((*D*, *B*, log = (*arg*) => console.log(*arg*)) => {

    const dropZone = *D*.querySelector('div')

    const input = *D*.querySelector('input')

*D*.addEventListener('dragover', (*ev*) => *ev*.preventDefault())

*D*.addEventListener('drop', (*ev*) => *ev*.preventDefault())

    dropZone.addEventListener('drop', (*ev*) => {

*ev*.preventDefault()

        log(*ev*.dataTransfer)

        file = *ev*.dataTransfer.files[0]

        log(file)

        handleFile(file)

    })

    dropZone.addEventListener('click', () => {

        input.click()

        input.addEventListener('change', () => {

            log(input.files)

            file = input.files[0]

            log(file)

            const dateStamp = file.lastModified

            handleFile(file)

        })

    })

    const handleFile = async(*file*) => {

        dropZone.remove()

        input.remove()

        const allowedFiles = ['SD\_workman.xml', 'apbstr.000.txt', 'DataTreeBP.CSV', 'DataTreeSubBP.CSV', 'FFSS.xlsx']

        if (

            allowedFiles.indexOf(*file*.name) == -1

        ) {

*B*.innerHTML = '<h3> Выбранный файл ' + `${*file*.name}` + ' не разрешен к загрузке!</h3>'

            const timer = setTimeout(() => {

                location.reload()

                clearTimeout(timer)

            }, 6000);

            return

        }

        const url = 'http://8-aris-bs:8888/api/uploadfilesfromclient'

        try {

            let reader = new FileReader()

*// reader.readAsArrayBuffer(file)*

            reader.readAsDataURL(*file*)

            reader.onloadend = async function() {

                let content = reader.result;

                const base64String = content;

*// try {*

                const result = await fetch(url, {

                    method: 'POST', *// или 'PUT'*

                    headers: {

                        "Accept": "application/json",

                        "Content-Type": "application/json",

                        "Accept-Encoding": "gzip",

                        "Access-Control-Allow-Origin": "\*",

                        "Content-Transfer-Encoding": "base64"

                    },

                    body: JSON.stringify({

                            content: base64String,

                            name: *file*.name,

                            lastModified: *file*.lastModified,

                            size: *file*.size

                        }) *// данные могут быть 'строкой' или {объектом}!*

                });

                const outtext = await result.text();

*B*.innerHTML = '<h3>' + outtext + '!</h3>'

                const timer = setTimeout(() => {

                    location.reload()

                    clearTimeout(timer)

                }, 5000);

            };

        } catch (error) {

            console.error('Ошибка:', error)

        }

    }

})(document, document.body)