Client-side developer role: pre-interview challenge
Time allowed: 2 hours

Thanks for your interest in applying for the role. We'd like you to complete a coding task before the interview so we can discuss it in person during your first meeting with us.

The goal of the task is for us to get a chance to see your coding approach, learn how you structure a task and see what kind of things you can do when allowed to be creative.

We've set the deadline at **two hours**: we're looking for something you'd be proud to see live on **theguardian.com**, not a tech demo.

The task

We'd like you to create a **JavaScript-based tabbed component** using data from our Content API. You've no doubt seen one before, but just to be clear, here's something similar to what we're aiming for:

Sport	The Guardian

1 The woman who nearly died making your iPad | Aditya Chakrabortty

2 Children given lifelong ban on talking about fracking

3 Teenager Hannah Smith killed herself because of online bullying, says father

4 How to recover from a sexting scandal | Amy Gray

5 McDonald's ties nine out of 10 workers to zero-hours contracts

6 Wayne Rooney prepared to submit transfer request at Manchester United

7 Why is skinny-shaming OK, if fat-shaming is not?

8 Liverpool ready to sell Stewart Downing for around £5m

9 Python escapes from pet shop and strangles two brothers

10 Gareth Bale taken off Tottenham's Twitter account home page

The details

The component will be a listing of **the most recent Guardian content** chosen from **three sections of the site**: UK news, Football and Travel. Each of these sections should be represented by a tab.

We'd like you to fetch this data **asynchronously** from our Content API (the link is below). Please do this on the **client-side** (all solutions should be possible to run in a web browser alone without any server-side requirements).

The content of the tab panes should be a numbered list of **at least five items of content** from that section. The `webTitle` and `trailText` fields should always be displayed, and the `webTitle` should be linked to the `webUrl`. You can use any other fields as you see fit – be creative!

The component should be built to be **mobile-first**: this means it should be **responsive** and attention should be given to keeping it **lightweight** and small in overall filesize.

Although the component will rely on JavaScript to fetch the data, the component itself should be built to **still be visible without JavaScript** (you could include a static version to demonstrate this, if necessary).

The rest of the design and implementation is up to you: feel free to use any third party libraries or tools to help as you see fit, or feel free to write everything from scratch.

We want to see robust code which we can rely upon to work **cross-browser** (from IE9 onwards) and **cross-device**. We're also interested to see what kind of CSS you'll use to produce an attractive and well-designed UI component.

External resources

You'll need a couple of things to get started:

The Guardian's Content API console: http://explorer.content.guardianapis.com/
(use this to explore searches and see the JSON endpoint URL)

The Guardian's style guide: http://pasteup.guim.co.uk/zones.html – optional
(don't feel you have to use it – if you need logos or section colours, it may help)

What to do when you've finished

After two hours, stop at a logical place and push your changes to a public Github repo. If you are unfamiliar with Github, **ZIP your project up (this is important)** and email it to the email supplied when receiving this test.

Good luck, and come prepared to talk about your code at the interview (we'll bring along a laptop with your work, don't worry).