

TASK

Create a currency calculator using the given code mock-up. The calculator must convert the sum in EUR into another currency. Currency exchange rates have to be loaded from the given URL.

The attached script is a mock-up of the final code. It separates business logic from presentation:

- main logic (accessing currency exchange rate file, conversion, preparation of output data) should be carried out in index.php
- view.php is loaded as the last instruction of index.php
- presentation logic (HTML code + little analysis of variable values and output code) should happen in "view.php"

Main tasks that have to be accomplished:

- analyze user input data, perform conversion only if all mandatory fields are filled and data corresponds to the expected type:
 - the amount can be recognized as a decimal value (without any letters, not empty),
 - the date corresponds to the date type and is greater than 1999-01-04 and less than or equal to the current date (date of request),
 - the currency code is correct and is present in \$currencies array,
 - shows an error message in case of any input validation errors;
- download the currency exchange rate file that corresponds to the selected currency from the remote server on the first calculation request (don't forget that the service may be unavailable, prepare a specific message for this case!);

- after initial download, save the currency exchange rate document in the xml/ directory under the name .xml (make sure directory permissions allow PHP to write there, Linux: `chmod a+w xml/`);
- the rates file contains rates only for working days, therefore, if the entered date is on weekend, use the closest previous working day instead;
- reuse the saved file for future requests but only if the file contains the date entered by the user (check "TIME_PERIOD" attribute values inside an XML document and compare with the entered date) otherwise download again;
- parse currency exchange rate document in order to find the needed currency rate that corresponds to the entered date;
- do calculation and round the result to 2 decimal places.

The variables that can be passed from index.php into view.php, are:

- \$result (empty string, "OK", "ERROR"): depending on the value, the calculation result or error message is shown on screen;
- \$result_message: a string to be displayed to the user
- either error message or success explanation;
- \$date: the date on which the conversion should be performed.

Currency exchange rates for the particular currency must be obtained from the URL

https://www.ecb.europa.eu/stats/policy_and_exchange_rates/euro_reference_exchange_rates/html/.xml where must be substituted with the corresponding currency code. For downloading the currency rates, you

can use the `file_get_contents` function in PHP (if this does not work, check the "allow_url_fopen" switch in `php.ini` file). The simplest XML data extractor is SimpleXML library, available in standard PHP bundle.

Example:

Suppose a user wants to convert 100 U.S. Dollars into Euro using the exchange rate on 7 March 2021. To make the conversion, your application must

- 1) check whether all data are entered correctly;
- 2) since 7 March 2021 is Sunday, use 5 March 2021 instead;
- 3) check whether a file with the name `usd.xml` is present in the folder `xml/` and whether this file contains a rate on 5 March 2021 (otherwise, the application must download the XML file from the URL https://www.ecb.europa.eu/stats/policy_and_exchange_rates/euro_reference_exchange_rates/html/usd.xml, save the downloaded file in the folder `xml/` and use it for conversion);
- 4) find the required currency exchange rate in the file with `TIME_PERIOD="2021-03-05"`, perform conversion and round the result to 2 decimal places: $100/1.1938=83.77$.