

## IMPOSTARE UN RANGE INPUT

Tramite Bootstrap incolliamo nell'HTML un form **Range Input**.

```
function populatePriceFilter(){
  let input = document.querySelector('#inputPrice')
  let selectedPrice = document.querySelector('#selectedPrice')

  let max = (Math.max.apply(Math, data.map(ad => Number(ad.price))))
  let min = (Math.min.apply(Math, data.map(ad => Number(ad.price))))

  // let max = (data.map(ad => Number(ad.price)).sort((a,b) =>
a-b).pop())
  // let min = (data.map(ad => Number(ad.price)).sort((a,b) =>
a-b).shift())

  document.querySelector('#maxPrice').innerHTML = max + `€`

  input.max = max
  input.min = min
  input.value = max

  selectedPrice.innerHTML = input.value + `€`

  input.addEventListener('input' , () => {
    selectedPrice.innerHTML = input.value = `€` \
  })
}

populatePriceFilter()
```

*Per impostare un range input apriamo una funzione.*

*Acquisiamo tramite il **querySelector** gli ID dell'input (il valore che sceglieremo con il range), e l'ID del selectedPrice (per far vedere all'utente il valore che sta selezionando sul range).*

*Assegniamo ad una variabile il **valore massimo** dei prezzi degli oggetti nel nostro Json, ed una variabile per il minimo (tramite il metodo **map()**).*

*Alla variabile **input** assegnamo i valori minimi e massimi che possono raggiungere.*

*Richiamiamo l'ID dell'elemento nel quale è segnato il valore massimo da mostrare al cliente nel form, e tramite l'**innerHTML** diamo valore uguale alla variabile contenente il valore massimo.*

*Creiamo un **addEventListener** all'input con parametro "input" e diciamo alla funzione che il prezzo selezionato.**innerHTML** sarà uguale all'input.*