



Analyste Programmeur en Automatisme, Robotique et Informatique Industrielle TS ARII

Module MF 2.7

Développer un programme en langage objet

Programmer dans un langage orienté objet

Programmation WINDEV

Patrick MONASSIER

MF 2.7 – Développer un programme en langage objet

Compétences

Traduire un modèle dans un langage objet

Objectifs

Programmer dans un langage orienté objet

Contenu

- *Le langage Windev*
 - *Transcription d'un algorithme*
 - *Traduction d'un modèle objet*

PC SOFT® 

WINDEV®

POO et WINDEV

Dans le cadre de ce cours, une formation à Windev est mise en place pour prendre en main la programmation objet dans le cadre d'un AGL. Pour ce faire, les étudiants chargeront sur leur PC la version gratuite et illimitée dans le temps de WINDEV EXPRESS 25.

WINDEV EXPRESS 25 : Gratuit et illimité dans le temps

Testez cette version Express, vos gains sont évidents :

- rapidité de développement étonnante,
- couverture fonctionnelle sans limite,
- intégration totale de votre SI et de votre existant,
- qualité de l'UI & UX sans égale.

TÉLÉCHARGEZ

(Gratuit et illimité
dans le temps)

Attention : après ce test, votre perception du développement va changer !

Téléchargement depuis Internet Explorer :

Ne pas "exécuter" directement le package d'installation.

Utiliser l'option "enregistrer" ou "enregistrer sous".

Une fois le téléchargement effectué, choisir "Ouvrir le dossier" et lancer l'exécutable téléchargé.

On utilise ce logiciel pour renforcer les concepts de programmation étudiés en année 1 et **développer les compétences en programmation objet**

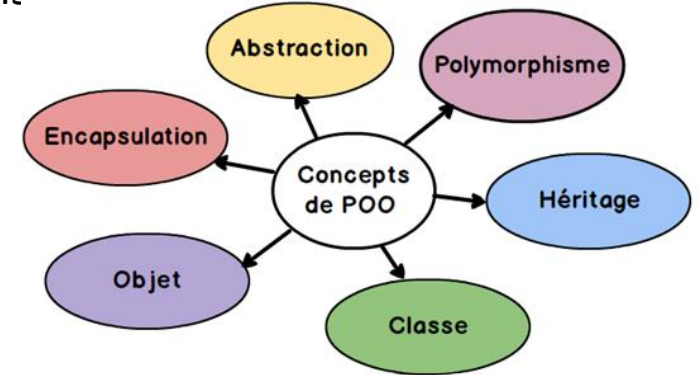
Le support de cours s'appuie sur la **documentation en ligne de Windev** qui fournit de nombreuses explications et des exemples pratiques d'implémentation

POO et WINDEV

WINDEV EXPRESS 25 est utilisé comme support de cours actif. Le cours se déroule sous forme d'exposé oral et de travaux pratiques dirigés, en utilisant les documentations de formation PC Soft et les exemples fournis par l'éditeur.

Ce powerpoint contient :

- Un rappel sur les concept de la POO
- Les liens sur les pages de formations et d'exemples fournies par l'éditeur
- Le forum Windev communauté des développeurs Windev



PCSOFT.

WINDEV
Développement
Windows, Linux
& .Net

Suite logicielle de développement d'applications **cross-plateformes**



Présentation de la POO (Programmation Orientée Objet)

La programmation orientée objet (appelée "POO") est une méthode de programmation dans laquelle les programmes sont organisés comme des ensembles d'objets. Chaque objet représente une instance d'une certaine classe, toutes les classes étant des membres d'une hiérarchie de classes unifiée par des relations d'héritage. La POO utilise donc des objets.

Chaque objet est une instance d'une certaine classe. Les classes sont reliées l'une à l'autre par des relations d'héritage.



Exemples didactiques (WINDEV) : WD POO Simple

[+] L'exemple "WD POO Simple" est un exemple didactique sur la POO avec WINDEV.

Concepts de la POO

Classe

Une classe rassemble la description d'une structure de données (les membres) et les traitements (les méthodes) qui manipulent cette structure. Une classe définit donc un type de donnée et son comportement.

WLangage et POO

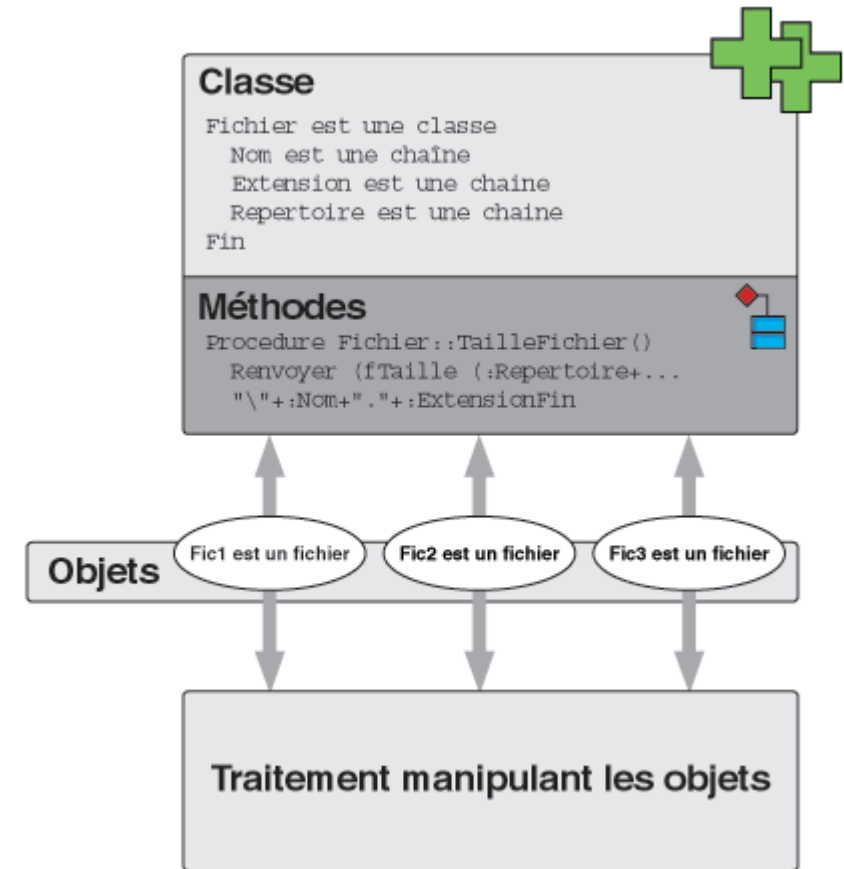
Le WLangage est orienté objet, en effet : il supporte les objets. les objets ont une classe associée les classes peuvent hériter d'attributs venant de "super classes » .

Important : L'objet de ce chapitre n'est pas de décrire les concepts de la programmation orientée objet, mais uniquement de décrire comment déclarer les classes, les méthodes et les objets en WLangage. Ce chapitre donne les syntaxes de la POO en WLangage et le principe de la création d'un programme orienté objet en WLangage. Le lecteur est supposé connaître les classes, les objets, les héritages, ... Si vous ne connaissez pas la POO, nous vous conseillons de lire des manuels spécialisés sur la POO avant de lire la suite de cette documentation.

Objet

Une classe permet de créer des objets. Chaque objet créé possède les membres décrits dans sa classe et peut-être manipulé par les méthodes de sa classe. On dit qu'un "objet est une instance de la classe". Une classe peut être considérée comme un modèle qui définit les membres et les méthodes communes à plusieurs objets.

Un membre est un paramètre de l'objet. Une méthode permet d'agir sur l'objet, pour modifier ses membres par exemple.



Constructeur et destructeur

La notion de Constructeur et Destructeur est importante puisque elle permet un appel automatique de méthode lors de la création d'un objet et lors de sa destruction.

La méthode Constructeur associée à une classe est automatiquement appelée lors de la déclaration d'un objet de la classe. Cela permet de s'assurer que les traitements d'initialisation de l'objet (affectation des membres, par exemple) ne seront pas oubliés par le développeur.

La méthode Destructeur associée à une classe est automatiquement appelée lors de la suppression de l'objet (sortie de procédure dans laquelle l'objet a été déclaré). Cela permet de libérer sans risque d'oubli les ressources utilisées par l'objet (zone mémoire, par exemple). Elle peut aussi être utilisée pour mettre à jour un fichier relatif à l'objet

Héritage

L'héritage permet d'inclure les caractéristiques d'une classe existante (classe de base) dans une nouvelle classe (classe dérivée). Cela permet de créer un nouveau type de donnée à partir d'un type connu, dans le but de lui ajouter des fonctionnalités, ou d'en modifier le comportement. La classe de base ne sera donc pas modifiée. Une classe peut hériter d'une ou de plusieurs autres classes dont elle devient une sous-classe. Une classe dérivée hérite donc des membres et des méthodes de sa ou ses classes mères, en plus de ses propres membres et méthodes. Il n'est pas nécessaire de dupliquer les membres et méthodes de la ou des classes mères

Encapsulation de données

L'encapsulation des données est sans aucun doute la notion la plus importante de la POO. Cette technique permet de garantir que les données membres de l'objet ne seront pas modifiées à tort par des fonctions (méthodes) extérieures à l'objet. Il est ainsi possible d'interdire à un utilisateur d'un objet l'accès à certains ou à tous ses membres. Les membres dont l'accès est interdit sont appelés membres privés. Il n'est possible d'y accéder qu'à partir des méthodes prévues à cet effet dans la classe.

Documentation PC Soft en ligne – *liens importants*

https://doc.pcsoft.fr/?6010008	Concepts de la POO
https://doc.pcsoft.fr/?6010009	Présentation de la POO
https://doc.pcsoft.fr/?6010010	Utilisation des objets
https://doc.pcsoft.fr/?6010006	Classes, membres, méthodes et propriétés
https://doc.pcsoft.fr/?6010001	Constructeurs et destructeurs
https://doc.pcsoft.fr/?6010002	Instanciation d'un objet
https://doc.pcsoft.fr/?6010005	Présentation d'un objet .NET

Analyste Programmeur en Automatismes, Robotique et
Informatique Industrielle
TS ARII

Module MF 2.7

Développer un programme en langage objet

Programmer dans un langage orienté objet

Programmation WINDEV

Fin de Présentation