

5) Writing Code (continued)

1. Initialize the Web3 Provider in **app.js** just after importing dependencies.

```
/*-----WEB3 Connection starts-----*/
var MyContractJSON = require(path.join(__dirname,
'build/contracts/Auction.json'));

const Web3 = require('web3')

web3 = new Web3(new
Web3.providers.HttpProvider('https://ropsten.infura.io/YOUR_INFURA
_ID'))

contractAddress = MyContractJSON.networks['3'].address;;
const contractAbi = MyContractJSON.abi;

MyContract = new web3.eth.Contract(contractAbi, contractAddress);

/*-----WEB3 Connection Ends-----*/
```

2. In the above specified code change the path to the contract json file and the Infura project ID.

3. Copy the below Javascript code to **routes/index.js**. (provided in zip file)

```
/*----- Function for serialize, sign and send transaction-----*/

const Tx = require('ethereumjs-tx');

function sendTransaction(methodCall, ethVal, addressFrom,
privateKey, cb) {
  const encodedABI = methodCall.encodeABI();
  const privateKeyHex = Buffer.from(privateKey, 'hex');

  web3.eth.getTransactionCount(addressFrom)
    .then(txCount => {
      var txData;
```



Server-Side Signed Txn (Part 4 of 4)

```
if (ethVal == null) {
  txData = {
    nonce: web3.utils.toHex(txCount),
    gasLimit: web3.utils.toHex(250000),
    gasPrice: web3.utils.toHex(1000),
    from: addressFrom,
    to: contractAddress,
    data: encodedABI
  }
} else {
  txData = {
    nonce: web3.utils.toHex(txCount),
    gasLimit: web3.utils.toHex(250000),
    gasPrice: web3.utils.toHex(1000),
    from: addressFrom,
    to: contractAddress,
    data: encodedABI,
    value: web3.utils.toHex(web3.utils.toWei(String(ethVal),
'ether'))
  }
}
const transaction = new Tx(txData);
transaction.sign(privateKeyHex);
const serializedTx = transaction.serialize().toString('hex');

web3.eth.sendSignedTransaction('0x' + serializedTx)
  .on('transactionHash', function (hash) {
    console.log('transactionHash: ' + hash);
  })
  .on('receipt', function (receipt) {
    console.log('receipt: ');
    console.log(receipt);
    cb(true);
  })
  .on('error', function (error) {
    console.log('error: ' + error);
    cb(false);
  })
  .catch(function (error) {
    console.log("====Web3 Exception Handled====\n");
    console.log(error);
    console.log("\n*=====*\n");
  });
})
}
```

/*-----*/



K B A

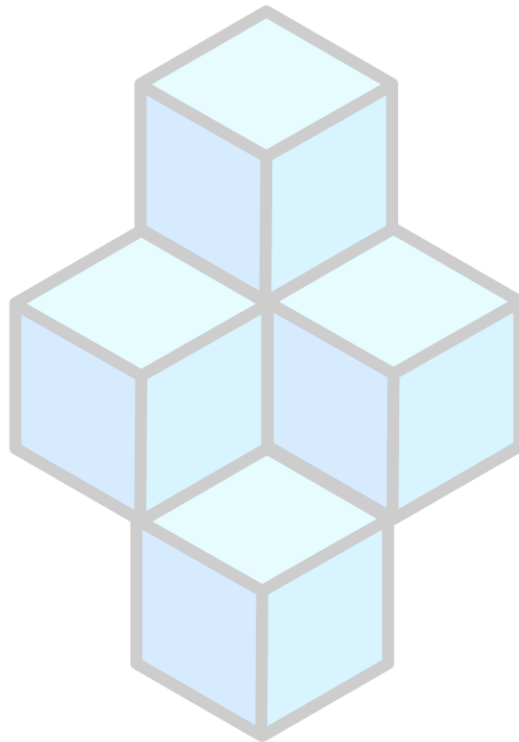
Server-Side Signed Txn (Part 4 of 4)

6) Run Code

Congrats!!! You have completed your coding. Now Start ExpressJS server using following command

```
$ npm start
```

Open <http://localhost:3000> in your browser.



K B A