

Developer interview task (Mobile)

Let's say that for our annual fundraiser we want to use currency conversion rates that are periodically updated automatically rather than having to constantly update them by hand. In order to do this, we sign up for a 3rd party service that provides us with daily conversion rates for the currencies that we support. The service is a simple API that outputs XML when called with the URL <http://toolserver.org/~kaldari/rates.xml>.

First, define a MySQL table that can store this data. You don't actually have to set up the table anywhere, just create an SQL file that contains the CREATE statement for the table.

Next, construct a PHP class that can handle all of the following tasks:

- Retrieving the data from the API
- Parsing the data
- Storing the data in your MySQL table
- Given an amount of a foreign currency, convert it into the equivalent in US dollars. For example:

input: 'JPY 5000'
output: 'USD 65.58'

- Given an array of amounts in foreign currencies, return an array of US equivalent amounts in the same order. For example:

input: array('JPY 5000', 'CZK 62.5')
output: array('USD 65.58', 'USD 3.27')
(This can be a separate function from #4.)

Finally create a mobile friendly landing page for iOS and Android to show donation amounts and currency options.

- The landing page should show a fixed amount of payment options (\$10, \$25, \$50, \$100) styled so they look like buttons.
- The landing page should allow the user to switch between different currencies. When a currency is selected the buttons should update with the new amounts.
 - You do not have to implement the donor workflow after the landing page but when the user selects a donation amount you should give an indication of a statement of intent e.g. "Make payment of \$25 USD!" or redirect to a page that 404s for example `/payment?value=20¤cy=usd`

Your final submission should include:

- a PHP file that defines the 'CurrencyConverter' class
- an SQL file that creates the 'exchange_rates' MySQL table
- Any JS/HTML/CSS to render the user facing page
- A small critic of your solution detailing what you would have done with more time and/or what you do not like about your current implementation.

Code should be checked into github for review