Présentation et Programme de la filière

INGENIEUR DEVELOPPEMENT SPECIALISATION : JAVA

Du Du 31/08/2026 au 30/11/2026

Filière

64 Jours | 448 Heures

V1.0 – Mai 2026

AJC INGÉNIERIE est une S.A.R.L. au capital de 7.622 € R.C.S CRETEIL 414 248 203 00047
N° de déclaration d'activité 11 79 51 886 75 (ce numéro ne vaut pas agrément de l'état)
161 Avenue de Verdun 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

La demande Client – Profils recherchés
Objectifs – Prérequis

Titre du poste

INGENIEUR DEVELOPPEMENT SPECIALISATION : JAVA

Nombre jour/heures

64 jours – 448 heures.

Type de formation

Distanciel.

Nombre de candidats souhaités

23

Paris - Public Visé

Toutes personnes inscrites comme demandeurs d'emploi issus d'une filière scientifique.

Objectif global

- Maitriser le langage Java, les Framework Spring et Angular.
- Être capable en fin de session de concevoir une application Web en architecture monolithique
- Acquérir le savoir être du consultant

Prérequis

- Des notions d'algorithmie seraient un plus

Évaluation - Compréhension - Acquisition

Pour chacun des modules - Pour chaque thème. Un test d'évaluation d'acquisition des compétences sera demandé aux stagiaires. Un questionnaire par thème se compose de 20 questions QCM - Avec une valeur de 5 points par bonne réponse - Une obligation de 75% est demandée pour considérer comme compétences acquises - Si ce résultat n'est pas atteint Le formateur reviendra sur les personnes en "Echec" avec le responsable pédagogique.
La fin de la formation sera consacrée à un projet final reprenant l'ensemble des acquis de la formation. Les apprenants participeront à une soutenance pour présenter leur projet devant un jury et démontrer leurs nouvelles compétences

Lieu

A distance

Coût de la formation par participant - Prise en charge par France Travail / ou OPCO

9500 € TTC

Modalités et délais d'accès

- Les postulants devront passer une série d'entretiens pour intégrer la formation durant les 35 jours précédant l'entrée en formation.
- Ils seront informés de leur inscription au plus tard 15 jours avant le début de la session.

La demande Client – Profils recherchés
Objectifs – Prérequis

Accessibilités aux personnes en situation de handicap

Les personnes en situation de handicap sont invitées à nous communiquer leurs besoins spécifiques. Nous ferons tout pour les mettre dans les meilleures conditions de suivi de la formation possible. (compensation, accessibilité…)

Nous tenons à votre disposition notre politique d'accessibilité en cas de situation de handicap (livret d'accueil disponible sur notre site ou sur demande).

Attestation/certification

Une attestation de fin de stage sera remise à tous les participants à l'issue de leur parcours.

Méthodes mobilisées

Alternance d'exercices, cas pratiques, QCM et de notions théoriques, Projet Fil Rouge.

Les supports de cours seront remis via notre la plate-forme de téléchargement Quest et/ou AJC Classroom. AJC met à la disposition de chacun un accès aux logiciels utiles dans le cadre de leur module.

Informations concernant les classes virtuelles :
Avec @JC CLASSROOM, les stagiaires profitent des mêmes interactions avec leur formateur et l'équipe pédagogique qu'en présentiel : échanges en visioconférence et par chats. La formation se déroule en connexion continue 7h/7.
Le formateur peut vérifier l'avancement du travail et évaluer des stagiaires à l'aide d'exercices et de cas pratiques. Cela lui permet d'apporter un suivi pédagogique et des conseils personnalisés.

Notre équipe technique envoie aux futurs stagiaires les modalités de connexion (accès, identifiants, dates, heures et numéro de la hotline) par mail dès leur inscription.
Si les stagiaires rencontrent un problème de connexion, ils peuvent joindre à tout moment (avant ou même pendant la formation) notre hotline assistance technique au 01 82 83 72 41 ou par mail

Contacts

Référent Pédagogique : Jean Marc Anciaux
+33 1 75 43 82 36
jm.anciaux@ajc-ingenierie.fr

Référent Handicap: Jeanne Aghel
+33 1 75 43 82 36
j.aghel@ajc-ingenierie.fr

Programme de la formation

Titre des modules		Nombre de jours	Nombre d'heures
COMPORTEMENTAL ET METHODES	SAVOIR-ETRE EN ENTREPRISE	1	7
	GESTION DU TEMPS ET DES PRIORITES	1	7
FONDAMENTAUX ET BASES DE DONNEES	ALGORITHMIE	3	21
	UML	1	7
	INIT BDD ET SQL	3	21
	UNIX	1	7
JAVA	JAVA OBJET	4	28
	JAVA AVANCE	5	35
WEB	XML ET JSON	1	7
	HTML, CSS, BOOTSTRAP	2	14
	JAVASCRIPT	3	21
	SERVLET / JSP	3	21
	ANGULAR	6	42
OUTILS ET METHODES	MAVEN ET GIT	1	7
	AGILE SCRUM	2	14
	TDD	1	7
FRAMEWORKS	JPA AVEC HIBERNATE	4	28
	SPRING CORE, DATA ET TEST	3	21
	SPRING MVC	4	28
	SPRING BOOT ET SECURITY	3	21
	QUARKUS	2	14
DEVOPS	DEVOPS : FONDAMENTAUX	1	7
	DOCKER	3	21
	JENKINS	1	7
PROJET	PROJET FINAL	5	35
Total Formation		**64**	**448**

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

COMPORTEMENTAL

AJC INGÉNIERIE est une S.A.R.L. au capital de 7.622 € R.C.S CRETEIL 414 248 203 00047
N° de déclaration d'activité 11 79 51 886 75 (ce numéro ne vaut pas agrément de l'état)

161 Avenue de Verdun 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

SAVOIR-ETRE EN ENTREPRISE

Durée : 1 jour

Objectif général

Adopter la bonne attitude au quotidien, trouver son style pour donner une bonne image de soi
Comprendre l'impact du comportement verbal et non verbal sur ses relations professionnelles
Appréhender les codes du savoir-être en entreprise

Contenu

Pourquoi développer son savoir être professionnel ?
- Pourquoi présenter une bonne image chez un employeur ?
- Quelles sont les conséquences pour soi et pour son quotidien ?
- Retour sur son rapport au savoir-être et au savoir-vivre avec les autres

Adopter le bon comportement professionnel en entreprise
- Comprendre l'impact qu'exerce son image sur soi et sur les autres
- Définir si son image contribue ou non à se mettre en valeur
- Apprendre à mettre en avant ses talents, savoir communiquer dans un milieu professionnel
- Quel vocabulaire adopter ?
- Prendre conscience de l'impact de son comportement non verbal dans la communication
- Accepter ses capacités et ses limites, ses défauts

Intégrer les règles incontournables de la vie en entreprise
- Comprendre le fonctionnement, les règles de l'entreprise et usages de chacun
- Mise au point sur la notion de harcèlement et de respect mutuel
- Utiliser les règles de politesse et les expressions appropriées à chaque circonstance
- Savoir gérer les situations embarrassantes, éviter les impairs, les rattraper s'ils se sont produits
- Présenter et recevoir des excuses
- Tutoyer ou vouvoyer ? Adopter la bonne distance

S'intégrer dans une équipe
- Intégrer et comprendre la culture de son entreprise : ses valeurs et ses codes
- Comprendre vite pour s'adapter rapidement : les personnes, les objectifs, les contraintes
- Développer son écoute active et optimiser sa communication

Intégrer les technologies dans l'appréciation du savoir-vivre
- Adopter les bons usages de la messagerie au travail, les codes et les limites
- Mise au point sur l'utilisation du téléphone et autres outils personnels dans un cadre professionnel
- Focus sur les bonnes pratiques métier : à déterminer en amont de la formation

Évaluation - Compréhension – Acquisition

Pendant la formation, le formateur évalue la progression pédagogique des participants via des QCM, des mises en situation et des travaux pratiques.

Programme détaillé

GESTION DU TEMPS ET DES PRIORITES

Durée : 1 jour

Objectif général

- Acquérir des outils et des méthodes de gestion du temps afin de mettre en place des comportements nouveaux
- Prendre conscience de son comportement
- Reprendre le contrôle de son temps

Contenu

Le temps : un allié de la croissance professionnelle

Connaître les différentes manières de structurer son temps

- Types de personnalités et structuration du temps
- Bilan de ses pratiques actuelles et de l'influence de son environnement
- Prise de conscience individuelle, premier **diagnostic et niveaux de motivation de chacun**

Savoir faire des choix

- Clarifier sa mission et les tâches qui en découlent
- Fixer et fractionner des objectifs
- Hiérarchiser ses priorités
- Savoir filtrer, sélectionner les véritables urgences

Maîtriser son temps sans subir

- Déterminer et agir sur les "voleurs de temps"
- Mieux renoncer pour mieux choisir

Gérer son temps avec les autres

Savoir dire "non"

- Gérer les interruptions
- Savoir déléguer

Utiliser ses forces positives

- Mieux connaître son capital énergie, ses rythmes de travail
- Contacter ses ressources positives, s'en servir comme multiplicateur d'énergie
- Savoir se concentrer, se motiver, s'arrêter, se relaxer

Intégrer le stress

- Rôle du stress, personnalités sensibles
- Se servir du "bon" stress, se protéger du "mauvais" stress
- Gestion des situations de stress les plus fréquentes ou cas particuliers

Qu'acceptez-vous de changer ?

- Déterminer les points réalistes de son contrat de changement
- Visualiser les résultats, modéliser ceux qui savent gérer leur temps

Évaluation - Compréhension – Acquisition

Pendant la formation, le formateur évalue la progression pédagogique des participants via des QCM, des mises en situation et des travaux pratiques.

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

GESTION DU STRESS

Durée : 1jour

Objectif général

Maîtriser les mécanismes du stress dans un écosystème exigeant, turbulent ou en mutation.
Canaliser son stress pour en faire une source de motivation et d'amélioration de sa performance.
Dominer ses émotions afin de rester centré sur les priorités réelles.
Développer des attitudes positives pour préserver son équilibre en situation relationnelle ou managériale.

Contenu

Compétences développées :
Connaissance d'actions à mettre en place pour prévenir les risques psycho-sociaux.
Mieux gérer une équipe : animer, motiver et accompagner.
Savoir communiquer avec ses collègues, son équipe et ses clients, en situation d'urgence ou de tension

1.Identifier le stress : ses origines et ses manifestations
2.Distinguer les amplificateurs de stress, notamment dans le monde du travail et notre société actuelle
3.Découvrir différents modèles de stress
4.En reconnaître les symptômes : physiologiques, émotionnels, comportementaux et sociaux
5.Evaluer son stress
6.Concevoir des plans d'actions pour y remédier

À la fin de cette formation

À la fin de la séance, les stagiaires seront capables d'identifier les mécanismes du stress pour mieux l'anticiper, en réduire les effets et améliorer ses performances

Méthodes pédagogiques

Ce programme s'appuie sur une approche pédagogique vivante et interactive grâce à des jeux de rôles, des exercices ludiques et un savant mélange de travaux individuels et en groupes.
Support de cours – Exercices pratiques – Mises en situation.

Évaluation - Compréhension - Acquisition

À la fin de ce module les participants devront répondre à 1 questionnaire QCM de 20 questions chacun.
Un minimum de 75% est attendu pour validation des acquis.

Références Tests

7.0 Gestion du stress

Programme détaillé

FONDAMENTAUX ET BASES DE DONNEES

AJC INGÉNIERIE est une S.A.R.L. au capital de 7.622 € R.C.S CRETEIL 414 248 203 00047
N° de déclaration d'activité 11 79 51 886 75 (ce numéro ne vaut pas agrément de l'état)

161 Avenue de Verdun 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

ALGORITHMIE

Durée : 3 jours

Objectif général

- Présenter les principes fondamentaux de la programmation et de l'algorithmique et expliquer les notions communes à tous les langages de programmation
- S'approprier les structures logiques et la démarche de résolution d'un problème de façon structurée et indépendante de toute contrainte matérielle ou logicielle
- Résoudre des problèmes plus ou moins complexes

Contenu

- Maîtriser les outils de l'algorithmique (Schémas de programme, types et structures de données, modules)
- Acquérir les bases des méthodes de programmation structurée nécessaires à l'apprentissage de tout langage de programmation
- Apprendre à raisonner sur un algorithme
- Maitriser les notions de fichiers
- Découvrir et mettre en œuvre la traduction d'un algorithme dans un langage de programmation

Évaluation - Compréhension – Acquisition

Pendant la formation, le formateur évalue la progression pédagogique des participants via des QCM, des mises en situation et des travaux pratiques.

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

ALGORITHMIE (1/2)

Durée : 3 jours

Objectif général

- Structurer des programmes selon un algorithme
- Maîtriser les éléments de lexique et de syntaxe d'un langage pour écrire un programme
- Compiler et exécuter un programme

Contenu

Les fondements de la programmation

- Qu'est-ce qu'un programme ? Qu'est-ce qu'un langage ? Les différents paradigmes. Quel langage pour quelle application ?
- Les compilateurs. Les exécutables.
- Les responsabilités d'un programmeur.
- Qu'est-ce qu'un algorithme ?
- Les besoins auxquels répond un algorithme.
- Le concept de pseudo-langage

Genèse d'un premier programme

- Ecriture d'un programme simple : syntaxe et instructions.
- Compilation et exécution du programme.
- Qu'est-ce qu'une librairie ? Son rôle, son usage

Règles de programmation

- Convention de nommage.
- Convention syntaxique.
- Utilisation des commentaires. Pourquoi commenter les développements ?
- Améliorer la lisibilité des programmes : indentation du code, découpage du code...

Les variables

- Qu'est-ce qu'une variable ?
- Pourquoi typer une variable ?
- Les types primitifs : entiers, chaînes de caractères, nombres réels, autres.
- Déclaration, définition et initialisation d'une variable.
- Les constantes.
- Saisie, affichage, affectation, conversion de type.
- Organiser ses données sous forme de tableaux.
- Les types évolués : enregistrement, matrice, arbre

Programme détaillé

ALGORITHMIE (2/2)

Durée : 3 jours

Contenu

Opérateurs et expressions

- Qu'est-ce qu'un programme ? Qu'est-ce qu'un langage ? Les différents paradigmes. Quel langage pour quelle application ?
- Les compilateurs. Les exécutables.
- Les responsabilités d'un programmeur.
- Qu'est-ce qu'un algorithme ?
- Les besoins auxquels répond un algorithme.
- Le concept de pseudo-langage

Les structures de contrôle

- Ecriture d'un programme simple : syntaxe et instructions.
- Compilation et exécution du programme.
- Qu'est-ce qu'une librairie ? Son rôle, son usage

Les procédures et les fonctions

- Convention de nommage.
- Convention syntaxique.
- Utilisation des commentaires. Pourquoi commenter les développements ?
- Améliorer la lisibilité des programmes : indentation du code, découpage du code

Introduction à la programmation objet

- Les concepts associés à la programmation objet : classe, attribut, méthode, argument.
- Introduction aux bonnes pratiques d'organisation de conception et d'organisation d'un programme

Maintenance, débogage et test des programmes

- Savoir lire et interpréter les différents messages d'erreurs.
- Utiliser un débogueur : exécuter un programme pas à pas, points d'arrêts, inspecter les variables pendant l'exécution

Évaluation - Compréhension – Acquisition

Pendant la formation, le formateur évalue la progression pédagogique des participants via des QCM, des mises en situation et des travaux pratiques.

Programme détaillé

UML

Durée : 1 jour

Objectif général

- Appréhender les différentes phases de la modélisation objet en UML.
- Comprendre la représentation et l'intérêt d'utilisation des différents diagrammes UML.
- Savoir traduire un besoin fonctionnel en s'appuyant sur les diagrammes UML.

Contenu

Les fondements de la programmation

- Qu'est-ce qu'un programme ? Qu'est-ce qu'un langage ? Les différents paradigmes. Quel langage pour quelle application ?

Les principales notions des approches objets et de l'UML

- Le modèle classe-instance et l'acquisition du vocabulaire de base de l'UML
- L'encapsulation des structures de données et des traitements dans les objets
- Les objets instances de classes, les classes spécifiant les objets
- Les méthodes, la collaboration et les envois de messages entre objets
- Les sous-classes et l'héritage
- Les interfaces

Analyse et conception par objets

- Le rôle des différentes activités et pour lesquelles utiliser l'UML
- Les grandes phases et les activités de développement.
- Les activités d'analyse et de la conception.
- Les bénéfices attendus d'une analyse et d'une conception objets.
- Les différents types de modèles : d'usage, statiques, dynamiques, d'architecture, de réalisation
- Les mécanismes d'extension et d'adaptation de l'UML : stéréotypes, contraintes et profils.

Les modèles statiques d'UML

- Les significations et notations des différents modèles de l'UML
- Diagramme de classes, les classes, les attributs.
- Liens et associations, cardinalités, rôles.
- Les contraintes.
- Relations de généralisation entre classes et héritage
- Agrégation et composition.
- Les opérations et leurs spécifications.

Use case

- (Re) formulation des besoins
- Les acteurs, les cas d'utilisation
- Les relations entre cas d'utilisation
- Les diagrammes de cas d'utilisation
- Cas d'utilisation et scénarios principaux

Programme détaillé

INIT BDD ET SQL

Durée : 3 jours

Objectif général

- Comprendre les principaux concepts des SGDBR (Système de Gestion des Bases de Données Relationnelles) et d'algèbre relationnelle utilisés dans le langage SQL
- Prendre en main un environnement SQL
- Appréhender l'écriture des requêtes SQL pour extraire des données et mettre à jour la base
- Manipuler les données dans une base avec SQL
- Savoir extraire les informations de plusieurs tables
- Assimiler les fonctions standards du langage SQL
- Être en mesure d'écrire des requêtes compatibles avec plusieurs SGBD

Contenu

Introduction
- Rappel sur le modèle relationnel
- Les normes et caractéristiques du langage SQL

Le langage d'interrogation des données (LID)
- La sélection de données
- Les restrictions ou conditions
- Les tris
- Les jointures entre tables

Utilisation des fonctions
- Fonctions arithmétiques
- Fonctions de chaînes de caractères
- Fonctions de regroupement (ou d'agrégation)
- Les clauses GROUP BY et HAVING

Utilisation des opérateurs ensemblistes
- Opérateur UNION
- Opérateur INTERSECT
- Opérateur EXCEPT ou MINUS

Utilisation de sous-interrogations
- Dans le where
- Dans la clause from
- La fonction OVER
- Sous requête synchronisée

Le langage de manipulation de données (LMD)
- L'insertion de données (insert)
- La mise à jour (update)
- La suppression d'informations (delete)

Notions sur le langage de définition de données (LDD)
- Création de tables : syntaxe
- Les types de données
- Les types de contraintes
- Modification de la définition d'une table
- Suppression d'une table
- Notions sur les vues, les séquences, les index et les synonymes

Programme détaillé

UNIX (1/2)

Durée : 1 jour

Objectif général

- Acquérir la connaissance des commandes fondamentales des systèmes d'exploitation Unix et Linux à travers des exercices modulaires de difficulté progressive
- Devenir autonome pour une première prise en main d'un système
- Passer l'étape importante de la maîtrise de l'éditeur "vi".

Contenu

Introduction
- Rappel sur le modèle relationnel
- Les normes et caractéristiques du langage SQL

Présentation d'UNIX L'historique
- Les fonctionnalités d'UNIX
- L'organisation du système d'exploitation

Connexion
- La connexion en mode texte (telnet, ssh)
- La connexion en mode graphique (XDMCP) La déconnexion

Environnement
- Les notions de UID et GID
- L'arborescence type sous UNIX Les
- principaux répertoires
- Le répertoire de connexion
- Le prompt

Commandes
- La forme générale d'une commande
- La recherche et l'exécution d'une
- commande
- Les premières commandes (id, hostname, pwd, ls, man...

Utilisation du shell
- Les différents shells
- Les caractères spéciaux
- L'interprétation de la ligne de
- commandes
- Les variables d'environnement
- Les variables utilisateur
- Les redirections d'entrées / sorties
- Le pipe
- Les calculs arithmétiques (expr, bc,
- let...)

Arborescence et répertoires
- L'arborescence type sous UNIX Les
- principaux répertoires
- La structure de l'arborescence UNIX
- Le déplacement dans l'arborescence
- (pwd, cd) La manipulation de
- répertoires (mkdir, rmdir)

Programme détaillé

UNIX (2/2)

Durée : 1 jour

Contenu

Fichiers
- Les différents types de fichiers
- La substitution de caractères
- La manipulation de fichiers (cat, more, pg, cp, mv, rm)

Droits
- Les principes
- Les droits par défaut
- La manipulation de droits (chmod)

Bases de l'éditeur de texte vi
- Les différents modes de vi
- L'utilisation de vi
- Le mode vi sur la ligne de commandes

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

JAVA

AJC INGÉNIERIE est une S.A.R.L. au capital de 7.622 € R.C.S CRETEIL 414 248 203 00047
N° de déclaration d'activité 11 79 51 886 75 (ce numéro ne vaut pas agrément de l'état)

161 Avenue de Verdun 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

JAVA OBJET (1 / 3)

Durée : 4 jours

Objectif général

- Acquérir les compétences pour écrire des scripts en Shell et exploiter les possibilités des filtres Unix/Linux.

Contenu

Présentation générale

- Principes fondateurs de l'objet : abstraction/encapsulation. Héritage, mise en œuvre.
- Présentation générale : le langage, les outils, la bibliothèque.
- Distributions de Java.

Aspects syntaxiques, types et expressions

- Structuration syntaxique d'une application Java.
- Exemple de syntaxe sur une application simplifiée.
- Vue externe d'une classe : syntaxe d'utilisation.
- Vue interne d'une classe : syntaxe
- d'implémentation.
- Notion de type. Utilisation comparée des types de base et des types Objet.
- Utilisation simple des types de base : les nombres entiers, les flottants, les types Char et Boolean.
- Notion d'expression.
- Exemples de déclarations : variables et constantes.
- Désignation comparée des types de base et des types Objet.
- Utilisation des opérateurs avec les objets.
- Cas des champs static ou variables de classes.
- Complément sur les types :
- utilisation de base des tableaux.
- Conversion types de base/type objet.
- Conventions d'écriture.

Méthodes et instructions

- Syntaxe d'invocation des méthodes.
- Méthodes de classes et méthodes d'instances.
- Définition et utilisation des méthodes.
- La surcharge des méthodes.
- Notion de sous-bloc.
- Catégories d'instructions.
- Principales instructions de contrôle : if, while, for, return, break.

Programme détaillé

JAVA OBJET (2 / 3)

Durée : 4 jours

Contenu

Utilisation de l'abstraction

- Exemple simple d'utilisation d'un objet : déclaration, instanciation ou fabrication, délégation.
- Utilisation des constructeurs d'objets : découverte de la documentation en ligne.
- Utilisation de l'interface programmatique des objets : exemple de la classe Date.
- Une classe très utilisée : la classe String.
- Particularités liées aux chaînes de caractères.
- Utilisation de la classe StringBuffer : exemple d'utilisation de la surcharge de méthodes.
- Utilisation de l'héritage
- Rappel du principe d'héritage et terminologie.

Utilisation de l'héritage.

- Exemple de graphe d'héritage.
- La classe Object et la généricité.
- Utilisation du polymorphisme.
- Spécialisation d'une référence polymorphe.
- Typage des références/typage des objets.
- Comportement des méthodes et typage.
- Généricité des classes conteneurs : exemple de la classe Vector.
- Les ajouts de JAVA 5 (TIGER) : les generics.

Utilisation du mécanisme d'interface

- Interface implicite et explicite d'une classe.
- Syntaxe associée aux interfaces explicites.
- Cas d'utilisation des références d'interfaces : flexibilité, limitation de la portée, polymorphisme.
- Exemple d'implémentation multiple d'interfaces.
- Synthèse sur l'intérêt des interfaces pour les méthodes.
- Utilisation des interfaces pour les constantes.
- Exemples avancés d'utilisation d'interfaces.

Programme détaillé

JAVA OBJET (3 / 3)

Durée : 4 jours

Contenu

Développement de classes

- Approche méthodologique, analyse statique, dynamique, métier.
- Notation UML : diagramme de classe, d'état/transition, de séquence.
- Squelette d'une classe : constituants de base, outils de génération automatique.
- Compléments sur les droits d'accès.
- Organisation en packages.
- Contraintes liées aux packages.
- Ecriture des constructeurs.
- Constructeur par défaut.
- Compléments sur l'écriture des constructeurs.
- L'auto-référence "this".
- Champs et méthodes statiques.
- La méthode Main.

Développement d'interfaces

- Rappels et compléments sur les principes.
- Syntaxe associée aux interfaces, cas des constantes.
- Définition d'interfaces pour les méthodes.
- Implémentation et extensions multiples d'interfaces.
- Implémentation partielle d'interface.
- Exemples sur l'utilisation d'interfaces.

Développement de classes dérivées

- Rappels des principes.
- Approche méthodologique pour le découpage en classes.
- Méthodes et classes abstraites.
- Classes abstraites et interfaces.
- Droit d'accès aux champs et héritage.
- Enchaînement des constructeurs et héritage.
- Redéfinition et surcharge.

Programme détaillé

JAVA AVANCE

Durée : 5 jours

Objectif général

- Devenir autonome sur Linux afin de garantir la bonne disponibilité des serveurs
- Pouvoir prendre en charge la responsabilité de l'administration de systèmes Linux
- Savoir intégrer Linux avec les autres systèmes d'exploitation de l'entreprise
- Être en mesure de garantir un premier niveau de sécurité d'une infrastructure Linux

Contenu

Présentation

- L'historique d'Unix et Linux
- Les caractéristiques de Linux, les Unix-Like, les distributions Linux
- Comment administrer le système : le mode texte et les outils d'administration
- La documentation : le man, les autres sources d'informations (Howto, ...)

Installer Linux et ses applications

- Introduction : Linux, les distributions Linux, les sources d'information
- Installer un système de type RedHat et un système de type Debian
- Administrer le système avec sudo sous Debian et RedHat
- Installer des applications sous RedHat : les paquets RPM, le système YUM
- Installer des applications sous Debian : les paquets DEB, le système APT

Administrer le système avec les commandes du mode texte

- Utiliser le Shell, connaître les commandes de base du système (rappels)
- Savoir lire des scripts Shell
- Gérer les utilisateurs : les commandes de gestion des comptes, les droits (rappels)
- Gérer les processus (rappels), gérer les bibliothèques partagées
- Savoir programmer des travaux périodiques
- Savoir organiser les journaux de bords et leur rotation

Gérer l'espace disque

- Comprendre la vision Linux des disques, partitionner des disques (Msdos, GPT)
- Gérer le LVM, gérer le swap
- Gérer les FS (ext2/ext3/ext4, xfs, ...) et les quotas

Gérer l'arrêt et le redémarrage

- Connaître les grandes étapes du démarrage (BIOS, bootloader, kernel, initramfs, init)
- Utiliser le chargeur ("bootloader") Grub
- Gérer le démarrage des services : init SysV, Upstart, systemd ; la notion de runlevel

Configurer TCP/IP en environnement Linux

- Ajouter un système (Debian, RedHat) dans un réseau IPv4/IPv6
- Connaître les commandes de diagnostics
- Comprendre le fonctionnement des systèmes INETD (inetd, xinetd), les wrappers

Programme détaillé

WEB

AJC INGÉNIERIE est une S.A.R.L. au capital de 7.622 € R.C.S CRETEIL 414 248 203 00047
N° de déclaration d'activité 11 79 51 886 75 (ce numéro ne vaut pas agrément de l'état)

161 Avenue de Verdun 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

XML ET JSON

Durée : 1 jour

Objectif général

- Lire et comprendre des documents XML et JSON
- Modéliser et définir des données en XML et JSON

Contenu

Introduction à XML et JSON

- Le modèle de données XML : éléments et attributs, document bien formé et valide.
- Représentation sérialisée ou arborescente, le modèle logique XML Infoset, le parsing de XML.
- La galaxie XML : standards techniques et standards métiers.
- XML et bureautique : les standards Open Document d'Open Office et OpenXML de Microsoft. EXI : l'XML compressé.
- Le modèle de données JSON : objet, tableau et valeurs littérales.
- Intégration avec les langages de programmation (JavaScript, PHP…). Les frameworks utilisant JSON (jQuery, Angular…).
- Le parsing de JSON. Différences avec XML.
- Les outils de développement XML et JSON.

Définition de données XML avec DTD et XML Schema

- Document Type Definition (DTD) et typage des documents.
- Définition d'éléments, d'attributs, d'entités ; éléments simples et composés, entités paramètres.
- XMLschema : types simples et types complexes, déclaration des éléments et des attributs.
- XMLschema : les constructeurs de collections, héritage de types, réutilisation de définitions.
- Les espaces de noms xmlns : intérêt pour l'intégration de données XML.
- Les bonnes pratiques : règles d'écriture DTD ou schémas XML, la gestion de versions.
- Les principaux outils de développement de DTD et schémas XML.

Définition de données JSON

- Schéma JSON : concepts de base, mots-clés de validation, mots-clés hyper-médias.
- Les méta-schémas pour définir les schémas JSON et les formats Hyper-Schema.
- Les schémas standards : ex. coordonnées géographiques, card, calendrier, adresse…
- Bibliothèques de validation de schémas JSON.

Programme détaillé

HTML, CSS ET BOOTSTRAP

Durée : 2 jours

Objectif général

- S'initier aux technologies standards du Web
- Comprendre le positionnement de ces technologies dans une architecture en couche
- Augmenter la productivité de création d'écrans avec Bootstrap

Contenu de la formation vente au téléphone

Introduction protocole HTTP
- Requêtes et Réponse HTTP
- En tête HTTP
- Codes retour serveur
- Analyse avec F12

Introduction langage HTML
- Contexte : web statique
- Balises HTML
- HTML et HTML 5
- Formulaire
- Audio et Vidéo
- Validation de champs

Introduction CSS
- Contexte : ergonomie et habillage web statique
- Feuille de style externe, interne et inline
- Notion de cascade
- Notion de class
- Notion de id
- Notion de block
- Sizing et Positionning

Introduction BOOTSTRAP
- Notion de framework
- Augmenter la productivité et l'ergonomie des écrans web
- CSS et Javascript BOOTSTRAP
- Installation et mise en œuvre

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

JAVASCRIPT

Durée : 3 jours

Objectif général

- Comprendre et maîtriser le langage JavaScript
- Développer avec le langage JavaScript

Contenu

Présentation du langage
- Historique et évolution
- Comment et quand utiliser javascript ?
- Comment organiser son code ?
- Environnements et outils de développement

Présentation technique
- Les variables, les types
- Les fonctions
- Les objets
- Première utilisation

Utilisation du DOM
- Présentation du Document Object Model (DOM)
- Fonctions de sélection
- Fonctions de création d'objet DOM
- Modifier les éléments du DOM

Gestion des évènements
- Présentation des évènements courants
- Lier un évènement à un objet du DOM
- Interagir avec les éléments du DOM

AJAX : Asynchronous JavaScript And XML
- Présentation et exemple d'utilisation

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

SERVLET / JSP

Durée : 2 jours

Objectif général

- Ecrire une application web dynamique
- Découvrir l'API des Servlets
- Comprendre l'architecture d'une application web Java
- Comprendre l'architecture d'un server d'application web Java
- Utiliser correctement les différentes techniques Servlet et JSP

Contenu

Architecture J2EE

Introduction à Apache Tomcat

Introduction

- J2EE : une spécification des implémentations, domaine d'application, l'aspect distribué et transactionnel
- Les finalités et les apports de J2EE, évolutivité des applications, portabilité, montée en charge, sûreté de fonctionnement, indépendance vis-à-vis des éditeurs, …
- L'approche composant à toutes les étapes de production et d'exploitation des applications
- L'architecture n-tiers, description des différents tiers et des composants associés
- La notion de conteneurs, leurs rôles, leurs services
- Types de containers (Servlet, EJB, etc.), panorama de l'offre
- Les différents rôles dans le développement d'une application J2EE: Editeur de plate-forme,
- Développeurs de composants, assembleur, Déploiement et exploitation
- Définition des technologies et APIs disponibles

Les applications Web

- Classification des applications : orientées présentation ou service, Modèle requête/réponse, rappels sur le protocole HTTP, cycle de vie d'une application web.
- Définition d'un module web, packaging, déploiement, mise à jour
- Configuration d'une application : mapping des URLs, paramètres d'initialisation, mapping des erreurs, déclaration des ressources

Les servlets

- Définition d'une servlet, technologie au cœur de J2EE
- Cycle de vie d'une servlet, gestion des évènements, des erreurs
- Partage d'information et notion de périmètre (requête, session, etc.)
- Implémenter les services du servlet, récupération de paramètre, construction de réponse
- Les filtres de requête ou de réponses
- Gestion de session utilisateur
- Définition et exemple d'une page JSP
- Cycle de vie d'une page JSP
- Éléments de syntaxe, notion de scriptlet
- Utilisation de librairies de balises

Programme détaillé

ANGULAR (1 / 3)

Durée : 5 jours

Objectif général

- Maîtriser les fondamentaux du Framework Angular et ses nouveautés
- Organiser, modulariser et tester ses développements JavaScript
- Savoir développer plus rapidement et tester des applications web Angular avec JavaScript et TypeScript
- Connaître les bonnes pratiques de développement et de mise en production

Contenu

Présentation générale

- Principes fondateurs de l'objet : abstraction/encapsulation. Héritage, mise en œuvre.
- Présentation générale : le langage, les outils, la bibliothèque.
- Distributions de Java.

Introduction

- Outils et IDE
- Packaging, npm
- Webpack
- Installation node.js / npm
- Installation angular-cli

TypeScript et ES6

- Installation TypeScript
- Transpiler EcmaScript
- Let, variables locales et constantes
- Typage et types natifs
- Paramètres optionnels, valeurs par défaut
- Classes et interfaces
- Gestion des modules
- Arrow functions

Le framework Angular

- Contexte
- Présentation générale
- Description de l'architecture du framework (MVC, workspaces, configurations, ...)
- Présentation des composantes du framework (modules, composants, directives, services, ...)
- La philosophie « composant »
- Commandes du angular-cli
- L'extension du navigateur Angular DevTools
- Les types de NgModules
- Métadonnées du NgModule
- Contexte de compilation des modules
- Import/Export de modules

Programme détaillé

ANGULAR (2 / 3)

Durée : 5 jours

Contenu

Les composants et template
- Métadonnées du composant (selecteur, template, styles, ...)
- Déclarations dans un NgModule
- ViewEncapsulation
- Interpolation / expression
- Property-Binding / Attribute-binding / Class-binding / Style-binding
- Variables locales

Directives
- Directives d'attributs (selecteur, ElementRef, HostListener, @Input)
- Directives de structure (ngIf, ngFor, Symbole *)
- ng-template

Evénements et cycle de vie
- Syntaxe de l'event-binding
- Types d'événement et handlers
- Objet $event
- Property-binding et @Input
- EventEmitter et @Output
- Two way data binding (opérateur banana in a box)
- Cycle de vie d'un composant (ngOnInit, ngOnChanges, ngOnDestroy)

Pipes
- Définition
- Les pipes standards (lowercase, uppercase, currency, decimal, ...)
- Le pipe key-value
- Créer son propre pipe

Services
- Définition
- Création de service
- Injection d'un service
- Providers (+ providedIn) et injectors
- Différence entre injecteur root et injecteur composant

Formulaires
- Control et ControlGroup
- Validations
- Gestions d'erreurs
- Gestion des modifications
- Groupes de champs avec FormBuilder

Programme détaillé

ANGULAR (3 / 3)

Durée : 5 jours

Contenu

Observables, RxJS et HttpClient

- Observables
- RxJS
- HTTP providers
- Requêtes
- Transformation des données et observables
- Options de requêtes

Routing

- Concepts de routage
- Router providers et config
- Router directives
- Méthodes de routage et paramètres

Programme détaillé

OUTILS ET METHODES

AJC INGÉNIERIE est une S.A.R.L. au capital de 7.622 € R.C.S CRETEIL 414 248 203 00047
N° de déclaration d'activité 11 79 51 886 75 (ce numéro ne vaut pas agrément de l'état)

161 Avenue de Verdun 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

MAVEN ET GIT (1 / 2)

Durée : 1jour

Objectif général

- Structurer un projet autour de Maven
- Gérer les dépendances et les repositories
- Comprendre les concepts de base de la gestion des versions et des apports de la décentralisation
- Installer et configurer l'outil GIT sous Windows
- Créer et initialiser un dépôt avec GIT
- Manipuler les commandes de GIT pour gérer les fichiers et les branches

Contenu

Présentation

- Au-delà d'un simple outil de build. Le monde Maven : gestionnaire de sources, tests automatisés, documentation...
- Mise en place d'un premier projet Maven
- Installation de Maven. Le POM (Project Object Model).
- Repository local et repository distant.
- Qu'est-ce qu'un plug-in Maven ?
- Qu'est-ce qu'un goal ?
- Structure standard d'un projet Maven. Contrôle du cycle de vie : installation, compilation, déploiement...
- Notions d'archétype, groupe, artefact, version, assemblies.
- Découpage d'un projet en modules.
- Héritage entre fichiers POM ; le super-POM.
- Exercice
- Installation de Maven et création d'un premier projet Maven.

Les dépendances

- Notion de dépendance et de dépendance transitive.
- Déclaration des dépendances dans le POM.
- Comment résoudre un conflit de dépendances ?
- Exercice
- Paramétrage de dépendances sim

Les repositories

- Limites du repository par défaut.
- Déclaration de plusieurs repositories.
- Gestion de priorités.
- Outils de gestion de repository (Nexus, Artifactory...).
- Exercice
- Migration d'un projet non structuré vers Maven. Paramétrage de dépendances et de repositories.

Programme détaillé

MAVEN ET GIT (2/ 2)

Durée : 1 jour

Contenu

Le travail en équipe

- Problématique générale du travail collaboratif.
- Différentes approches adoptées par les logiciels de gestion des versions.
- Quid sur les bonnes pratiques d'utilisation de ces logiciels.

Présentation de GIT

- Concepts de base du contrôle de version.
- La gestion centralisée ou distribuée.
- Les différentes solutions de gestion de versions : (GIT, CVS, SVN, Mercurial, Bazaar...).
- Apports la décentralisation. Principe de fonctionnement.

Utilisation de GIT, les fondamentaux

- Le modèle objet GIT : blob, tree, commit et tag.
- Le répertoire GIT et le répertoire de travail.
- L'index ou staging area.
- Création et initialisation un dépôt.
- Les concepts de branche, tag et de dépôt.

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

AGILE SCRUM

Durée : 2 jours

Objectif général

- Découvrir la gestion de projets agiles avec la méthode Scrum

Contenu

Adhérer à l'agilité

- Gestion de projet : prédictive vs agile
- Pourquoi l'agilité ?
- Démarche d'adoption de l'agilité
- Le manifeste agile
- Panorama des méthodes agiles

Apprendre les pratiques agiles

- Le Lean Management : objectif, principes
- Kanban : principe, avantage, cycle de vie d'une étiquette
- Pratiques XP (eXtreme Programming)

Maîtriser la méthode agile Scrum

- Le cadre Scrum / Guide
- Cycle de vie d'un projet Scrum
- Rôles définis par Scrum : Product Owner, Scrum Master, Team
- Time boxes : Sprint planning, Sprint Review, Sprint Retrospective, Daily Scrum
- Artéfacts : Product Backlog, Sprint Backlog, Burndown chart
- Règles et principes clé de Scrum
- Responsabilités / rôle
- Choix de la taille du sprint
- Constitution de l'équipe
- Faiblesses de la méthode

Programme détaillé

TDD

Durée : 1 jour

Objectif général

- Maîtriser la démarche et la mise en œuvre du Test Driven Development
- Intégrer les tests dans le cycle de développement d'une application Java
- Prendre en main les principaux outils de tests et d'intégration continue

Contenu

Définition et principes du TDD

- Le test dans le processus de développement. Processus, qualité, tests. Typologie des tests.
- Origine du TDD. L'agilité et les tests.
- Cycle de développement. Les 3A.
- Gestion des exceptions.
- Refactoring et conception émergente.
- Gestion des scénarios. Gains du TDD ?

Tests automatisés avec le framework JUnit

- Le besoin d'un framework de test. JUnit.
- Alternatives (TestNG) et outillage complémentaire.
- Bonnes pratiques associées à JUnit.

Les bonnes pratiques de développement Agiles

- TDD et gestion des données SGBDR, des interfaces graphiques, des interfaces Web.
- Travaux pratiques
- Mise en œuvre de pratiques.

Les objets Mock et Stub

- La théorie.
- Application de la théorie sans utiliser de bibliothèque.
- Découverte des bibliothèques du marché.
- Etude en détail de Mockito.

Techniques d'écriture de tests

- Fixtures. Qualités d'un code de test.
- Tests basés sur la responsabilité, l'implémentation.
- Styles de TDD.

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

FRAMEWORKS

AJC INGÉNIERIE est une S.A.R.L. au capital de 7.622 € R.C.S CRETEIL 414 248 203 00047
N° de déclaration d'activité 11 79 51 886 75 (ce numéro ne vaut pas agrément de l'état)

161 Avenue de Verdun 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

JPA AVEC HIBERNATE (1 / 2)

Durée : 4 jours

Objectif général

- Etablir un mapping entre des objets java et des tables relationnelles
- Créer, mettre à jour et supprimer des objets persistants
- Maîtriser le langage de requêtes JPQL
- Gérer des transactions

Contenu

Techniques de persistance Java et JPA

- Les différents mécanismes de persistance : API Java et Frameworks.
- La solution Java Persistance API (JPA).

Développer une classe persistante

- Coder la classe persistante.
- Effectuer le mapping Objet/relationnel.
- Configurer et démarrer le moteur JPA.
- Effectuer une requête JPQL.
- Sauvegarder un objet persistant.

Mapping Objet/relationnel avec JPA

- Contexte et objectifs d'un ORM.
- Principe de développement des classes persistantes.
- Utilisation des annotations pour configurer un mapping Objet/Relationnel.
- Mapping des classes et des associations.
- Stratégie de mapping pour l'héritage.

Manipuler les objets persistants

- Les différentes techniques de lecture.
- Les stratégies de chargement.
- Principe du lazy loading.
- Les opérations CRUD (Create/Read/Update/Delete).
- Cycle de vie des objets persistants.
- Synchronisation avec la base de données.

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

JPA AVEC HIBERNATE (2 / 2)

Durée : 4 jours

Contenu

Utilisation avancée du mapping

- Clé primaire composée, mapping multitables.
- Contrôler les requêtes INSERT et UPDATE.
- Associations de type list, map et many-to-many.

Le langage JPQL

- Les requêtes d'interrogation.
- Opérations sur les chaînes de caractères et les données temporelles.
- Jointures internes, externes et rapportées.
- Principe des sous-requêtes.
- Requêtes sur les ensembles.

Transactions et accès concurrents

- Rappel des propriétés d'une transaction.
- La gestion transactionnelle avec JPA.
- Intégration dans une application Web et EJB.
- Verrouillage pessimiste et optimiste.

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

SPRING CORE, DATA ET TEST

Durée : 3 jours

Objectif général

- Connaître les bases du Framework Spring
- Savoir gérer la configuration des composants d'une application avec Spring
- Connaître les bonnes pratiques de développement avec Spring
- Connaître les apports de la Programmation Orientée Aspect (AOP)

Contenu

Introduction à Spring

- Concepts de conteneur léger
- Vue d'ensemble et exemples d'utilisation
- Pattern "Inversion de Contrôle (IoC) ; Injection de dépendance"
- Tests unitaires en isolation
- Approche MVC avec Spring MVC

Mise en œuvre de Spring

- Les Beans, BeanFactory et ApplicationContext
- Modes singleton ou prototype
- Gestion des propriétés, "collaborators"
- Méthodes d'injection de dépendance
- Configuration de Beans spécifique à Spring, cycle de vie
- Définition de Bean abstrait et héritage

Spring et l'accès aux données

- Pattern DAO avec JDBC et les Classes abstraites de Spring
- Implémentation DAO avec les APIs Hibernate
- Démarcation de transactions par programmation et déclaration

Gestion des transactions

- Concept de transaction
- Gérer les transactions avec Spring
- Transactions programmatiques
- Transactions déclaratives

Spring Data

- La notion de "Repository".
- Le requêtage (Query method, l'annotation "Query"...).
- Les points d'extensions (intégration à la couche Web).
- Spring Data JPA : requêtage JPA et Query DSL, transaction, configuration.
- Spring Data MongoDB : requêtage MongoDB et Query DSL, utilisation du template, configuration.
- Spring Test
- Spring et le Test Driven Development
- Les annotations de Test

Spring Test

- Spring et le Test Driven Development
- Les annotations de Test

Programme détaillé

SPRING MVC

Durée : 4 jours

Objectif général

- Développer des applications Web avec Spring et Spring MVC.

Contenu

Les bases de Spring Web MVC

- Pattern modèle-vue-contrôleur dans Spring MVC
- La DispatcherServlet
- Présentation du modèle de programmation des contrôleurs
- Les vues dans Spring MVC
- Simplication de la configuration

Options de configuration de Spring MVC

- Beans d'infrastructure dans Spring MVC
- Mapping d'URL
- Intercepteurs et adaptateurs
- Résolution des exceptions
- Source de messages

Gestion de la présentation dans Spring MVC

- Structuration des pages avec JSP et Thymeleaf

Utilisation des vues dans Spring MVC

- Vues et résolution
- Chaîne de résolution des vues
- Alterner les vues
- Vues JSON

Formulaires avec Spring MVC

- Rendu des formulaires
- Conversion des données
- Data binding
- Validation avec Spring et Bean Validation (JSR 303)
- Gestion des objets de formulaire

Personnalisation de l'apparence avec Spring MVC

- Support de l'internationalisation
- Changement du look-and-feel avec les thèmes

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

SPRING BOOT, REST ET SECURITY (1 / 2)

Durée : 3 jours

Objectif général

- Mettre en œuvre le module Spring boot
- Développer des applications riches avec Spring
- Maîtriser la configuration et la sécurité

Contenu

Introduction
- Le module Spring Boot
- Les requis

Les principales fonctionnalités
- Le support de différents types d'application
- Convention over configuration
- L'autoconfiguration
- La gestion simplifiée des dépendances avec les starters
- Le support de Maven et Graddle

La création d'une application
- La création d'un projet dans STS
- La création avec Spring Initializr
- La création d'un projet avec Maven

Une application Spring Boot
- Une application standalone
- La classe SpringApplication
- La configuration d'une application
- Une application de type webapp

Les dépendances
- Les starters

La configuration des propriétés
- Les propriétés
- L'utilisation de fichier .properties
- L'utilisation de fichier YAML
- La définition de valeurs aux propriétés
- La bannière ASCII

Le support de Spring Boot dans STS

Programme détaillé

SPRING BOOT, REST ET SECURITY (2 / 2)

Durée : 3 jours

Contenu

Spring Boot DevTools

- Des propriétés par défaut
- Le redémarrage automatique de l'application
- Le débogage distant
- Le support du Live Reload
- La persistance des sessions HTTP entre les redémarrages

Mise en œuvre de fonctionnalités

- REST
- Spring Security
- Basic auth
- JWT
- Les tests d'intégration
- Le logging
- Le cache
- Le scheduling
- Les Servlets

Le déploiement d'une application

- Le packaging
- L'exécution d'une application
- Une application Autoexecutable
- Les Profiles

Spring Boot Actuator

- L'activation
- Les endpoints
- Les métriques personnalisées

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

QUARKUS

Durée : 3 jours

Objectif général

- Savoir développer des services REST en architecture microservice
- Savoir dialoguer avec une solution de messaging asynchrone
- Savoir accéder à une base de données de façon non bloquante
- Savoir sécuriser son API

Contenu

Introduction et vue d'ensemble de PowerShell

- Installer PowerShell
- Vue d'ensemble des objets
- Travailler avec les Cmdlets
- Complétion, Alias et Historique
- Les variables et les types
- Présenter les informations avec un formatage spécifique

Quarkus ? facile !

- Support des standards
- Convention over configuration
- Outils de développement (DevServices, LiveReload, Continuous Testing)

Quarkus API Rest

- REST avec JAX-RS
- implémentation ReastEasy
- Gestion centralisée des erreurs (@Provider)
- Tests avec RestAssured (@QuarkusTest)
- Documentation avec OpenAPI
- Accéder à des services REST distants (@RegisterRestClient)

Quarkus & bases de données relationnelles

- Configurer l'accès à la base de données
- JPA / Hibernate / PanacheORM
- Data Caching (@CacheResult)

Quarkus Sécurité

- Authentification via OAuth 2
- Authorization (@RolesAllowed, @PermitAll, @DenyAll, @Authenticated, @TestSecurity)

Programme détaillé

DEVOPS

AJC INGÉNIERIE est une S.A.R.L. au capital de 7.622 € R.C.S CRETEIL 414 248 203 00047
N° de déclaration d'activité 11 79 51 886 75 (ce numéro ne vaut pas agrément de l'état)

161 Avenue de Verdun 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

DEVOPS : FONDAMENTAUX

Durée : 1 jour

Objectif général

- Appréhender l'intérêt de la culture DevOps
- Découvrir les patterns de conception DevOps
- Identifier les enjeux de l'industrialisation des déploiements applicatifs
- Savoir fluidifier les interactions entre les différentes équipes projet
- Mettre en place des chaines de production plus fiables

Contenu

Origines de DevOps

- Les nouvelles exigences du marché
- La réponse des Géants du Web
- Définition de DevOps

Rappels sur l'agilité

- Les valeurs fondatrices du Manifeste agile
- Les rôles de l'équipe agile
- Les promesses de l'agile
- Scrum : le processus et les rituels
- Kanban

Objectifs et définition de DevOps

- Constats : des douleurs récurrentes
- Biz, Dev et Ops : des points de vue différents mais un objectif commun
- DevOps : étendre l'agilité au monde de la production
- Redistribution des rôles entre Dev et Ops

Les 4 piliers de DevOps :

- Culture, méthode et organisation
  - Méthodes, rituels et attitudes
  - Modèles d'organisation : feature team & component team
  - L'obsession de la mesure
- Architectures et patterns
  - Patterns de scalabilité et de disponibilité
  - Patterns d'exploitabilité
  - Patterns de déploiement
  - Le Cloud : facilitateur de l'architecture DevOps
- L'infrastructure par le code
  - Définition : l'infrastructure par le code
  - Responsabilités des différents outils
  - Stratégies de déploiement et cycles de vie des composants
  - Cartographie des outils
  - Docker et son écosystème
- Construction et déploiement continu
  - Définition : déploiement continu
  - Usine d'intégration et de déploiement en continu
  - La chaine CI/CD dans le monde du IaaS
  - La chaine CI/CD dans le monde du PaaS

Programme détaillé

DOCKER (1 / 2)

Durée : 3 jours

Objectif général

- Appréhender l'intérêt de la culture DevOps
- Connaître les caractéristiques d'un conteneur Linux et découvrir Docker
- Installer et utiliser Docker
- Maîtriser la création d'images
- Connaître et configurer une Registry (publique et privée)
- Maîtriser les notions réseaux de Docker (drivers, links)
- Comprendre et maîtriser la persistance des données (drivers, volumes)
- Maîtriser la notion de service Docker avec Docker-compose

Contenu

Introduction

- Revue des valeurs et principes de l'agilité
- Livraison continue et apport du mouvement DevOps
- Organisation des environnements de projet (local, dev, build, staging, prod…)
- Démarche qualité, gestion de version et des configurations

Appréhender la virtualisation avec Docker

- Machine de développeur unique, multiples environnements
- Les différentes formes de virtualisation et leur concept
- Présentation des avantages et des cas d'utilisation des conteneurs
- Présentation de Docker et de son architecture
- Cas de Windows et MacOS

Exécuter un projet dans Docker

- Installer Docker
- Build et exécution d'un projet au sein d'un conteneur
- Découvrir le Dockerfile
- Comprendre le cycle de vie du conteneur
- Administrer et superviser un conteneur depuis le docker host (exec, inspect, logs…)

Manipuler des images Docker

- Présentation du concept d'images Docker (Docker Hub, images personnalisées)
- Les différentes méthodes de conception d'une image Docker
- Créer une image à partir d'un conteneur (commit)
- Créer une image à partir d'un Dockerfile
- Les instructions dans un Dockerfile (FROM, COPY, ADD, EXPOSE, ENTRYPOINT, CMD)
- Gérer le cycle de vie des images (labels, tags, versionning mineur/majeur)
- Sélectionner et récupérer une image depuis la communauté "Docker Hub"
- Le concept des layers et du cache (optimisation)
- La registry et le stockage des images (registry privée, registry "Docker Hub")

Programme détaillé

DOCKER (2 / 2)

Durée : 3 jours

Contenu

Configurer le réseau pour Docker

- Le conteneur dans son réseau (stack réseau Docker)
- Le port forwarding (PAT)
- Liaisonner des conteneurs (links)
- Les différents réseaux proposés par Docker (drivers, les impacts et cloisonnements)

Gérer les systèmes de fichier pour Docker

- Le principe de volumes associés à un conteneur
- Créer et persister des volumes docker
- Gérer les modèles de configuration et leurs bonnes pratiques

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

JENKINS (1 / 2)

Durée : 1 jour

Objectif général

- Comprendre les principes d'intégration continue
- Intégrer Jenkins avec les autres outils (SCM, gestionnaire de tickets…)
- Mettre en place un serveur Jenkins automatisant les build
- Automatiser les tests, les audits de code et les déploiements sur la plate-forme d'intégration Jenkins
- Déployer Jenkins sur les projets
- Administrer l'architecture Jenkins

Contenu

L'Intégration Continue

- Définition, principes
- Notions de génie logiciel
- Best practices d'intégration continue
- La chaîne de fabrication logicielle

Utilisation de Jenkins

- Concepts, définitions
- Présentation de Jenkins comme serveur de build
- Archétype de projet
- Déclencheurs de build
- Résultat du build
- Workspace
- Visite guidée de l'interface
- Jenkins dans l'IDE
- Installation et démarrage de Jenkins
- Configuration générale
- Installation des plugins

Construire un projet Java avec Maven

- Rappels Maven
- Création d'un job
- Accès aux sources
- Paramétrage de Maven
- Rapports de test unitaires
- Envoi de mails de notification
- Déploiement automatique
- Rapports d'analyse qualité
- Habilitations

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

JENKINS (2 / 2)

Durée : 1 jour

Contenu

Configurer le réseau pour Docker
- Le conteneur dans son réseau (stack réseau Docker)
- Le port forwarding (PAT)
- Liaisonner des conteneurs (links)
- Les différents réseaux proposés par Docker (drivers, les impacts et cloisonnements)

Construction des projets complexes
- Enchaînements de projets Maven
- Construire une application J2EE complète
- Construire un projet avec Ant
- Conjuguer plusieurs outils
- Déployer dans les référentiels Maven
- Piloter le déploiement d'applications

Utilisation de Jenkins en Cluster
- Configuration des esclaves
- Modes de démarrage Unix, Windows
- Répartition des jobs entre esclaves
- Bonnes pratiques de déploiement

Administration de Jenkins
- Configuration des vues Jenkins
- Considérations multi plates-formes
- Visite guidée de la JENKINS_HOME
- Monitorer et sauvegarder Jenkins
- Scripts Jenkins en langage Groovy
- Utiliser la ligne de commande d'administration

161 Avenue de Verdun. 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

PROJET

AJC INGÉNIERIE est une S.A.R.L. au capital de 7.622 € R.C.S CRETEIL 414 248 203 00047
N° de déclaration d'activité 11 79 51 886 75 (ce numéro ne vaut pas agrément de l'état)

161 Avenue de Verdun 94200 Ivry Sur Seine - Tél.: 01 75 43 86 72
formonsnous@ajc-ingenierie.fr - www.trouveljob.fr - www.ajc-ingenierie.fr

Programme détaillé

Projet Partie Finale

Durée : 5 jours

Objectif général

- Permettre aux participants de mettre en œuvre tout ce qu'ils ont appris au cours des sessions de formations précédentes.
- Savoir développer une architecture en couche à forte valeur ajoutée en privilégiant les interfaces.
- Apprendre à gérer les risques d'un projet et faire des choix de conception adaptés au problème.
- Apprendre à effectuer des tests de validation.
- Réaliser un ou plusieurs rédactionnels de suivi de projet.

Contenu

Déroulement du module

- Préparation à la communication orale et écrite pour la soutenance du projet
- Les stagiaires travaillent en toute autonomie, en groupe. Ils sont libres d'effectuer les choix adaptés, de développer les parties dont ils jugent avoir le plus besoin et d'apporter leurs propres solutions aux problèmes posés.
- Le formateur encadre les stagiaires par sa présence et répond aux questions. Il intervient pour épauler un groupe en difficulté ou pour faire le point à l'ensemble du groupe sur des notions non acquises. Il peut être amené à approfondir ou compléter certaines connaissances.

À votre disposition
pour discuter du futur

Contact

161 Avenue de Verdun,
94200 Ivry Sur Seine

Tel : 01 75 43 86 72

formonsnous@ajc-ingenierie.fr

www.ajc-ingenierie.fr
www.trouveljob.fr