

Formation : Développement web en Python avec Flask

Cours pratique - 4j - 28h00 - Réf. FLK

Prix : 2580 CHF H.T.

NEW

Cette formation vous apprendra à développer des applications web avec Flask, un framework léger et flexible en Python. De l'installation à la mise en production, vous découvrirez la gestion des routes, des formulaires, des bases de données, ainsi que l'authentification et la sécurisation de votre application.

Objectifs pédagogiques

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

- ✓ Comprendre le fonctionnement du framework Flask et son architecture
- ✓ Savoir créer et configurer une application Flask
- ✓ Maîtriser la gestion des routes, des requêtes HTTP et des templates Jinja2
- ✓ Gérer les formulaires, les sessions et les bases de données avec Flask-SQLAlchemy
- ✓ Mettre en place un système d'authentification et sécuriser une application Flask
- ✓ Déployer une application Flask sur un serveur en ligne

Public concerné

La formation est destinée aux développeurs souhaitant acquérir une connaissance approfondie des bases du framework Flask.

Prérequis

Une bonne maîtrise de Python, des bases en HTML/CSS, ainsi qu'une compréhension générale du modèle client-serveur et des requêtes HTTP.

Vérifiez que vous avez les prérequis nécessaires pour profiter pleinement de cette formation en faisant [ce test](#).

PARTICIPANTS

La formation est destinée aux développeurs souhaitant acquérir une connaissance approfondie des bases du framework Flask.

PRÉREQUIS

Une bonne maîtrise de Python, des bases en HTML/CSS, ainsi qu'une compréhension générale du modèle client-serveur et des requêtes HTTP.

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.

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

1 Introduction à Flask

- Présentation du framework Flask et ses cas d'utilisation.
- Comparaison entre Flask et Django.
- Comprendre le modèle client-serveur et l'architecture MVC.
- WSGI et le moteur de templates Jinja2.

Travaux pratiques

Présentations générales.

2 Installation et première application

- Installation et configuration de Flask sur Windows et Linux.
- Création d'une première application Flask ("Hello World").
- Modifier le port et l'adresse IP du serveur Flask.

Travaux pratiques

Installer Flask et créer une première application affichant "Hello World".

3 Gestion des routes et des requêtes

- Définition et gestion des routes avec @app.route.
- Utilisation des règles dynamiques dans les URL.
- Gestion des méthodes HTTP (GET, POST, PUT, DELETE).
- Redirections et gestion des erreurs (404, 500).

Travaux pratiques

Créer différentes routes avec @app.route, utiliser des règles dynamiques et tester les méthodes HTTP.

4 Templates et affichage dynamique

- Introduction aux templates Jinja2.
- Héritage de templates et utilisation des placeholders.
- Inclusion de fichiers CSS et JavaScript dans un projet Flask.
- Affichage dynamique avec les variables et les boucles dans Jinja2.

Travaux pratiques

Utiliser Jinja2 pour afficher dynamiquement des données et organiser les templates avec l'héritage.

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.

5 Gestion des formulaires et des sessions

- Introduction à l'objet Request et gestion des données envoyées par un formulaire.
- Utilisation des cookies et des sessions en Flask.
- Gestion des formulaires avec Flask-WTF.
- Validation des entrées utilisateurs et gestion des erreurs.
- Utilisation de Flask-Flash pour afficher des messages.

Travaux pratiques

Créer un formulaire avec Flask-WTF, valider les entrées et stocker des données en session.

6 Base de données et Object-Relational Mapping (ORM)

- Introduction à SQLAlchemy et Flask-SQLAlchemy.
- Configuration et connexion à SQLite, MySQL et PostgreSQL.
- Définition des modèles et relations (One-to-Many, Many-to-Many).
- Création et gestion des migrations avec Flask-Migrate.
- Exécution de requêtes SQL et ORM (Create, Read, Update, Delete).

Travaux pratiques

Configurer Flask-SQLAlchemy, créer un modèle de base de données et exécuter des requêtes CRUD.

7 Authentification et sécurité

- Mise en place de l'authentification avec Flask-Login.
- Création des modèles et formulaires d'inscription et de connexion.
- Hachage des mots de passe avec Bcrypt et stockage sécurisé.
- Gestion des rôles et permissions avec Flask-Principal.
- Sécurisation des routes avec @login_required.

Travaux pratiques

Mettre en place un système d'inscription et de connexion avec Flask-Login et sécuriser les mots de passe.

8 Déploiement et maintenance

- Déploiement d'une application Flask sur Heroku ou AWS.
- Configuration des variables d'environnement et du fichier Procfile.
- Mise en place des logs et monitoring des erreurs.
- Gestion des erreurs (404, 500) et mise en place des pages d'erreur personnalisées.

Travaux pratiques

Déployer l'application sur Heroku ou AWS et configurer les logs et variables d'environnement.

Dates et lieux

CLASSE À DISTANCE

2026 : 15 sep., 15 déc.