

Passos para Configuração no Visual Studio Code

1. Instalar o JDK (Java Development Kit):

- Baixe e instale o JDK apropriado para o seu sistema operacional a partir do site oficial da Oracle ou de outra distribuição de JDK.

2. Instalar o Visual Studio Code:

- Baixe e instale o VS Code a partir do site oficial: [Visual Studio Code](#).

3. Instalar Extensões no VS Code:

- Abra o VS Code e instale as seguintes extensões:
 - **Java Extension Pack:** Inclui ferramentas essenciais para desenvolvimento em Java, como o suporte a linguagem, depuração e gerenciamento de projetos.
 - **Weka:** Certifique-se de que o arquivo JAR do Weka está disponível para o seu projeto.

4. Configurar o Projeto no VS Code:

- Crie uma nova pasta para o seu projeto Java no VS Code.
- Adicione o arquivo transacoes.arff na raiz da pasta do projeto.
- Crie um arquivo FraudDetection.java na mesma pasta.

5. Adicionar o JAR do Weka ao Projeto:

- Baixe o arquivo JAR do Weka a partir do site oficial.
- Coloque o arquivo JAR na pasta do projeto ou em uma subpasta como lib.
- Certifique-se de configurar o caminho do JAR no classpath do seu projeto.

Código Java para Detecção de Fraude

```
import weka.core.Instances;
import weka.core.converters.ConverterUtils.DataSource;
import weka.classifiers.Classifier;
import weka.classifiers.trees.J48;
import weka.core.Instance;

public class FraudDetection {

    public static void main(String[] args) {
        try {
            // Carregar o dataset de transações do arquivo transacoes.arff
            DataSource source = new DataSource("transacoes.arff");
            Instances dataset = source.getDataSet();
```

```
import weka.core.Instances;
import weka.core.converters.ConverterUtils.DataSource;
import weka.classifiers.Classifier;
import weka.classifiers.trees.J48;
import weka.core.Instance;

public class FraudDetection {

    public static void main(String[] args) {
        try {
            // Carregar o dataset de transações do arquivo transacoes.arff
            DataSource source = new DataSource("transacoes.arff");
            Instances dataset = source.getDataSet();

            // Definir o índice do atributo classe (fraude ou não fraude)
            dataset.setClassIndex(dataset.numAttributes() - 1);

            // Treinar o classificador J48 (árvore de decisão)
            Classifier classifier = new J48();
            classifier.buildClassifier(dataset);

            // Exibir o modelo treinado
            System.out.println(classifier);

            // Classificar uma nova transação
            Instance novaTransacao = dataset.firstInstance(); // Exemplo de nova transação
            double classificacao = classifier.classifyInstance(novaTransacao);
            System.out.println("Classificação: " + dataset.classAttribute().value((int) classificacao));

        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Executando o Código no VS Code

1. **Abrir o Terminal no VS Code:**
 - Vá para Terminal > New Terminal no menu do VS Code.
2. **Compilar o Código Java:**
 - No terminal, navegue até a pasta do projeto e compile o código, certificando-se de incluir o JAR do Weka no classpath:

```
javac -cp ".:weka.jar" FraudDetection.java
```

```
javac -cp ".:weka.jar" FraudDetection.java
```

Executar o Código Java:

- Execute o programa com o seguinte comando:

```
java -cp ".:weka.jar" FraudDetection
```

```
java -cp ".:weka.jar" FraudDetection
```

Detalhes do Código

1. Carregar o Dataset:
 - Utiliza DataSource para carregar os dados do arquivo transacoes.arff.
2. Definir o Atributo Classe:
 - Define o último atributo do dataset (fraud) como a classe a ser prevista.
3. Treinar o Classificador:
 - Utiliza a árvore de decisão J48 para treinar o modelo.
4. Classificar Novas Transações:
 - Classifica uma nova transação e imprime o resultado.

Referência:

<https://chatgpt.com/>