unction resolveSomething(str\_num) {

if (typeof str\_num === "string") {

//sa trimom obrišem white space

let newString = str\_num.trim();

//ispijumeo jeli string manji ili veći od 5

if (newString.length < 5) {

//ako je manni od 5 sve prebacujemo u lower case

return newString.toLowerCase();

} else {

//ako je veci sve prebacujemo u upper case

return newString.toUpperCase();

}

}

//ispitujemo je li paraetar broj, ako je onda ga potenciramo sa 10

// i tražimo znamenke od 2 do 5

else if (typeof str\_num === "number") {

let newNum = Math.pow(str\_num, 10).toString().substring(1, 5);

return parseInt(newNum);

}

//ako je undifined onda ispisemo danasnji datum u obliku 29.6.21

else {

let a = [];

let d = new Date();

//uzmemo zadnje dvije znamenke od 2021, znaci 21

let y = d.getFullYear().toString().substring(2, 4);

a.push(d.getDate(), d.getMonth() + 1, parseInt(y));

return a.join(".");

}

}

console.log(resolveSomething(" fDfd"));

console.log(resolveSomething("tekst"));

console.log(resolveSomething(" tek st "));

console.log(resolveSomething(123));

console.log(resolveSomething());

function short\_Days(date) {

let obj = {

0: "NED",

1: "PON",

2: "UTO",

3: "SRI",

4: "CET",

5: "PET",

6: "SUB",

};

let day = date.getDay();

return obj[day];

/\*for (const key in obj) {

if (day.toString() === key) {

return obj[key];

}

}\*/

}

console.log(short\_Days(new Date()));

console.log(short\_Days(new Date(2015, 10, 1)));

console.log('Exercise 02');

// select all list elements

const e2ListItems = document.querySelectorAll('li');

console.log(e2ListItems)

const isOrderedList = (list) => {

const ordered = list.parentNode.nodeName

if(ordered === "OL") {

return true

}

}; // your condition here

const isUnorderedList = (list) => {

const unOrdered = list.parentNode.nodeName

if(unOrdered === "UL") {

return true

}

}; // your condition here

e2ListItems.forEach(element => {

if (isUnorderedList(element)) {

console.log('Unordered list');

} else if (isOrderedList(element)) {

console.log('Ordered list');

}

});

console.log('Exercise 02');

// select all list elements

const e2ListItems = document.querySelectorAll('li');

console.log(e2ListItems)

const isOrderedList = (list) => {

const ordered = list.parentNode.nodeName

if(ordered === "OL") {

return true

}

}; // your condition here

const isUnorderedList = (list) => {

const unOrdered = list.parentNode.nodeName

if(unOrdered === "UL") {

return true

}

}; // your condition here

e2ListItems.forEach(element => {

if (isUnorderedList(element)) {

console.log('Unordered list');

} else if (isOrderedList(element)) {

console.log('Ordered list');

}

});

console.log('Exercise 07');

document.querySelector(".ex7").addEventListener("click", (event) => {

console.log(event.target.innerText)

})

// let buttons = document.querySelectorAll("button")

// document.querySelectorAll("button").buttons.forEach(button => {

// button.addEventListener("click", function (event) {

// console.log(event.target.innerText)

// })

// });