|  |  |
| --- | --- |
|  | Java exam |

# Overview

For this exam you will be building a backend for a new feature of the Discount Ascii Warehouse ecommerce platform.

Your application will produce a list of "Popular purchases", so customers can see who else bought the same products as them. To complete the exam your application will need to accept HTTP requests to /api/recent\_purchases/:username and respond with a list of recently purchased products, and the names of other users who recently purchased them.

There is no frontend component to this exam, you're just building the backend.

# Other requirements

* your application must cache API requests so that it can respond as quickly as possible.
* if a username is provided that cannot be found, the API should respond with "User with username of '{{username}}' was not found"

# Where does the data come from?

Data about users, products and purchases is available to you via an API you can set up and host locally: <https://github.com/x-team/daw-purchases/blob/master/README.md#api-reference>

To work out the "Popular purchases":

* fetch 5 recent purchases for the user: GET /api/purchases/by\_user/:username?limit=5
* for each of those products, get a list of all people who previously purchased that product: GET /api/purchases/by\_product/:product\_id
* at the same time, request info about the products: GET /api/products/:product\_id
* finally, put all of the data together and sort it so that the product with the highest number of recent purchases is first.

Example response:

[  
 {  
 *"id": 555622,  
 "face": "｡◕‿◕｡",  
 "price": 1100,  
 "size": 27*,

"recent": [  
 "Frannie79",  
 "Barney\_Bins18",  
 "Hortense6",  
 "Melvina84"  
 ]  
 },  
 ...  
]

# What to include in your solution

Your solution should be a zipped (or gzipped) archive including:

* all source code required to run the application
* step-by-step instructions for the reviewer to follow so that they can run your application.
* any other notes or rationale you think is helpful for the reviewer to consider when grading your solution.