

Take Home Test

Your assignment is to implement the **Medal Count App**, defined below, as if you were submitting it for Code Review for Production.

After submission of the test, if we feel it's beneficial to your assessment, we'll provide a Code Review where you will either be asked to make changes or respond with changes you would make based on the feedback.

This test is meant to help us assess your ability to deliver Production Ready code against a brief. We are especially interested to see how you choose to structure your code, your tooling choices, along with your ability to deliver against the brief. Concrete implementations are always our preference; however, we understand that you may not have the time to implement everything as you would prefer. We ask that at minimum you deliver a working solution. Anything you're unable to implement, due to time constraints, we'd recommend you detail in writing. For anything you don't deliver, as much as possible, explain the output you'd be looking to achieve along with your estimate of the time necessary to deliver.

Medal Count Mini App

The mini app is meant to display medal results during the Olympic games. The app shows up to 10 countries that have won the most medals of a given kind in the Olympics games.

Requirements

1. Please time box your development time to 3hrs or less. If you spend extra time planning before development, that's ok.
2. It can accept one parameter from the url:
 - a. **sort**
determines the type of medals to sort by. It can have one of the following values: 'total', 'gold', 'silver' and 'bronze'.
3. If no **sort** parameter is passed the default sort should be gold.
4. Sort Ranking
 - a. When ranking by total medals, ties are broken by most gold.
 - b. When ranking by gold, ties are broken by most silver.
 - c. When ranking by silver, ties are broken by most gold.
 - d. When ranking by bronze, ties are broken by most gold.
5. Users can click the column headers to re-sort the countries by gold, silver, bronze, or total medals won.
6. The medals data should not be re-fetched from the server when sort changes.
7. The flags should be rendered using the **flags.png** file provided.
 - a. Flag icons are in alphabetical order according to their country code

8. The list of countries and their medals use the attached **medals.json** file.
 - a. This should be loaded dynamically using an Ajax call (simulate a call to a live API endpoint).
 - b. Give an appropriate error message if there was an error retrieving the JSON data (if in real production scenario the API is down).
9. Please follow the design shown in the **Examples** section below.
10. Your implementation must be in Typescript & React, and ideally uses Next.js
11. Work must be tracked using Git and shared via a Git platform such as GitHub, GitLab, etc. with public access
12. If you choose to use a starter project or framework such as those listed under "[Creating A React App](#)" please put the bootstrap changes in their own commit(s) and place the rest of the code you implement in one or more separate commits.
 - a. The goal here is we are clearly able to see the code and files you produced vs what was automatically generated.
13. We're especially interested in how you architect the application and structure the code to achieve loose coupling and separation of concerns

Examples

Sort By Gold

Note 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 |

Sort By Total

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

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

Sort By Silver

Note 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 |

Sort By Bronze

Note 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 |