

Manipolazione del DOM

1. HTML

```
<body>
  <h1 class="text-center">Rubrica</h1>
  <div class="container">
    <div class="row">
      <div id="listaContatti" class="col-12 d-flex flex-wrap">
        <!-- <div class="rubrica-card">
          <div class="avatar">
            <i class="fas fa-user fa-3x"></i>
          </div>
          <div class="info">
            <span >NOME</span>
            <span>TELEFONO</span>
          </div> -->
        </div>
      </div>
    </div>
  </div>
```

2. Astrazione card JS

```
visualizza: function(){
  console.log('Visualizza');
  let listaContatti = document.querySelector("#listaContatti")
  listaContatti.innerHTML = ""

  this.contatti.forEach((contatto)=>{
    let contattoElement = document.createElement("div")
    contattoElement.classList.add("rubrica-card")
    contattoElement.classList.add("shadow")

    contattoElement.innerHTML = `
      <div class="avatar">
        <i class="fas fa-user fa-3x"></i>
      </div>
      <div class="info">
        <span >${contatto.nome}</span>
        <span>${contatto.telefono}</span>
      </div>
    `

    listaContatti.appendChild(contattoElement)
  })
},
```

3. Funzione Ricerca con *throw* errori e metodo *filter*

```
ricerca: function(nome){
  console.log('Ricerca');
  if(!nome){
    throw `Scrivi un nome`
  }
  let wrapperSearch = document.querySelector("#wrapperSearch")
  wrapperSearch.innerHTML = ""
  let contattoRicerca = this.contatti.find((contatto)=>
    contatto.nome.toLocaleLowerCase().includes(nome.toLowerCase()))

  if(contattoRicerca) {
    wrapperSearch.appendChild(this.creaCardContatto(contattoRicerca, false))
  } else{
    throw `Contatto ${nome} inesistente`
  }
},
```

4. Evento Ricerca con blocco *try-catch* e *timeout*

```
let searchContacts = document.querySelector('#searchContacts')
searchContacts.addEventListener(('click'),()=>{
  let nomeSearch = document.querySelector('#nomeSearch')

  try{
    rubrica.ricerca(nomeSearch.value)
  } catch (error){
    showError(error)
  }
  nomeSearch.value = ""
})

function showError(error){
  let wrapperAlert = document.querySelector('#wrapperAlert')
  let alert = document.createElement('div')
  alert.classList.add("fixed-top", "alert", "alert-danger", "alert-dismissible", "m-4")
  alert.setAttribute("role","alert")
  alert.innerHTML = `
    <span>${error}</span>
    <button type="button" class="btn-close" data-bs-dismiss="alert"></button>`
  wrapperAlert.appendChild(alert)
  setTimeout(() => {
    wrapperAlert.innerHTML = ""
  }, 2000);
}
```