

# Formation : Langage Perl, programmation avancée

Formation pratique - 2j - 14h00 - Réf. LAP  
Prix : 1470 CHF H.T.

Ce cours vous apprendra à améliorer la performance de vos programmes et à les maintenir efficacement. Vous maîtriserez les principales primitives du noyau, la création d'interfaces graphiques et de masques de saisie. Vous découvrirez enfin l'approche plus moderne de la programmation Objet.

## Objectifs pédagogiques

À l'issue de la formation, le participant sera en mesure de :

- ✓ Manipuler les fonctions systèmes avec Perl
- ✓ Mettre en oeuvre les concepts Objet
- ✓ Utiliser Perl/Tk
- ✓ Mettre en oeuvre GTK

## Public concerné

Informaticiens concernés par des procédures d'exploitation en environnements Unix, Linux ou Windows.

## Prérequis

Bonnes connaissances du langage Perl ou connaissances équivalentes à celles apportées par le cours "Langage Perl" réf. PRL.

## Modalités d'évaluation

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques...

Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

## Programme de la formation

### PARTICIPANTS

Informaticiens concernés par des procédures d'exploitation en environnements Unix, Linux ou Windows.

### PRÉREQUIS

Bonnes connaissances du langage Perl ou connaissances équivalentes à celles apportées par le cours "Langage Perl" réf. PRL.

### COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

### MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques...

Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

## 1 Programmation système

- Rappel des principes des processus : fork et exec.
- Rappel des principes de la gestion des signaux.
- Envoyer un signal à des processus : la fonction kill.
- Gérer les signaux : le pragma sigtrap, installer un gestionnaire de signal.
- Programmation des sockets et services de IO::Socket à Net::Server.
- Présentation de la programmation des threads.

### Travaux pratiques

Écriture de programmes Perl pour manipuler les fonctions systèmes.

## 2 Perl et l'Objet

- Quelques définitions : Programmation Orientée Objet, classe, instance, constructeur, destructeur, héritage...
- Mise en oeuvre de l'Objet : le paquetage pour la classe, le hachage pour les attributs, les fonctions pour les méthodes.
- La création d'instance : la bénédiction (bless) de la référence de l'objet.
- Programmer un constructeur.
- Agir quand l'objet est détruit, la méthode DESTROY.
- L'héritage : le tableau @ISA et la classe UNIVERSAL.
- La surcharge de méthode, l'utilisation de SUPER.
- Abstraction : l'avenir de l'objet dans Perl, de Class::Std et Class::Accessor à Moose.

### Travaux pratiques

Programmation de classes. Mise en oeuvre des concepts Objets (héritage, surcharge...).

## 3 Interface graphique : de Perl/Tk à GTK

- Comprendre et maintenir du code écrit en Perl/Tk.
- Présentation et installation de Perl/Tk.
- Gestionnaire de géométrie : Pack, Grid, Place.
- Les Widgets par famille.
- Les boutons : Button, Checkbutton, Radiobutton.
- Libellé et zone de saisie : Label, Entry.
- Ascenseur, liste, menu et saisie de texte : Scrollbar, Listbox, Menubutton, Optionmenu, Text.
- Les conteneurs : Canvas, Frame, Toplevel.

### Travaux pratiques

Création d'une application avec interface graphique en Perl/Tk et gestion événementielle.

## 4 Écrire du nouveau code en utilisant GTK

- Lien entre Perl et GTK : installer GTK et ses bibliothèques.
- Widget : hiérarchie, création, accélérateurs clavier, conteneurs de bases...
- Widget : signaux et événements.
- Les types de boutons : Toggle (interrupteurs), radio...

### Travaux pratiques

Manipulation de différents composants : boutons, boîte de boutons, Widget cadre, ...

### MOYENS PÉDAGOGIQUES ET TECHNIQUES

- Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les formations pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
- À l'issue de chaque formation ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
- Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le participant a bien assisté à la totalité de la session.

### MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

### ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Pour toute question ou besoin relatif à l'accessibilité, vous pouvez joindre notre équipe PSH par e-mail à l'adresse [psh-accueil@orsys.fr](mailto:psh-accueil@orsys.fr).

## 5 Evolutions

- Fonctionnalités apportées par Perl 5.10, 5.12, 5.14.
- Perl 6 : un nouveau langage.
- Nouveautés : les sigils, typage statique et dynamique, paramètres formels des fonctions.
- Un véritable langage Objet.
- Les jonctions et l'évaluation paresseuse.