

# Diario di lavoro

---

Luogo	SAM Trevano
Data	2018-10-12

## Lavori svolti

Ho creato la funzione che cancella tutti i dati degli input. Questa funzione viene richiamata quando si preme il tasto cancella.

```
//metodo che cancella tutti i dati inseriti
function deleteData(){
    if(confirm("Sicuro di voler cancellare tutti i dati?")){
        //numero di input da cancellare meno i 2 bottoni finali
        var inputLength = inputList.length-2;
        for (var i = 0; i < inputLength; i++) {
            inputList[i].value = "";
            inputList[i].style.borderBottom = "0.2vw solid white";
        }
    }
}
```

Metodo che controlla che la data sia corretta dove controllo che la data inserita non sia nulla o futura.

```
//metodo che controlla che sia stata inserita una data corretta
function checkDate(id){
    //salvo i millisecondi della data odierna
    var date = new Date();
    var userDate = new Date(inputList[id].value);

    //controllo che la data non sia del futuro e che il valore non sia nullo
    if(date.getTime()-userDate.getTime()>0 && !isNaN(userDate)) {
        inputList[id].style.borderBottom = "0.2vw solid green";
    }else{
        inputList[id].style.borderBottom = "0.2vw solid red";
    }
}
```

Fonte: [https://www.w3schools.com/js/js\\_date\\_methods.asp](https://www.w3schools.com/js/js_date_methods.asp)

Metodo che controlla che il nap sia corretto. Deve essere un numero di 4 o 5 cifre.

```
//metodo che controlla che sia stata inserita una data corretta
function checkDate(id){
    //salvo i millisecondi della data odierna
    var date = new Date();
    var userDate = new Date(inputList[id].value);

    //controllo che la data non sia del futuro e che il valore non sia nullo
    if(date.getTime()-userDate.getTime()>0 && !isNaN(userDate)) {
        inputList[id].style.borderBottom = "0.2vw solid green";
    }else{
        inputList[id].style.borderBottom = "0.2vw solid red";
    }
}
```

Ho aggiunto i metodi che controllano i campi non obbligatori professione e hobby.

```
//metodo che controllo che il numero di caratteri del campo hobby non sia superiore a 500 caratteri
function checkHobby(id){
    var hobby = inputList[id].value;
    if(hobby.length <= 500){
        inputList[id].style.borderBottom = "0.2vw solid green";
    }else{
        inputList[id].style.borderBottom = "0.2vw solid red";
    }
}

//metodo che controllo che il numero di caratteri del campo profession non sia superiore a 500 caratteri
function checkProfession(id){
    var profession = inputList[id].value;
    if(profession.length <= 500){
        inputList[id].style.borderBottom = "0.2vw solid green";
    }else{
        inputList[id].style.borderBottom = "0.2vw solid red";
    }
}
```

Ho modificato il metodo che controlla il numero civico. Il numero civico deve essere composto

```
//metodo che controlla se il numero civico è valido
function setCivicNumber(id){
    var value = inputList[id].value;
    var regex = /^[a-zA-Z0-9]+$/;
    var civicNumber = regex.test(value);

    if(civicNumber && value.length <= 4 && value.length > 0 && value != "0"){
        inputList[id].style.borderBottom = "0.2vw solid green";
    }else{
        inputList[id].style.borderBottom = "0.2vw solid red";
    }
}
```

Controlli eseguito e testate fino ad adesso:

- Data di nascita
- No. Civico
- Nap
- Hobby
- Professione

Controlli da svolgere:

- nome
- cognome
- indirizzo
- città
- email

Aggiunta dell'icona info a lato degli input.

Fonte: [https://www.w3schools.com/css/tryit.asp?filename=trycss\\_tooltip](https://www.w3schools.com/css/tryit.asp?filename=trycss_tooltip)

**Problemi riscontrati e soluzioni adottate**

-

Terminare controllo campi