

Java niveau intermédiaire

Enrichir et approfondir sa pratique des principales APIs Java

Filière : **Développement logiciel** Sous-filière : **Java & XML**

RÉFÉRENCE	DURÉE	PRIX UNITAIRE HT
JA-PGI	3 JOURS (21H)	2 050 €

Objectifs pédagogiques

- Consolider les connaissances acquises lors du cours Java Base
- Passer en revue les packages essentiels du développeur Java

Public cible

- Architectes
- Consultants
- Développeurs
- Chefs de projets
- Concepteurs

Pré-requis

Avoir suivi la formation Java les bases ou posséder les connaissances équivalentes

Programme de la formation

Syntaxe avancée

- Les classes internes et anonymes
- Les constructeurs
- Les blocs d'initialisation
- Les types génériques
- La syntaxe des lambdas (Java 8)
- La syntaxe des références de méthode (Java 8)

OXiane Institut

98 avenue du général Leclerc
92100 Boulogne-Billancourt

RCS Nanterre 430 112 250 000 21 / Code NAF 6202A
Organisme de formation N° 11 92 16 52 492



Le package java.lang

- La classe Object
- L'interface Comparable : égalité et comparaison
- L'interface Clonable : copie d'objet
- Les wrappers et l'autoboxing : conversions de type
- Les chaînes : String, CharSequence, Appendable, StringBuffer, StringBuilder, ...
- Les extensions syntaxiques : Iterable, AutoCloseable
- Les énumérations : classe Enum
- Les opérations mathématiques : Math et StrictMath
- Les annotations standards (@Deprecated, @Override, ...)
- Les exceptions : Throwable, Exception, Error, RuntimeException, ...
- Les classes utilitaires : System, Runtime, Process et ProcessBuilder
- Les bases du parallélisme : Runnable, Thread, ...
- L'introspection : Class, Package, ClassLoader ...
- Les autres éléments du package java.lang

Le package java.math

- Les nombres réels et les erreurs d'arrondis
- Les nombres étendus : BigInteger, BigDecimal
- La gestion des arrondis : MathContext et RoundingMode

Le package java.util

- Les collections : Collection, List, Queue, Set, Map, ...
- Itérer sur les collections : Enumeration et Iteration
- Les classes d'implémentations de collections
- Les classes utilitaires : Collections et Arrays
- La gestion du temps : Date, Calendar, ...
- La représentation de la monnaie : Currency
- Le paramétrage : Properties
- L'internationalisation : Locale, ResourceBundle, Formatter, ...
- Les classes utilitaires : Scanner, StringTokenizer, Random, ...
- Les autres éléments du package java.util : Observer, Observable, ServiceLoader, ...

Le package java.text

- La comparaison des chaînes de caractères : Collator, RuleBasedCollator
- Le formatage textuel : Format, MessageFormat, NumberFormat, DateFormat, ...

OXiane Institut



Le package `java.io`

- La gestion de fichiers : `File`, `FileFilter`, `FilenameFilter`, ...
- La gestion des flux binaires : `InputStream`, `OutputStream`, ...
- La gestion des flux textes : `Reader`, `Writer`, ...
- La gestion des flux d'objets : `Serializable`, `Externalizable`, ...
- Les classes utilitaires : `Console`, `StreamTokenizer`, `RandomAccessFile`

Le package `java.nio`

- La « nouvelle » gestion de fichiers : `FileStore`, `FileSystem`, `Path`, `FileSystems`, `Files`, `Paths`, `PathMatcher`, `WatchService`, ...
- Les transferts de données : `Buffer`, `Channel`, `Channels`, ...

Le package `java.net`

- Les accès réseau : `URL`, `URLConnection`, `URLEncoder`, `URLDecoder`
- Les protocoles TCP et UDP : `InetAddress`, `NetworkInterface`, `Socket`, `ServerSocket`, ...

Le package `java.sql`

- Le framework JDBC : `DriverManager`, `Connection`, `Statement`, `PreparedStatement`, `ResultSet`, ...

Le package `java.util.concurrent`

- Le parallélisme avancé (`Futur`, `Executor`, `ExecutorService`, `Executors`, ...)
- Les collections synchronisées : `BlockingQueue`, `ConcurrentMap`, ...
- Le package `java.util.concurrent.atomic` : les conteneurs thread-safe
- Le package `java.util.concurrent.locks` : la gestion explicite des locks

Le package `java.time` (Java 8)

- Les nouvelles classes temporelles : `Instant`, `Duration`, `LocalDate`, `LocalTime`, `Period`, `YearMonth`, `Temporal`, ...

Les packages `java.util.function` et `java.util.stream` (Java 8)

- Les `FunctionalInterface` : `Consumer`, `Predicate`, `Fonction`, `Supplier`, ...
- Les interfaces de streams : `BaseStream`, `Stream`, `Collector`, ...
- La construction des streams : `stream()`, `parallelStream()`, `iterate()`, `generate()`, ...
- Les fonctions d'aggrégats : `forEach`, `filter`, `sorted`, `map`, `collect`, ...

Aperçu de quelques autres packages

- Le package `java.util.logging` : les traces
- Le package `java.util.prefs` : la gestion des préférences utilisateurs

OXiane Institut



-
- Le package java.util.jar : la gestion des jar
 - Le package java.util.zip : la gestion des zip
 - Le package java.util.regex : les expressions régulières
 - Le package java.awt : les interfaces graphiques natives
 - Le package javax.swing : les interfaces graphiques riches

Qualité

Cette formation est accessible aux personnes en situation de handicap, nous contacter en cas de besoin d'informations complémentaires.



Programme mis à jour le **7 novembre 2023**