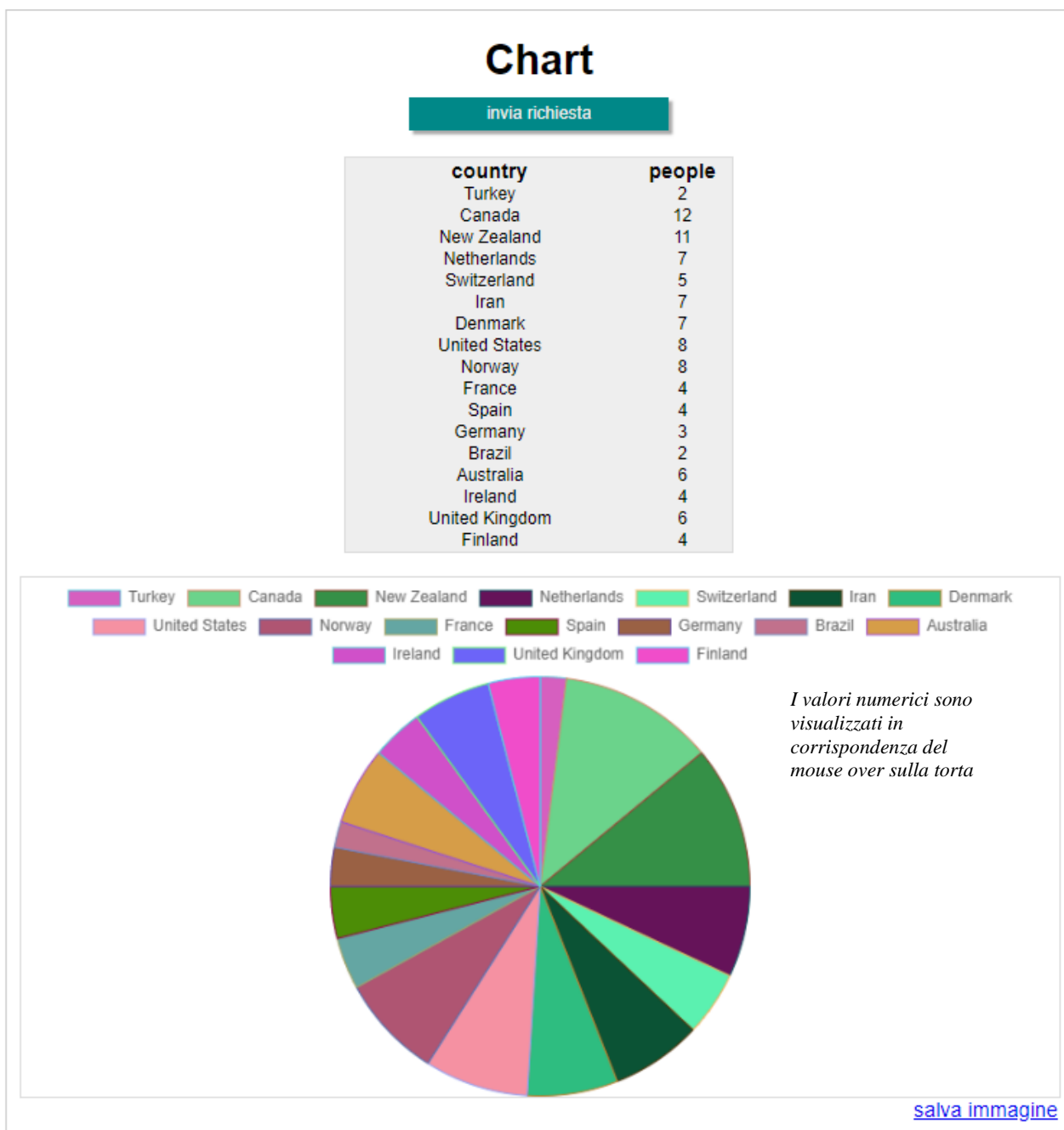


Esercizio 05 : chart js

Scrivere una applicazione che:

- Richieda a randomuser.me un elenco di 100 persone
- Visualizzi sotto forma di tabella (creata dinamicamente) il numero di persone appartenenti a ciascuna nazionalità
- Riporti la stessa informazione sotto forma di grafico
- Aggiunga dinamicamente in coda un pulsante che consenta di salvare l'immagine su disco.



Note

- 1) Attenzione che l'istruzione `new chart()`, per quanto possa sembrare strano, è ASINCRONA, per cui il tracciamento del grafico viene avviato e poi l'elaborazione continua.
Se subito dopo il `new chart()` accedo al canvas lo troverò vuoto.
- 2) Tra una chiamata e l'altra, anche se viene reistaziato, il `chart` mantiene i dati precedenti, per cui prima di istanziare il chart occorre necessariamente 'ripulirlo' con un destroy

```
if(chart)    // != undefined
    chart.destroy();
```
- 3) Nel diagramma `bar`, per fare in modo che l'asse Y parta da 0, si può utilizzare l'opzione indicata nell'esempio base di `chart.js` / `getting started` (e anche sugli appunti di teoria)
Se non viene specificato, le barre sono visualizzate a partire dal valore inferiore.
Abbastanza brutto.
- 4) Per salvare il canvas su disco, all'interno dell'`onClick()` si potrebbe settare l'attributo `href` del tag `<a>` utilizzando il seguente metodo tipico dell'oggetto `chart`:

```
$(this).prop("href", myChart.toBase64Image())
```

dove `myChart` è il nome assegnato all'oggetto `Chart`.