# React - Take Home Challenge

# Introduction:

This programming challenge is not designed to be technically complex, but simple and straight forward. It has been designed to take less than a day, but those experienced in React could complete within a couple of hours. Feel free to take as much time as you need to complete it to the point you’re happy with.

The challenge has 3 levels. First level is for those learning React, second level requires more knowledge of React and Javascript, and 3rd level shows an understanding of supporting technologies. To show your expertise, pay attention to the finer details, use more up to date solutions to the problem, and focus on how you design your components to make them more reusable and testable.

Complete the test yourself without any help from anyone else. It is absolutely fine to use any online resources you may find, as long as you don’t copy and paste code from them. Keep track of technical decisions you make (in your own notes or in commit messages) as we are likely to ask you to explain and defend those decisions.

# Challenge:

Create a simple timezones listing react application. Something similar to <https://www.timeanddate.com/time/zone/australia> but a lot more simplified.

Create a React application using Create React App (<https://create-react-app.dev/docs/getting-started/> ) and complete the following:

* Level 1:
  + Use a public API to get the list of timezones around the world. You can register a free token to use : <https://timezonedb.com/references/list-time-zone> or use other public API accessible.
  + Display the list of timezones in a table (Timezone abbreviation, name and Offset)
  + Each time zone would have a link which when clicked would show the current time in that timezone (HH : MM : SS) in a separate panel.
  + Use Bootstrap to improve the look of the page. Add margins, and paddings and lay elements out in a way that looks clean and professional. You don’t need to spend time on adding design elements.
* Level 2: If you can successfully complete the above
  + Use React Router to create two pages, first page would have the list and second page would display the time for a given timezone.
  + Provide links to go back and forth between the two pages
  + Make sure the links are unique, can be refreshed, and the display page uses timezone as parameter
  + Make the second page display a running time i.e. the time is updated once every second
* Level 3: If you can successfully complete the above:
  + Use Redux or another state management solution to store what timezones have been selected before and in which order and display the selections in a side panel in both pages.
  + Make the listing page also show active time. Each table now shows the current time for all timezones and they get updated once every second at the same time.
  + Store the selections to local storage to make it last when page is refreshed or a new tab is opened.

# Submission:

Push the above to a public repository accessible on github and email us the link.

Pay attention to:

* Commit messages and aim to capture the above dot points in separate commits when possible,
* Ensure your code runs without errors or warnings,
* Code quality and cleanness is important,
* Use any tools and libraries that you find helpful,
* Avoid going overboard or over-engineering the solution,
* Ensure all dependencies required for running the app is captured in package.json. We should only need to install dependencies, and start the app. If anything else is required outline in README.md file,
* Adding some unit test coverage would earn you a good bonus point,