### ****TP 2 : Automatisation Complète d'une Application Java avec Apache Ant****

**Durée :** 3 heures  
**Objectif :** Concevoir une application Java plus avancée et automatiser son cycle de vie avec Apache Ant : **compilation, tests, packaging, documentation et déploiement**.

## ****📌 Partie 1 : Mise en place du projet (30 min)****

### Instructions :

### ****Structure du projet****

Créer un projet avec l’arborescence suivante :

TP\_Ant\_Advanced/

│── src/

│ ├── main/

│ │ ├── java/

│ │ │ ├── com/

│ │ │ │ ├── app/

│ │ │ │ │ ├── Main.java

│ │ │ │ │ ├── Calculator.java

│ ├── test/

│ │ ├── java/

│ │ │ ├── com/

│ │ │ │ ├── app/

│ │ │ │ │ ├── CalculatorTest.java

│── lib/

│── build/

│── dist/

│── docs/

│── build.xml

* src/main/java/ : Code source de l’application.
* src/test/java/ : Fichiers de tests unitaires.
* lib/ : Librairies externes (ex. JUnit).
* build/ : Dossier de compilation.
* dist/ : Contient les fichiers .jar générés.
* docs/ : Documentation JavaDoc.
* build.xml : Fichier de build Ant.

## ****📌 Partie 2 : Développement de l'application****

1. Créer la classe Calculator avec des méthodes pour additionner, soustraire, multiplier et diviser des entiers.
2. Créer une classe Main qui utilise la classe Calculator pour effectuer des calculs et afficher les résultats.

### ****Partie 3 : Création de la classe de test****

### ****Création de la classe de test CalculatorTest****

1. Créer la classe de test CalculatorTest.java qui va tester les méthodes de la classe Calculator. Dans src/test/java/com/app/,

## ****📌 Partie 4 : Automatisation avec Ant****

### ****Configuration du fichier**** build.xml

1. Créez un fichier build.xml pour automatiser les différentes étapes de votre projet avec Ant.
2. Configurez la tâche init pour créer les répertoires nécessaires (build, dist, docs).
3. Ajoutez une tâche build pour compiler votre code source avec javac.
4. Ajoutez une tâche run pour exécuter l'application principale.
5. Ajoutez une tâche test pour exécuter les tests unitaires avec JUnit.
6. Ajoutez une tâche doc pour générer la documentation JavaDoc.
7. Ajoutez une tâche jar pour générer un fichier .jar contenant votre application.

## ****📌 Partie 5 : Génération et déploiement****

1. Configurez Ant pour exécuter les tests unitaires en utilisant la tâche test.
2. Configurez Ant pour générer un fichier .jar de votre application avec la tâche jar.
3. Configurez Ant pour générer la documentation JavaDoc avec la tâche doc.
4. Configurez Ant pour nettoyer votre projet en supprimant les fichiers compilés et la documentation avec la tâche clean.