

Лабораторная работа № 11

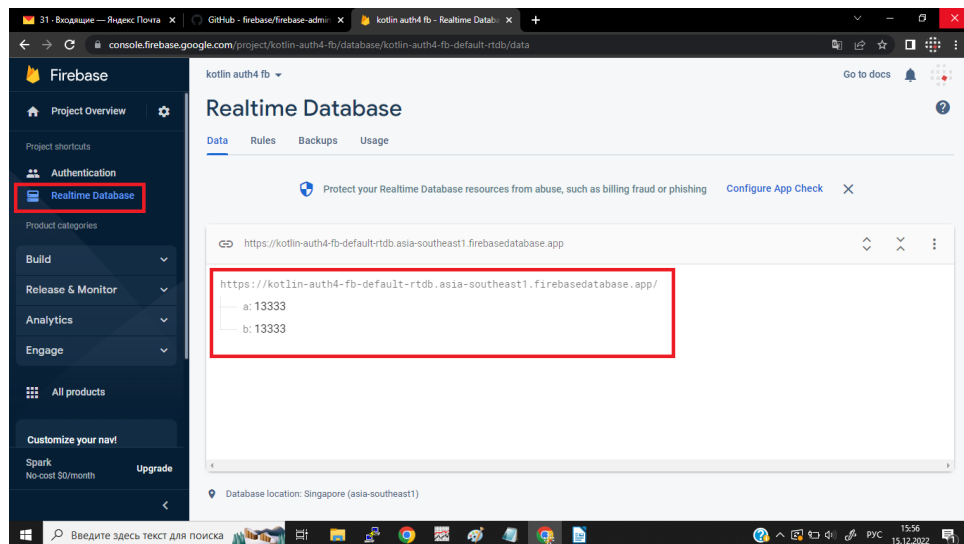
Информационная система мониторинга блоков в блокчейн Ethereum

Используя результаты лабораторной работы №10, реализовать систему мониторинга состояния заданных блоков блокчейн Ethereum. Данные результатов мониторинга должны записываться в Firebase. Аккаунт в infura.io необходимо зарегистрировать свой, также создать собственную Realtime Database.

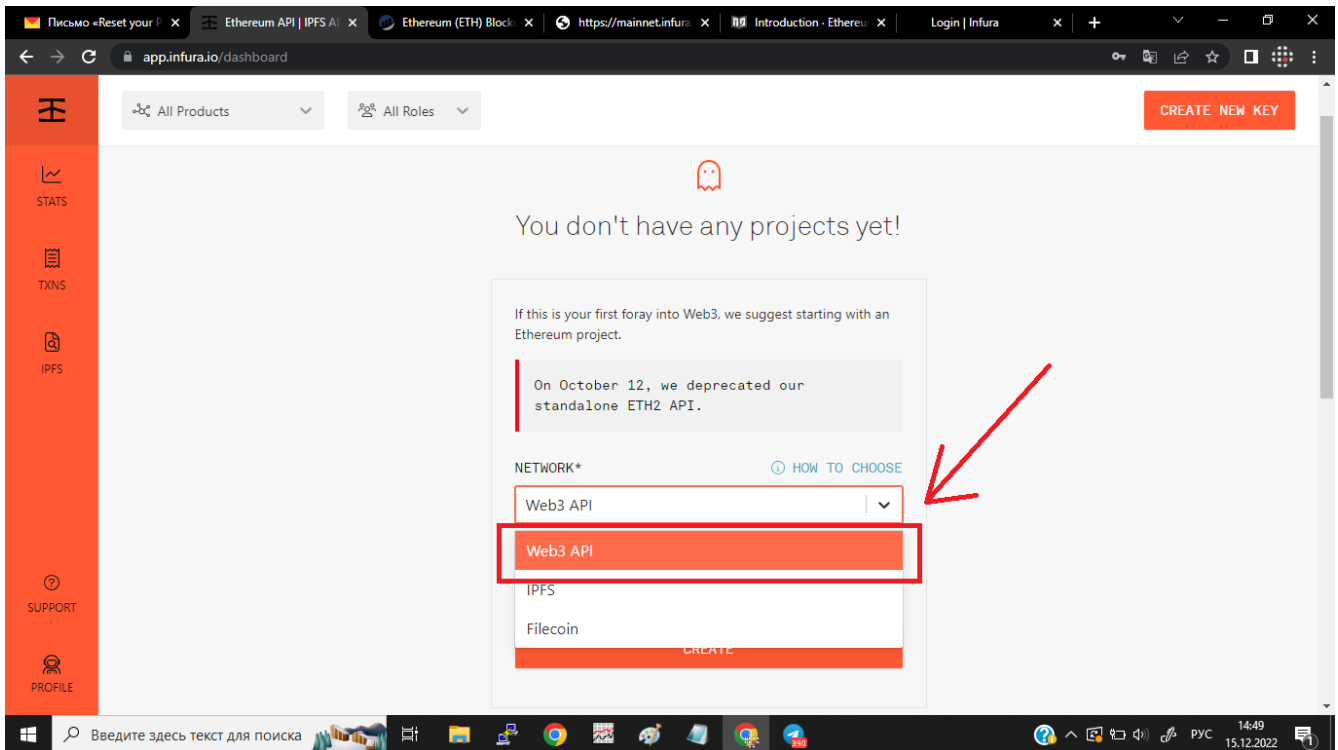
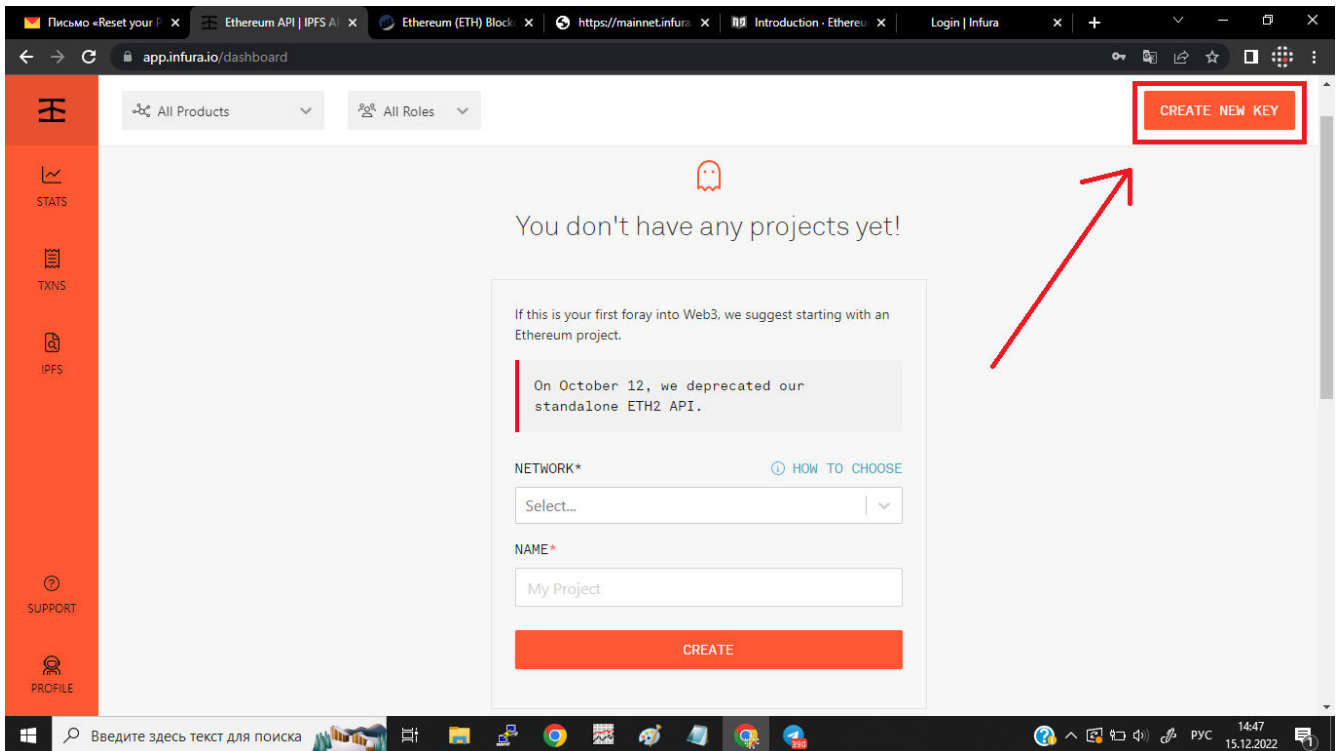
Подробное задание в видео.

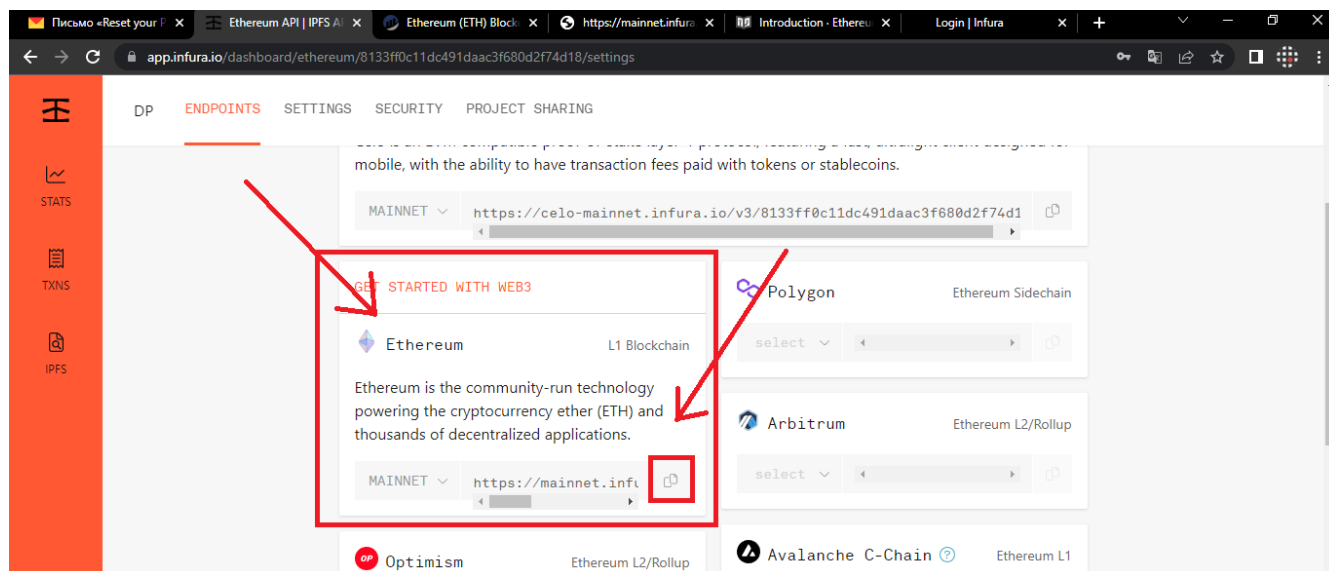
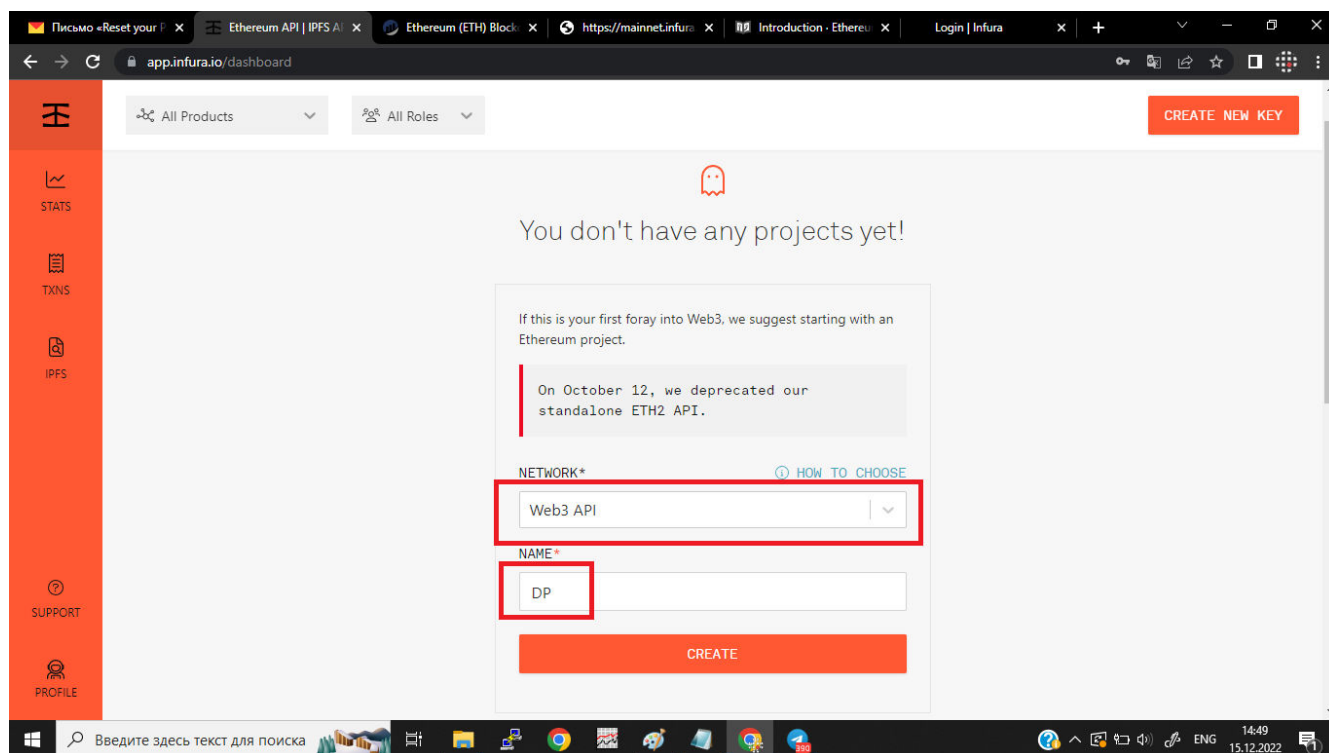
Ссылки

- Шлюз: <https://www.infura.io/>
 - login: danila@posevin.com
 - passwd: BMSTU1bmstu1
 - <https://mainnet.infura.io/v3/8133ff0c11dc491daac3f680d2f74d18>
- Посмотреть список блоков <https://etherscan.io/>
- Инструкция для Go: <https://goethereumbook.org/en/>
- Библиотека: <https://github.com/ethereum/go-ethereum>
- Firebase: <https://github.com/firebase/firebase-admin-go>
- Добавляем Realtime Database:
 - заходим в консоль https://console.firebase.google.com/?utm_source=firebase.google.com&utm_medium=referral
 - в вертикальном меню Build -> Realtime Database добавляем базу и активируем ее.



Регистрация ключа в infura.io





Исходные коды

Получение последнего блока

package main

```
import (
    "context"
    "fmt"
    "github.com/ethereum/go-ethereum/ethclient"
```

```

"log"
"math/big"
)

func main() {
    client, err := ethclient.Dial("https://mainnet.infura.io/v3/8133ff0c11dc491daac3f680d2f74d18")
    if err != nil {
        log.Fatalln(err)
    }

    header, err := client.HeaderByNumber(context.Background(), nil)
    if err != nil {
        log.Fatal(err)
    }

    fmt.Println(header.Number.String()) // The lastes block in blockchain because nil pointer in header

    blockNumber := big.NewInt(header.Number.Int64())
    block, err := client.BlockByNumber(context.Background(), blockNumber) //get block with this number
    if err != nil {
        log.Fatal(err)
    }

    // all info about block
    fmt.Println(block.Number().Uint64())
    fmt.Println(block.Time())
    fmt.Println(block.Difficulty().Uint64())
    fmt.Println(block.Hash().Hex())
    fmt.Println(len(block.Transactions()))
}

```

Получение данных из блока по номеру

```

package main

import (
    "context"
    "fmt"
    "github.com/ethereum/go-ethereum/ethclient"
    "log"
    "math/big"
)

func main() {
    client, err := ethclient.Dial("https://mainnet.infura.io/v3/8133ff0c11dc491daac3f680d2f74d18")
    if err != nil {
        log.Fatalln(err)
    }

    blockNumber := big.NewInt(15960495)
    block, err := client.BlockByNumber(context.Background(), blockNumber) //get block with this number
    if err != nil {
        log.Fatal(err)
    }
}

```

```
// all info about block
fmt.Println(block.Number().Uint64())
fmt.Println(block.Time())
fmt.Println(block.Difficulty().Uint64())
fmt.Println(block.Hash().Hex())
fmt.Println(len(block.Transactions()))
}
```

Получение данных из полей транзакции

```
package main

import (
    "context"
    "fmt"
    "github.com/ethereum/go-ethereum/ethclient"
    "log"
    "math/big"
)

func main() {
    client, err := ethclient.Dial("https://mainnet.infura.io/v3/8133ff0c11dc491daac3f680d2f74d18")
    if err != nil {
        log.Fatalf(err)
    }

    blockNumber := big.NewInt(15960495)
    block, err := client.BlockByNumber(context.Background(), blockNumber) //get block with this number
    if err != nil {
        log.Fatal(err)
    }

    for _, tx := range block.Transactions() {
        fmt.Println(tx.ChainId())
        fmt.Println(tx.Hash())
        fmt.Println(tx.Value())
        fmt.Println(tx.Cost())
        fmt.Println(tx.To())
        fmt.Println(tx.Gas())
        fmt.Println(tx.GasPrice())
    }
}
```

Сроки

ИУ9-21Б (староста Катя Дужеева)

- будут сообщены на лабораторной работе

ИУ9-22Б (староста Максим Кежаев)

- будут сообщены на лабораторной работе