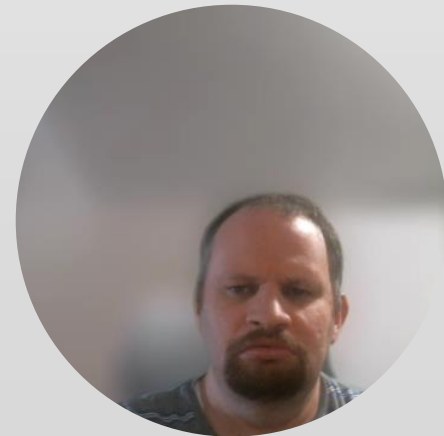


Interoperability Technology in Healthcare



Interoperability Tech and Healthcare Defined

- Exchange and Integration of healthcare data
- Secure access, sharing and use of HI
- Exchange Standards: HL7, FHIR, CDAs, CCDAs, ICD Codes
- Methods of Exchange
- Semantic Interoperability
- TEFCA and QHINS



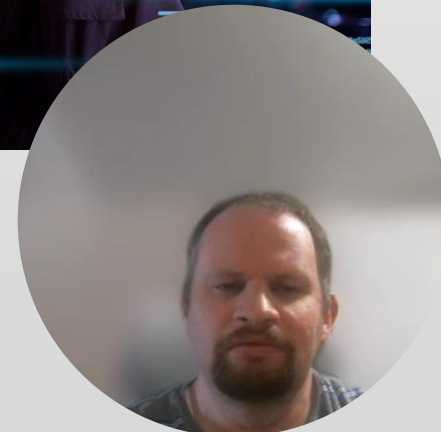
HL7 FHIR Standard

- Restful APIs
- Sandbox Environments
 - Mock Scenario Testing
 - App Development
 - Innovation
- Benefits of Sandboxes
 - Prototyping and Testing
 - Education and Training
 - Standard Compliance
 - Collab



Trends of Interoperability in Healthcare

- 21st Century Cures Act and Patient Access to APIs
- Data integration and aggregation
- IoMT
 - IEEE 11073 and IHE
- Clinical Decision Support
 - Point of Care
- Secure Data Exchange and Privacy
 - Messaging Protocols, encryption, and access controls



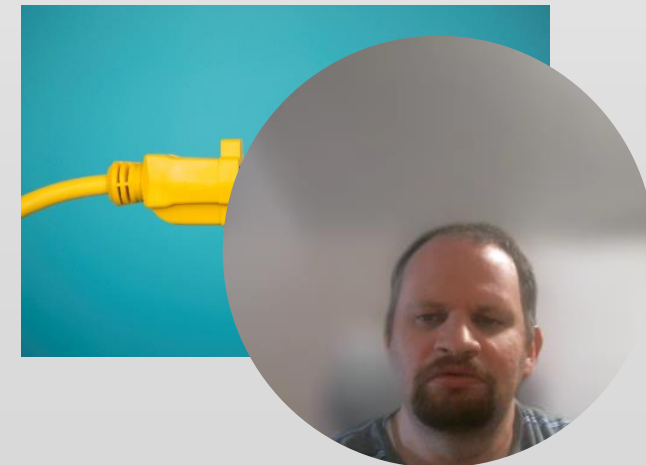
Interoperability Opportunities in Healthcare

- Improved care coordination
- Enhanced Patient Engagement
- Aggregation and Analysis of Large Datasets



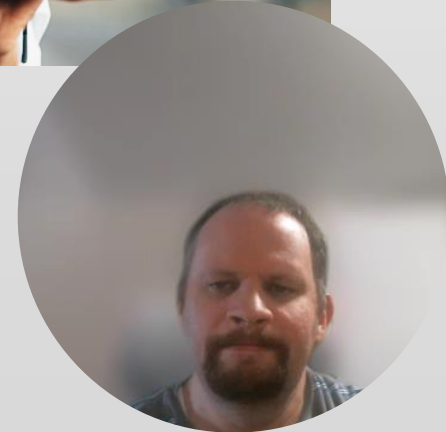
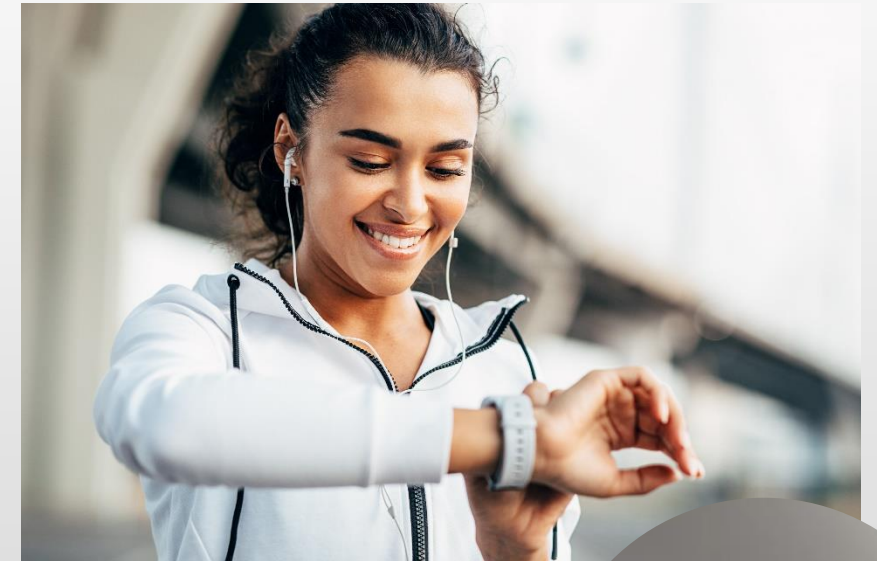
Interoperability Threats in Healthcare

- Data Privacy and Security
- Information exchange between disparate systems
- Technical Barriers
 - Incompatible Data Standards
 - Differing System Interfaces
 - Variations in Data Capturing Practices
- Resistance to Change, Awareness, Stakeholder Buy-in



Cotiviti's Explorations

- QHINs
- Access of COBs to patients
 - Easy to understand
 - Patient Empowerment
 - Patient Centered Dashboard Design
- IoMT Technology
 - Fraud Waste and Abuse of Devices, Rx, etc.



Block Chain Technology

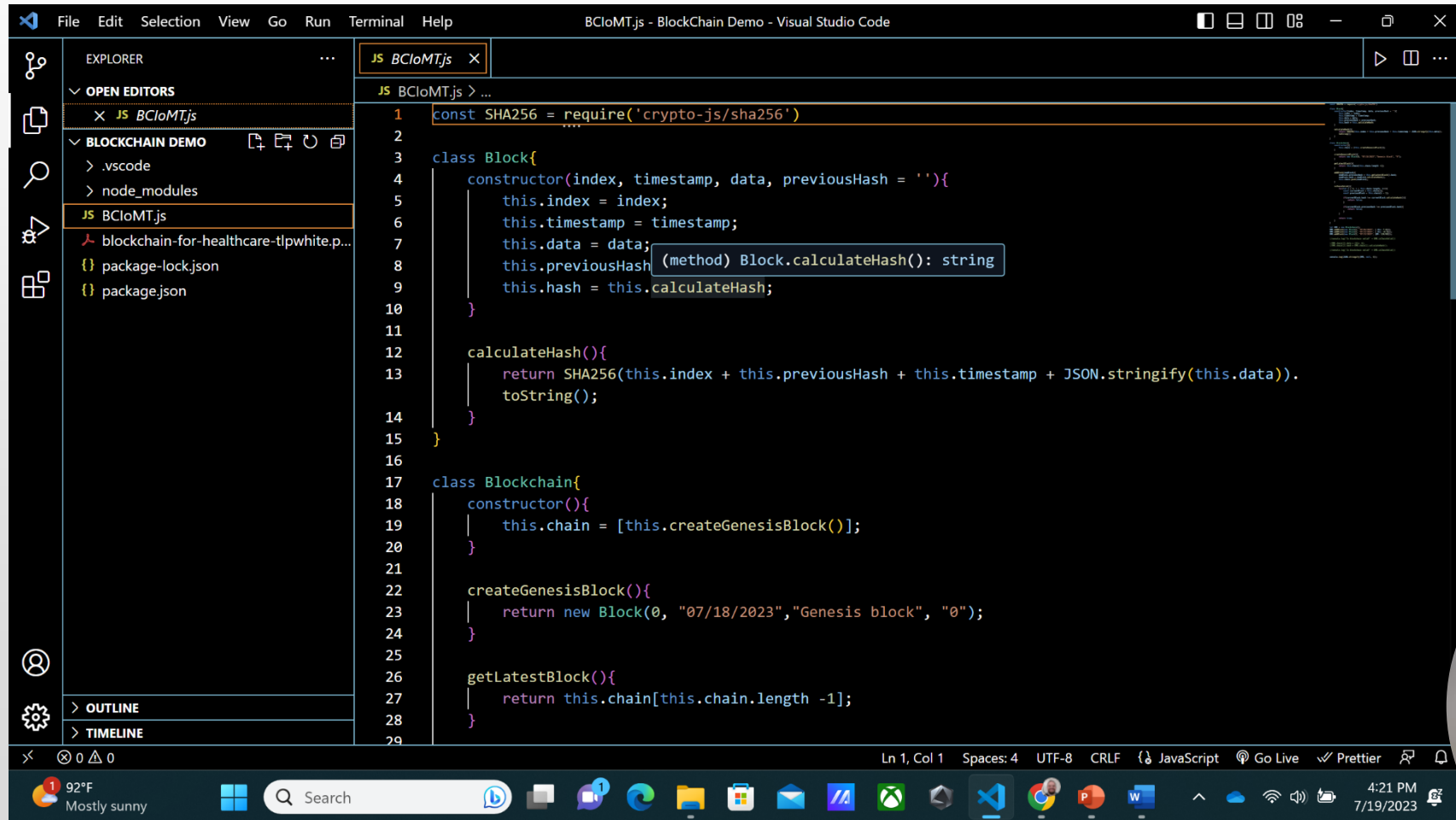
- Originally used for bitcoin
- A perspective technology for IoMT Tracking
 - Secure
 - Collaborative
 - Way to monitor against claims to determine proper use of medical devices
 - ie. CGM, BP

https://www.youtube.com/watch?v=SSo_ElwHSd4

https://www.youtube.com/watch?v=zVqczFZr124&list=PLU5Yn2kbD2-__keY75aBKkH5FmRRn_fE9&index=1&t=600



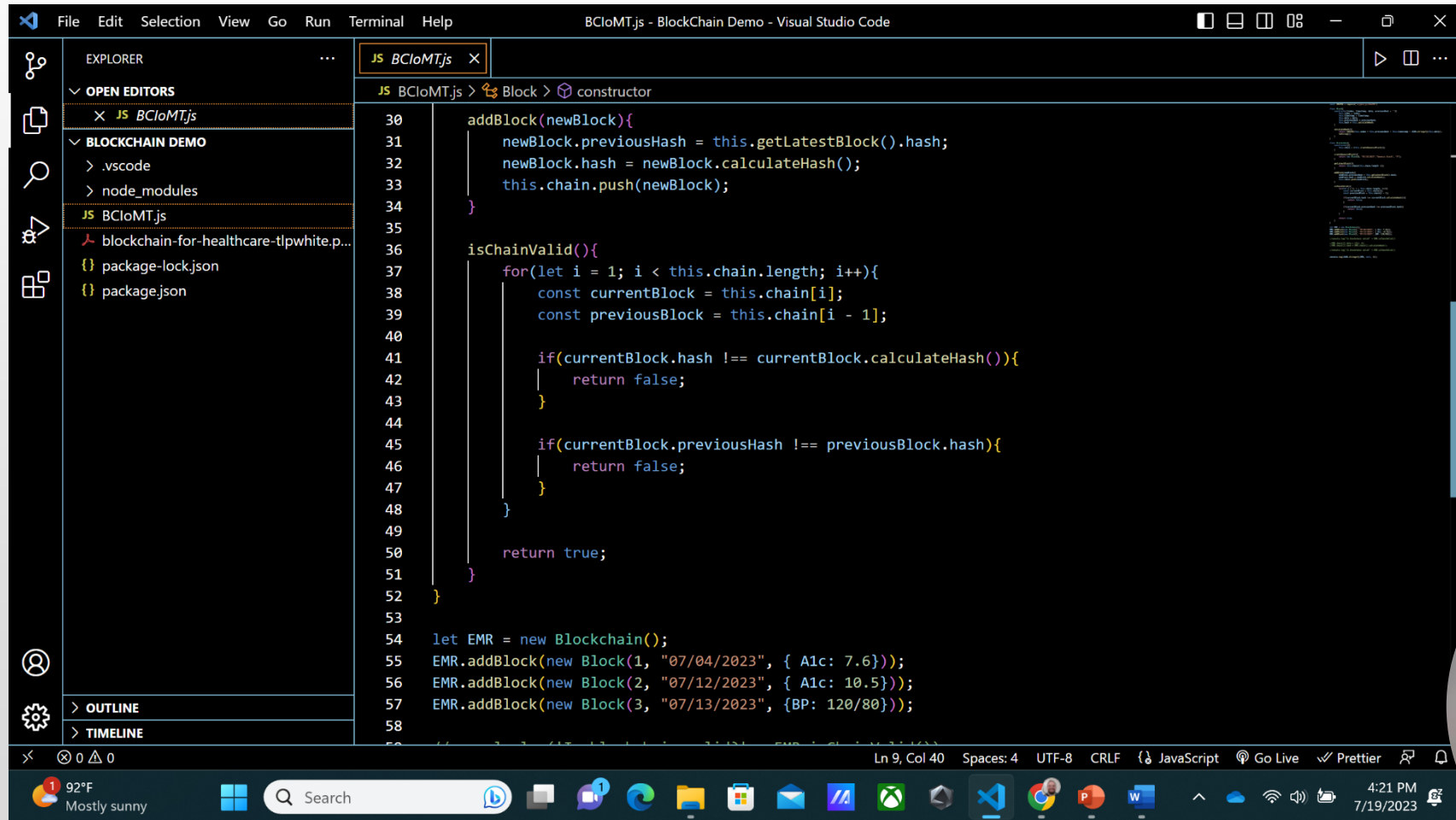
Block Chain Demo Code



```
1 const SHA256 = require('crypto-js/sha256')
2
3 class Block{
4   constructor(index, timestamp, data, previousHash = ''){
5     this.index = index;
6     this.timestamp = timestamp;
7     this.data = data;
8     this.previousHash = (method) Block.calculateHash(): string
9     this.hash = this.calculateHash;
10  }
11
12  calculateHash(){
13    return SHA256(this.index + this.previousHash + this.timestamp + JSON.stringify(this.data)).
14      toString();
15  }
16
17  class Blockchain{
18    constructor(){
19      this.chain = [this.createGenesisBlock()];
20    }
21
22    createGenesisBlock(){
23      return new Block(0, "07/18/2023", "Genesis block", "0");
24    }
25
26    getLatestBlock(){
27      return this.chain[this.chain.length -1];
28    }
29  }
```



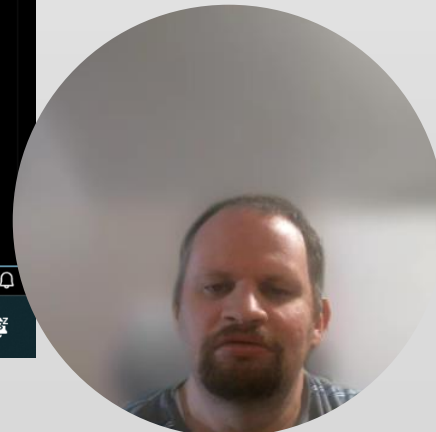
Block Chain Demo Code



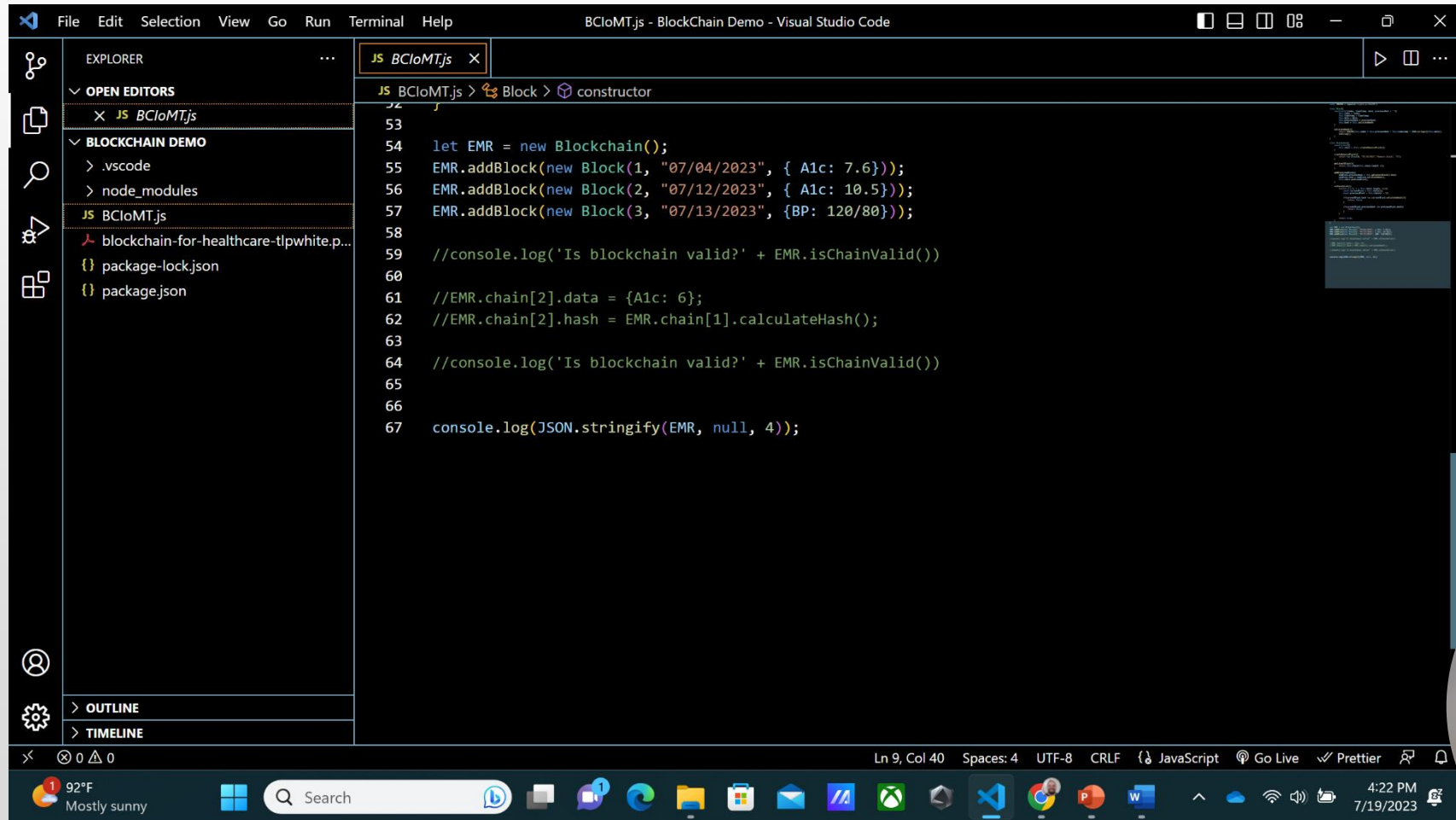
The screenshot displays the Visual Studio Code editor with the file `BClonMT.js` open. The Explorer sidebar on the left shows the project structure, including the `BLOCKCHAIN DEMO` folder and the `BClonMT.js` file. The main editor area shows the following JavaScript code:

```
30     addBlock(newBlock){
31         newBlock.previousHash = this.getLatestBlock().hash;
32         newBlock.hash = newBlock.calculateHash();
33         this.chain.push(newBlock);
34     }
35
36     isChainValid(){
37         for(let i = 1; i < this.chain.length; i++){
38             const currentBlock = this.chain[i];
39             const previousBlock = this.chain[i - 1];
40
41             if(currentBlock.hash !== currentBlock.calculateHash()){
42                 return false;
43             }
44
45             if(currentBlock.previousHash !== previousBlock.hash){
46                 return false;
47             }
48         }
49
50         return true;
51     }
52 }
53
54 let EMR = new Blockchain();
55 EMR.addBlock(new Block(1, "07/04/2023", { A1c: 7.6}));
56 EMR.addBlock(new Block(2, "07/12/2023", { A1c: 10.5}));
57 EMR.addBlock(new Block(3, "07/13/2023", {BP: 120/80}));
58
```

The status bar at the bottom indicates the current position is Line 9, Column 40, with 4 spaces, UTF-8 encoding, and CRLF line endings. The bottom of the screen shows the Windows taskbar with the date and time as 4:21 PM on 7/19/2023.



Block Chain Demo Code



```
File Edit Selection View Go Run Terminal Help BClomT.js - BlockChain Demo - Visual Studio Code

EXPLORER
OPEN EDITORS
  JS BClomT.js
BLOCKCHAIN DEMO
  .vscode
  node_modules
  JS BClomT.js
  blockchain-for-healthcare-tlpwhite.p...
  package-lock.json
  package.json

OUTLINE
TIMELINE

JS BClomT.js
52
53
54 let EMR = new Blockchain();
55 EMR.addBlock(new Block(1, "07/04/2023", { A1c: 7.6}));
56 EMR.addBlock(new Block(2, "07/12/2023", { A1c: 10.5}));
57 EMR.addBlock(new Block(3, "07/13/2023", {BP: 120/80}));
58
59 //console.log('Is blockchain valid?' + EMR.isChainValid())
60
61 //EMR.chain[2].data = {A1c: 6};
62 //EMR.chain[2].hash = EMR.chain[1].calculateHash();
63
64 //console.log('Is blockchain valid?' + EMR.isChainValid())
65
66
67 console.log(JSON.stringify(EMR, null, 4));
```

