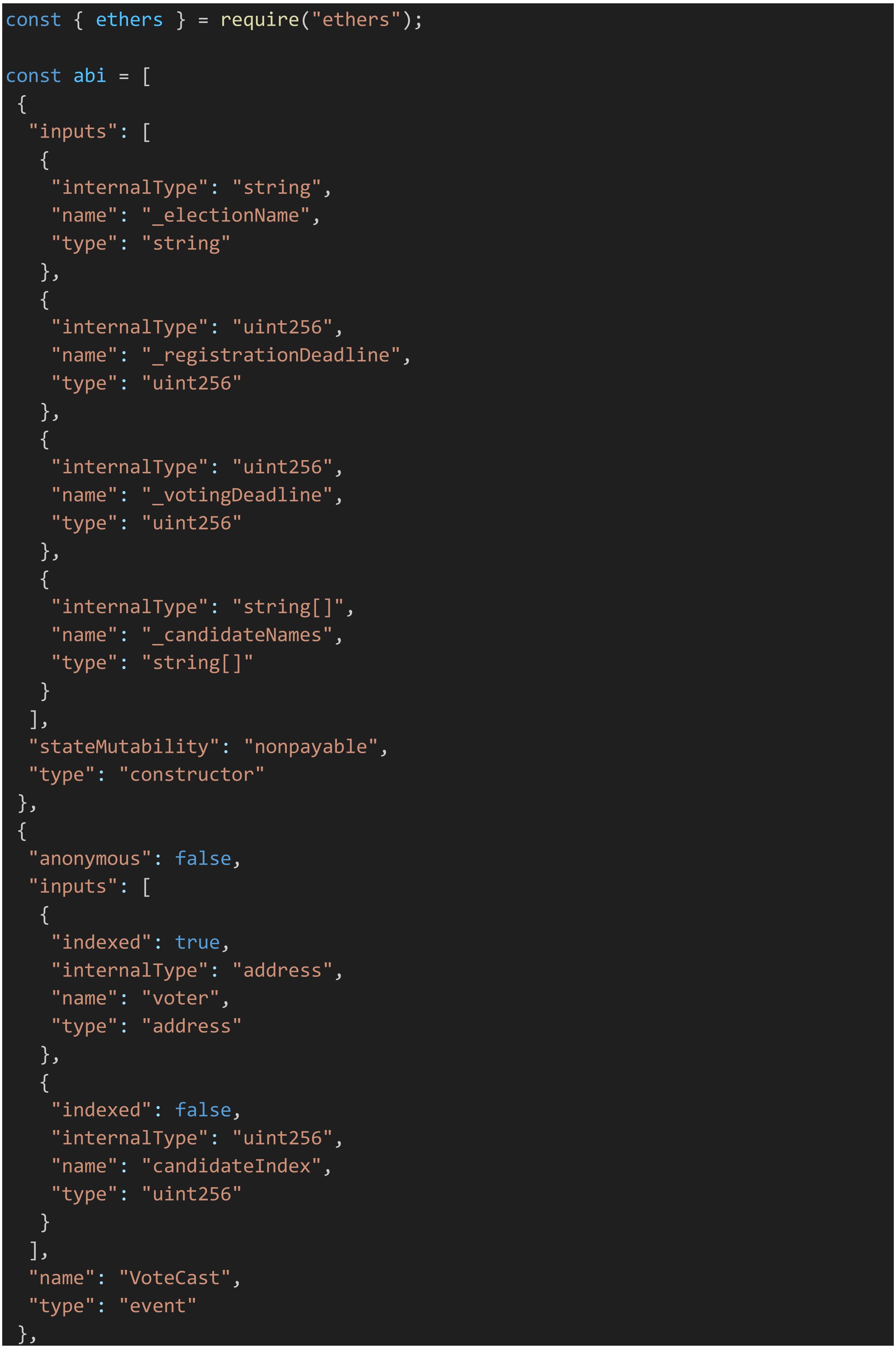
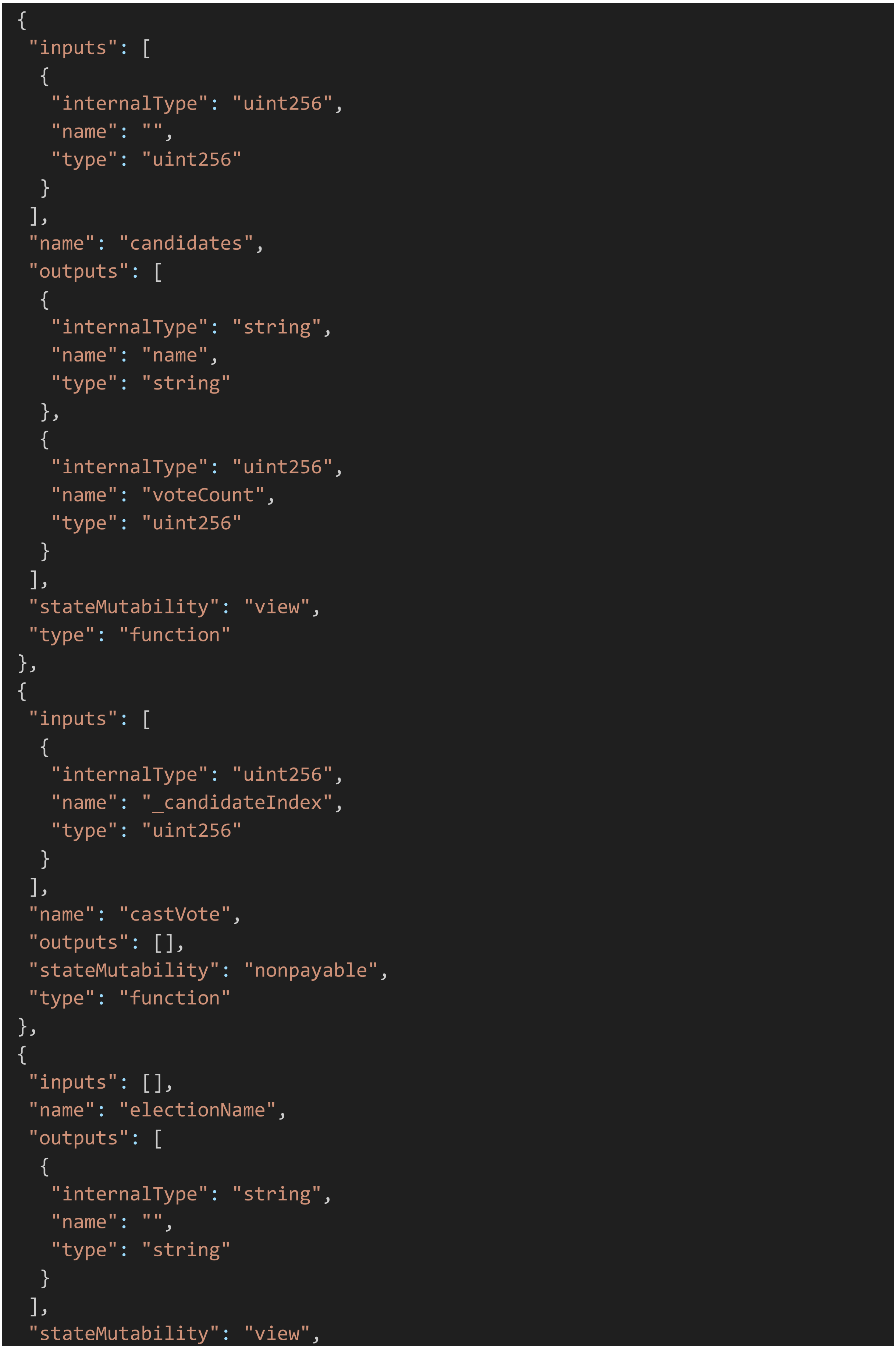
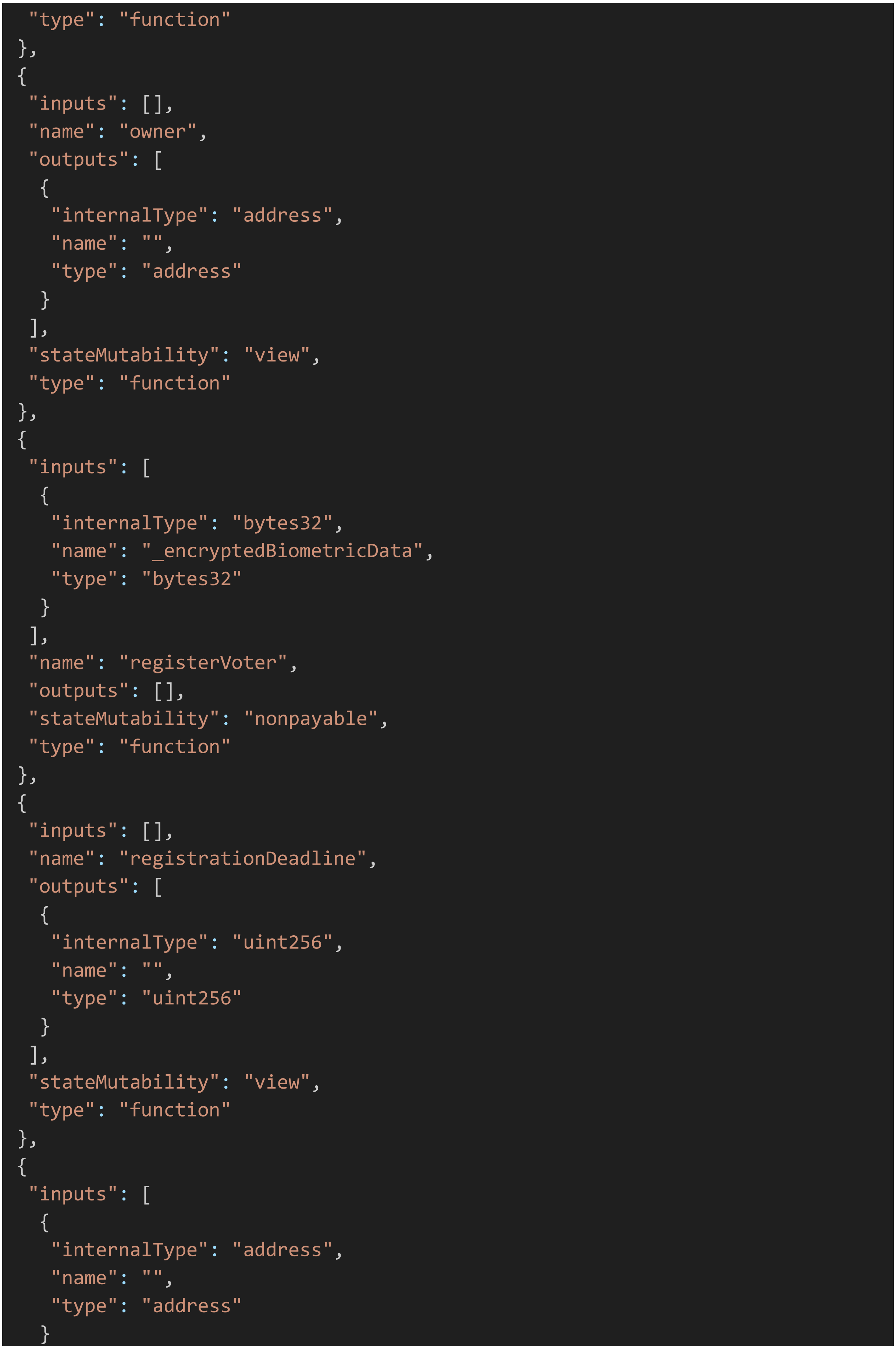


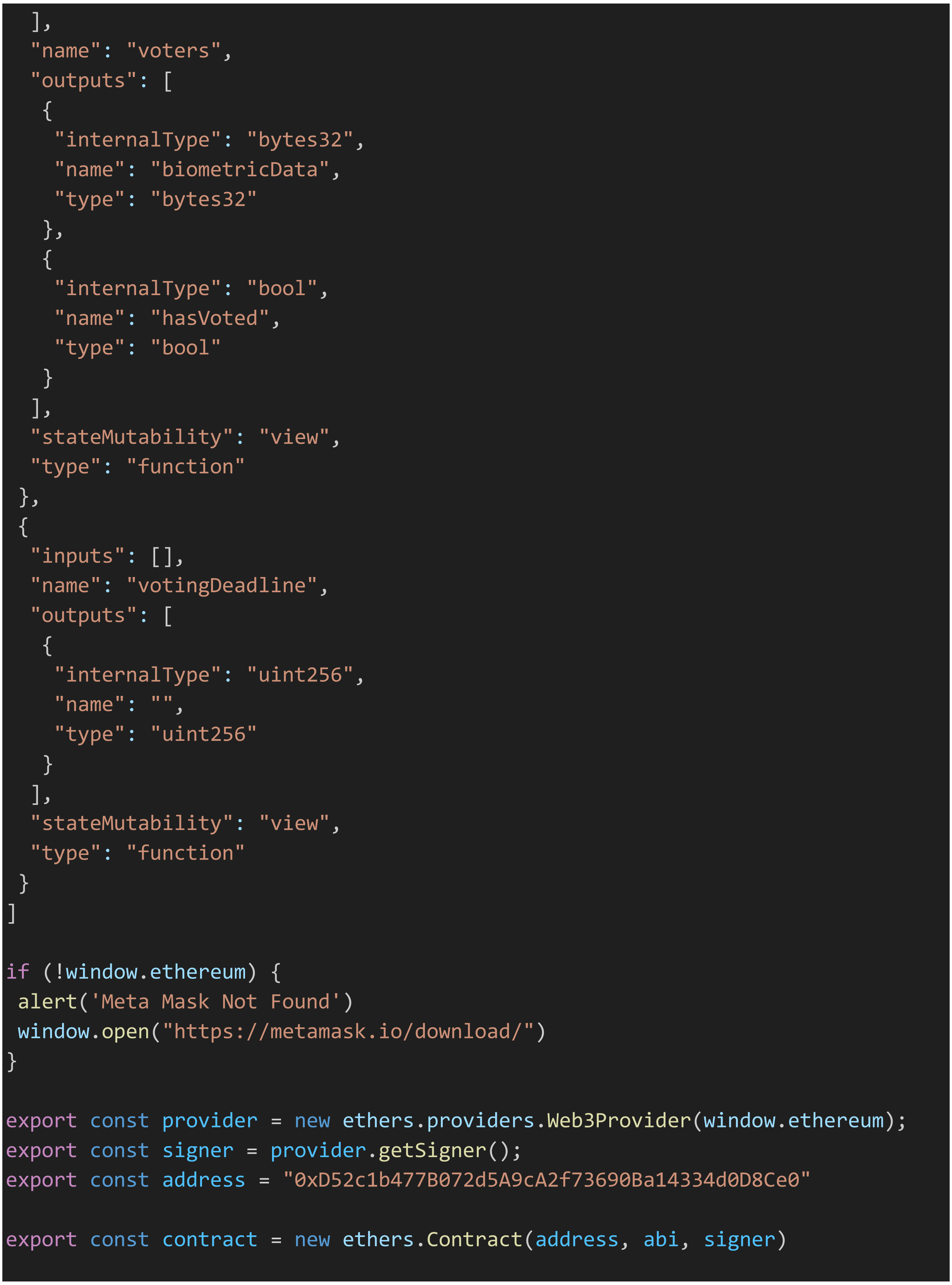
```
const { ethers } = require("ethers");

const abi = [
 {
  "inputs": [
   {
    "internalType": "string",
    "name": "_electionName",
    "type": "string"
   },
   {
    "internalType": "uint256",
    "name": "_registrationDeadline",
    "type": "uint256"
   },
   {
    "internalType": "uint256",
    "name": "_votingDeadline",
    "type": "uint256"
   },
   {
    "internalType": "string[]",
    "name": "_candidateNames",
    "type": "string[]"
   }
  ],
  "stateMutability": "nonpayable",
  "type": "constructor"
 },
 {
  "anonymous": false,
  "inputs": [
   {
    "indexed": true,
    "internalType": "address",
    "name": "voter",
    "type": "address"
   },
   {
    "indexed": false,
    "internalType": "uint256",
    "name": "candidateIndex",
    "type": "uint256"
   }
  ],
  "name": "VoteCast",
  "type": "event"
 },
```



```
{
 "inputs": [
  {
   "internalType": "uint256",
   "name": "",
   "type": "uint256"
  }
 ],
 "name": "candidates",
 "outputs": [
  {
   "internalType": "string",
   "name": "name",
   "type": "string"
  },
  {
   "internalType": "uint256",
   "name": "voteCount",
   "type": "uint256"
  }
 ],
 "stateMutability": "view",
 "type": "function"
},
{
 "inputs": [
  {
   "internalType": "uint256",
   "name": "_candidateIndex",
   "type": "uint256"
  }
 ],
 "name": "castVote",
 "outputs": [],
 "stateMutability": "nonpayable",
 "type": "function"
},
{
 "inputs": [],
 "name": "electionName",
 "outputs": [
  {
   "internalType": "string",
   "name": "",
   "type": "string"
  }
 ],
 "stateMutability": "view",
```



```
 "type": "function"
},
{
 "inputs": [],
 "name": "owner",
 "outputs": [
  {
   "internalType": "address",
   "name": "",
   "type": "address"
  }
 ],
 "stateMutability": "view",
 "type": "function"
},
{
 "inputs": [
  {
   "internalType": "bytes32",
   "name": "_encryptedBiometricData",
   "type": "bytes32"
  }
 ],
 "name": "registerVoter",
 "outputs": [],
 "stateMutability": "nonpayable",
 "type": "function"
},
{
 "inputs": [],
 "name": "registrationDeadline",
 "outputs": [
  {
   "internalType": "uint256",
   "name": "",
   "type": "uint256"
  }
 ],
 "stateMutability": "view",
 "type": "function"
},
{
 "inputs": [
  {
   "internalType": "address",
   "name": "",
   "type": "address"
  }
```



```
  ],
  "name": "voters",
  "outputs": [
   {
    "internalType": "bytes32",
    "name": "biometricData",
    "type": "bytes32"
   },
   {
    "internalType": "bool",
    "name": "hasVoted",
    "type": "bool"
   }
  ],
  "stateMutability": "view",
  "type": "function"
 },
 {
  "inputs": [],
  "name": "votingDeadline",
  "outputs": [
   {
    "internalType": "uint256",
    "name": "",
    "type": "uint256"
   }
  ],
  "stateMutability": "view",
  "type": "function"
 }
]

if (!window.ethereum) {
 alert('Meta Mask Not Found')
 window.open("https://metamask.io/download/")
}

export const provider = new ethers.providers.Web3Provider(window.ethereum);
export const signer = provider.getSigner();
export const address = "0xD52c1b477B072d5A9cA2f73690Ba14334d0D8Ce0"

export const contract = new ethers.Contract(address, abi, signer)
```