes précédents... de la phase 1
 - Rappels sur Delphi, le langage Pascal et le langage Object Pascal
 - Présentation de FireMonkey
 - Préparer Delphi pour le développement multiplateforme
 - Un premier projet



Jour 16 - Session du 16 avril 2020

- Lors de cette session nous avons vu les points suivants :
 - Projets FMX
 - L'assistant de création de projets
 - Le gestionnaire de projets
 - Les options de projet
 - Le concepteur de fiches FMX
 - Des exemples d'écrans FMX
 - Différences entre FMX et VCL
 - Les fiches multiplateforme
 - Second projet multiplateforme



Jour 17 - Session du 17 avril 2020

- Lors de cette session nous avons vu les points suivants :
 - Les propriétés des composants FMX
 - Les événements des composants FMX
 - Des composants FMX de base
 - Exercice 6 : faites vos jeux



Jour 18 - Session du 20 avril 2020

- Lors de cette session nous avons vu les points suivants :
 - Retour sur la session 17
 - La compilation conditionnelle
 - Les animations
 - Les effets
 - Moteur physique : Box 2D
 - La 3D dans un projet FireMonkey



Jour 19 - Session du 21 avril 2020

- Lors de cette session nous avons vu les points suivants :
 - Les sprite sheets
 - Mettre un peu d'ambiance
 - Correction exercice 6 : le jeu du pendu (première partie)



Jour 20 - Session du 22 avril 2020

- Lors de cette session nous avons vu les points suivants :
 - Les cadres dans FMX
 - Correction exercice 6 : le jeu du pendu (suite et fin)
 - Les modules de données FMX
 - Les bases de données avec FMX
 - LiveBindings



Jour 21 - Session du 23 avril 2020

- Lors de cette session nous avons vu les points suivants :
 - Des écrans dans un programme



Jour 22 - Session du 24 avril 2020

- Lors de cette session nous avons vu les points suivants :
 - Rappels sur la portée des déclarations
 - Installation de Navigator
 - Exemples pour FireMonkey de FMX Express sur Github
 - Prévisualiser les écrans
 - TImageList de FMX
 - Programmation synchrone ou asynchrone ?
 - Les méthodes anonymes
 - Exercice 7a : répertoire téléphonique FMX
 - Exercice 7b : liste de courses FMX



Jour 23 - Session du 27 avril 2020

- Lors de cette session nous avons vu les points suivants :
 - Les boites de dialogue « simples » en FMX
 - TPath, TFile, TDirectory de l'unité System.IOUtils
 - Les fichiers binaires



Jour 24 - Session du 29 avril 2020

- Lors de cette session nous avons vu les points suivants :
 - Correction exercice 7b : liste de courses FMX
 - Les listes d'actions
 - Exemple d'utilisation d'une barre de progression dans une boucle avec blocage de l'interface utilisateur (en asynchrone)



Les processus



Les processus

- Depuis longtemps Delphi permet de gérer les processus dans nos programmes et avec des ordinateurs à plusieurs noyaux et plusieurs threads, il est maintenant possible d'en bénéficier pleinement.
- Les processus sont cependant un vaste sujet difficile de traiter en quelques minutes, je vais donc vous donner des informations de base et vous laisserai vous documenter de votre côté.



Les processus

- Les processus et tâches permettent d'exécuter du code en même temps que l'interface utilisateur. On n'est pas vraiment limité en nombre : le système d'exploitation est censé se débrouiller par lui-même quel que soit l'appareil.
- Les processus sont gérés par la classe TThread de System.Classes
- On peut en créer un nouveau en héritant de cette classe manuellement ou à l'aide de l'assistant « nouveau fichier ».



Les processus

- On peut également créer un processus anonyme avec « `TThread.CreateAnonymousThread().Start;` » pour lancer un bloc de code en tâche de fond à l'aide d'une méthode anonyme.
- Les processus s'exécutent sur le même cœur que le programme. Pour profiter pleinement de votre ordinateur il est maintenant recommandé d'utiliser des tâches avec l'unité `System.Threading`.



Les processus

- Comme pour les processus on peut créer une tâche en héritant de la classe TTask.
- On peut aussi en créer ponctuellement avec « TTask.Run(); »
- Delphi nous permet de gérer des pools de tâches et suivre leur progression globalement ou à l'unité.



Les processus

- Autre méthode de classe utile dans les tâches : « TParallel.For » qui permet de soumettre une boucle en tâche de fond.
- Par contre « TParallel.For » ne garanti pas l'ordre de passage des itérations de la boucle, donc chaque niveau de la boucle doit être indépendant des autres.



Les processus

- Une chose à avoir en tête lorsqu'on fait des processus et tâches c'est qu'on est susceptible de générer des violations d'accès et des plantages de nos programmes si n'y prend pas garde.
- TOUTE action de la part d'un processus vis-à-vis de l'interface utilisateur doit être synchronisée avec celle-ci. Je dis bien TOUTE !
- Pour cela on a un certain nombre de solutions, y compris pour échanger des informations entre processus.



Les processus

- Retenez surtout que si vous voulez accéder à l'interface utilisateur depuis un processus vous devez utiliser:
 - « TThread.Synchronize » pour bloquer le processus, faire le travail à synchroniser et reprendre le processus.
 - « TThread.Queue » pour faire le traitement à synchroniser plus tard (ou maintenant si possible).
 - « TThread.ForceQueue » pour traiter le code à synchroniser comme un événement, dans la boucle normale de traitement.



Les processus

- En plus de synchroniser le code avec l'interface utilisateur il y a aussi des cas où on doit verrouiller l'accès à certaines ressources de nos programmes (fichiers, variables globales, objets).
- Delphi fourni les solutions classiques :
 - MUTEX
 - Sémaphores
 - Sections critiques
- On a aussi une classe TMonitor pour protéger des objets.



L'instruction « case »



L'instruction « case »

- « case » est une instruction du langage Pascal.
- Un « case » est une façon de simplifier des « if » en chaîne lorsqu'on teste la même variable avec des valeurs numériques ou énumérées.
- Plutôt que des « if ... then ... else if ... then ... else if ... then ... » on utilise un « case ».



L'instruction « case »

- On peut aussi utiliser « case » dans le type RECORD afin de déclarer la même zone mémoire avec des champs de types différents. Je n'aborderai pas la question ici, reportez-vous à la documentation du type RECORD dans la documentation du langage Pascal et de Delphi.



Les types énumérés et les ensembles



Les types énumérés et les ensembles

- Dans nos exemples en VCL ou FireMonkey nous avons codé à plusieurs reprises les alignements de façon manuelle.
- La propriété Align est de type TAlignLayout dans FireMonkey et de type TAlign dans la VCL.
- Dans les deux cas c'est un type énuméré : on saisit un nom pour lui donner une valeur et ce nom est pris dans une liste de mots définis au niveau du type.



Les types énumérés et les ensembles

- C'est ce qu'on appelle une énumération et donc un type énuméré.
- Deux possibilités pour y accéder dans Delphi : soit directement avec les valeurs (comme c'est le cas pour le TAlign de la VCL), soit avec le nom du type en premier (comme c'est le cas pour le TAlignLayout de FMX).
- Cela dépend de la directive de compilation SCOPEDENUMS à mettre à ON dans le cas où on veut tout écrire (ce qui est recommandé afin de simplifier la saisie).



Les types énumérés et les ensembles

- Les ensembles sont simplement des types qui acceptent plusieurs valeurs d'un type énuméré.
- Nous en avons déjà croisés :
 - Les listes de lettres dans les programmes d'exemples sur la VCL
 - Les activations et désactivations de valeurs de la propriété StyleSettings dans FMX.
- Un ensemble est un « set of XXX » avec « XXX » correspondant à un type énuméré.



Les types énumérés et les ensembles

- Dans Delphi les valeurs des types énumérés sont en fait des entiers commençant à 0 en partant de la gauche de nos listes de valeurs.
- On peut modifier ce comportement en fixant une valeur numérique sur les noms qui nous intéressent ou tous (ce qui est fortement recommandé si on les stocke dans des fichiers ou bases de données).
- Reportez-vous à cet article si vous désirez passer de la valeur au texte et inversement :
- <https://developpeur-pascal.fr/p/0001-jouons-un-peu-avec-les-types-enumeres-et-leurs-valeurs.html>



La fonction « ifthen »



La fonction « ifthen »

- Dans certains langages on a un opérateur d'affectation permettant de réaliser un test. Cela n'existe pas en Pascal.
- Dans Delphi nous avons en revanche une fonction « ifthen() » qui fait la même chose : on lui passe un test ou un booléen en premier paramètre puis les valeurs à retourner si c'est vrai ou si c'est faux.
- Ifthen() est disponible dans l'unité System.StrUtils et dans l'unité System.Math selon les types de valeurs à renvoyer.



Les types génériques



Les types génériques

- Lors des exemples sur les programmes avec plusieurs écrans j'ai voulu utiliser une méthode avec un type générique pour gérer les écrans à afficher mais ça ne fonctionnait pas car je n'avais pas correctement déclaré les types des 3 écrans. J'avais laissé pour utiliser une méthode à l'ancienne mais ai laissé mon code en commentaires.
- Reprenons cet exemple de programme avec plusieurs cadres comme écrans et finissons le avec les bons types génériques.



Les types génériques

- Les types génériques sont apparus avec la version 2009 de Delphi.
- Ils permettent de gérer du code source sans savoir exactement sur quoi on va travailler... ou plutôt en sachant ce qui nous est nécessaire pour travailler et sans se préoccuper de la réalité à la compilation ou l'exécution du programme.



Les types génériques

- On parle de types génériques car c'est une façon de décrire des procédures, des fonctions, des classes, des interfaces, des structures et des méthodes sans dire lors de leur déclaration sur quoi ils portent.
- C'est à l'utilisation qu'on fixe les types réellement utilisés.
- Par convention on utilise « `<T>` », « `<T1,T2>` », « `<T1, T2, T3>` », ... dans les déclarations.



Les collections



Les collections

- Les collections sont disponibles dans System.Collections pour la version standard et System.Collections.Generic pour la version la plus récente et évoluée.
- On peut utiliser des collections pour éliminer les traitements que l'on faisait sur des tableaux.
- On a ainsi des listes, des files, des piles, des dictionnaires.



Les collections

- Pour utiliser les collections standard on doit hériter des classes proposées ou gérer des pointeurs vers les types de données à traiter.
- Avec les collections génériques on peut stocker ce que l'on veut sans se poser vraiment de question.

Le compilateur gère la bonne implémentation et quand il ne le peut pas la librairie nous demande de passer la méthode générique permettant de traiter les données (comme par exemple lors d'une comparaison).



Les collections

- Vous l'aurez sans doute compris : je recommande d'utiliser les collections génériques dans vos programmes lorsque vous avez des choses complexes à stocker qui n'auraient pas leur place dans un tableau ou un tableau dynamique.



A suivre...

La rediffusion de cette session sera disponible
sur <https://vasur.fr/delphibootcamp>

Discutez sur les réseaux sociaux avec le hashtag
#DelphiBootCamp

