Instructions for Setting Up and Running the Blockchain Project

Introduction

This guide will help you set up and run the AIBTCCoin blockchain project. Follow the steps below to get your development environment ready and execute the blockchain application for the first time.

Prerequisites

- **Node.js and npm**: Ensure you have Node.js (version 12 or higher) and npm installed.
- **MySQL**: Install MySQL on your system and make sure it's running.

Step 1: Clone the Repository

Clone the repository to your local machine using the following command:

```
git clone https://github.com/J-Oliver12/AIBTCCoin-main
```

Navigate to the project directory:

```
cd AIBTCCoin
```

Step 2: Install Dependencies

Install the required Node.js packages:

```
npm install
```

Step 3: Set Up MySQL Database

1. **Create Database and Tables**:
    - Open MySQL command-line client or any MySQL GUI tool.
    - Execute the `create_tables.sql` script to set up the database and tables.

```
CREATE DATABASE blockchain;
USE blockchain;

CREATE TABLE blocks (
    id INT AUTO_INCREMENT PRIMARY KEY,
    hash VARCHAR(255) NOT NULL UNIQUE,
    previous_hash VARCHAR(255),
    timestamp BIGINT,
    nonce INT,
    difficulty INT
);

CREATE TABLE transactions (
    id INT AUTO_INCREMENT PRIMARY KEY,
    hash VARCHAR(255) NOT NULL UNIQUE,
    block_hash VARCHAR(255) NOT NULL,
    from_address VARCHAR(255),
    to_address VARCHAR(255),
    amount DECIMAL(18, 8),
    timestamp BIGINT,
    signature TEXT,
    FOREIGN KEY (block_hash) REFERENCES blocks(hash)
);
```

Step 4: Configure Database Connection

Update the database connection details in `src/db.js`:

```
const mysql = require('mysql2');
const db = mysql.createConnection({
  host: 'localhost',
  port: 3306,
  user: 'root',  // Replace 'root' with your actual MySQL user
  password: 'yourpassword',  // Replace 'yourpassword' with your actual MySQL password
  database: 'blockchain'
});
db.connect(err => {
  if (err) {
    throw err;
  }
  console.log('Connected to database');
});
module.exports = db;
```

Step 5: Generate Key Pair

Generate a key pair for signing transactions. Run the `keygenerator.js` script:

```
node src/keygenerator.js
```

This script will output a public key and a private key. Keep these keys secure as they will be used for creating and signing transactions.

Step 6: Run the Blockchain Application

Execute the main application:

```
node index.js
```

This will perform the following actions:

- Create an initial blockchain with a genesis block.
- Add an initial balance to your public key address.
- Mine the initial pending transactions.
- Create and sign a new transaction.
- Mine the new transactions.
- Print the balances of the involved addresses.