A picture containing text

Description automatically generated

**BlockChain Technology Lab**

**(20CP406P)**

**Lab ASSIGNMENT - 3**

Icon

Description automatically generated with medium confidence

B.Tech in Computer Science and Engineering Dept.,

Pandit Deendayal Energy University,

Gandhinagar

**Logo

Description automatically generated**

**Name: Vrushank Ariwala**

**Roll No.: 19BCP141**

**Branch: CSE**

* **Aim:-**

**Understand and Create Blocks with all the parameters and Link them with Hash Pointers**

* **Code:-**

from hashlib import sha256

def hash(p\_key, send\_addr, recv\_addr, amount):

    ans = sha256((p\_key+send\_addr+recv\_addr+amount).encode()).hexdigest()

    return ans

class node:

    def \_\_init\_\_(self, prev=None, data=None, next=None):

        self.prev = prev

        self.data = data

        self.next = next

        b = True

        for i in range(9999999):

            x = sha256((self.prev+data+str(i)).encode()).hexdigest()

            if x[:4] == '0000':

                self.hash = x

                self.nonce = i

                b = False

                break

        if b:

            self.nonce = -1

            self.hash = sha256((self.prev+data+str(1)).encode()).hexdigest()

class BLOCKCHAIN:

    def \_\_init\_\_(self):

        self.head = None

    def add\_end(self, data):

        # new\_node = node(data=data)

        if self.head is None:

            self.head = node(data=data, prev="0000000000000000000000000000000000000000000000000000000000000000")

            return

        last = self.head

        while last.next:

            last = last.next

        new\_node = node(data=data, prev=last.data)

        last.next = new\_node

        return

    def print\_all(self):

        current = self.head

        while current:

            print("data :", current.data)

            print("Hash :", current.hash)

            print("Nonce :", current.nonce)

            print()

            current = current.next

        return

a = BLOCKCHAIN()

a.add\_end('Hello')

a.add\_end('Sir')

a.add\_end(',')

a.add\_end('Assigment')

a.add\_end('not')

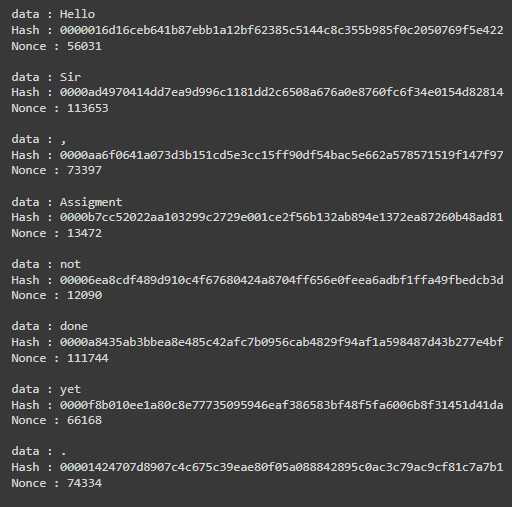
a.add\_end('done')

a.add\_end('yet')

a.add\_end('.')

a.print\_all()

* **Output:-**

****