

## Medal Count Widget

The widget is meant to be embedded on client's websites during the Olympic games. The widget shows up to 10 countries that have won the most medals of a given kind in the Olympics games. It has the following requirements:

- It can accept up to two parameters:
  - `element_id` – an id of the element that the module will use as its container.
  - `sort` - determines the type of medals to sort by. It can have one of the following values: 'total', 'gold', 'silver' and 'bronze'.
- If no sort parameter is passed the sort should be by gold medals.
- When ranking by total medals, ties are broken by most gold.
- When ranking by gold, ties are broken by most silver.
- When ranking by silver, ties are broken by most gold.
- When ranking by bronze, ties are broken by most gold.
- Users can click the column headers to re-sort the countries by gold, silver, bronze, or total medals won. With each sort the possibility exists that new countries will enter and depart the top 10. The medals data should not be re-fetched from the server when sort changes.
- Please follow the design shown in the screenshots on the following pages.

A sprite with the 13 countries flags (in alphabetical order according to their country code) can be found here: <https://s3-us-west-2.amazonaws.com/reuters.medals-widget/flags.png>

To get the list of countries and their medals use the following URL: <https://s3-us-west-2.amazonaws.com/reuters.medals-widget/medals.json>. Note that this should be loaded dynamically using an Ajax call (simulate a call to a live API endpoint). Give an appropriate error message if there was an error retrieving the JSON data (if in real production scenario the API is down, you can simulate that by running the widget but turning off your wifi/internet connection). Also note that although this is hosted on S3, a CORS file was set on this folder and thus this data

can be retrieved with a regular Ajax request.

We prefer that the solution will be implemented using react or if you are familiar with ClojureScript you can submit it as ClojureScript code using reagent library as the React wrapper.

If you're using other libraries please explain briefly why you chose them. Your code can be on github (preferred), or emailed back as a zip file containing all the assets needed (html, css, js). If you use a zip file please email it to:

[talgat@gmail.com](mailto:talgat@gmail.com).

The following pages contain screenshots of the widget in different states.

## Medal count table sorted by total gold

Notice that the tiebreak between Netherlands and Germany is handled by total silver.

MEDAL COUNT						
						TOTAL
1		RUS	13	11	9	33
2		NOR	11	5	10	26
3		CAN	10	10	5	25
4		USA	9	7	12	28
5		NED	8	7	9	24
6		GER	8	6	5	19
7		SUI	6	3	2	11
8		BLR	5	0	1	6
9		AUT	4	8	5	17
10		FRA	4	4	7	15

## Medal count table sorted by total medals

Notice that the tiebreak between France and Sweden is handled by the total gold.

MEDAL COUNT					
					TOTAL
1	 <b>RUS</b>	13	11	9	<b>33</b>
2	 <b>USA</b>	9	7	12	<b>28</b>
3	 <b>NOR</b>	11	5	10	<b>26</b>
4	 <b>CAN</b>	10	10	5	<b>25</b>
5	 <b>NED</b>	8	7	9	<b>24</b>
6	 <b>GER</b>	8	6	5	<b>19</b>
7	 <b>AUT</b>	4	8	5	<b>17</b>
8	 <b>FRA</b>	4	4	7	<b>15</b>
9	 <b>SWE</b>	2	7	6	<b>15</b>
10	 <b>SUI</b>	6	3	2	<b>11</b>

## Medal count table sorted by total silver

Notice that the tiebreak between USA, Netherlands and Sweden is handled by the total gold.

MEDAL COUNT						
						TOTAL
1		RUS	13	11	9	33
2		CAN	10	10	5	25
3		AUT	4	8	5	17
4		USA	9	7	12	28
5		NED	8	7	9	24
6		SWE	2	7	6	15
7		GER	8	6	5	19
8		NOR	11	5	10	26
9		FRA	4	4	7	15
10		CHN	3	4	2	9

## Medal count table sorted by total bronze

Notice that the tiebreak between Russia and Netherlands is handled by total gold

MEDAL COUNT					
					TOTAL
1	 USA	9	7	12	28
2	 NOR	11	5	10	26
3	 RUS	13	11	9	33
4	 NED	8	7	9	24
5	 FRA	4	4	7	15
6	 SWE	2	7	6	15
7	 ITA	0	2	6	8
8	 CAN	10	10	5	25
9	 GER	8	6	5	19
10	 AUT	4	8	5	17