Part A – PHP Response Objects

In the files folder is a file called albums.json. It is the albums.xml file you used in the first assignment but in json format. Write a php program called partajson.php which will receive, in the query string, a single name value pair of genre=*something* where *something* is one of the valid genres in the file (the same list as assignment 1).

When the php program is called, it must retrieve the passed genre. It then reads the JSON file in, creates a set of json objects (json\_decode), goes through the objects to find the records matching the genre. The php file returns a content type of json (application/json), the length of the total matching json objects being returned and the json objects themselves. Use getAlbums.php from assignment 1 and the example on Moodle to help you.

Create a simple html file which allows the user to select a genre (no you do not have to read the genre list in from the file like in the assignment, but you can if you like). It then calls the partajson.php file which will return the records matching that genre. Display the records on the screen when they are returned. Note, you do NOT have to make the output pretty in this case. A simple display of the json objects is fine. NOTE: A display of a string is NOT fine.

Part B – PHP Classes and Objects

Write a class-based all-in-one web form called partb.php that calculates the correct amount of change to return when performing a cash transaction. The class Change should be a separate file called Change.php that is required by the all-in-one form.

The page allows the user (a cashier) to enter the cost of a transaction and the exact amount of money that the customer hands over to pay for the transaction. Use (public) set and get functions in the Change class to store and retrieve amounts from private attributes within the class. Add function(s) to the class to determine the largest number of each denomination to return to the customer.

Assume that the denominations are bills at $50, $20, $10, $5 and coins at twonies, loonies, quarters, dimes and nickels (these would be the attributes of the Change class).

For example, the user enters $10.65 as the amount and $50.15 as the money received. The difference is $39.50. Have a function numberBack in the class which receives two parameters, the amount and the bill/coin being checked.

* Call numberBack with $39.50 and $50, function returns 0 (no $50 bills in change)
* Set number of fifties in Change class to 0 (if not already initialized as such)
* Call numberBack with $39.50 and $20, function returns 1 (1 $20 bill in change)
* Set number of twenties in Change class to 1
* Call numberBack with $19.50 and $10, function returns 1 (1 $10 bill in change)
* …continue until amount reaches 0.
* In the end, the change class should have 0 $100, 0 $50, 1 $20, 1 $10, 1 $5, 2 $2, 0 $1, 2 quarters, 0 dimes and 0 nickels

The result would be displayed in the form and the user can clear the form or enter new amounts. Once you get it working with multiples of 5 cents, do it to the penny and round to the nearest nickel. So change of $3.08 would be $3.10 and change of $4.62 would be $4.60

Test numbers:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Amount | Tendered | Change   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 50 | 20 | 10 | 5 | 2 | 1 | .25 | .10 | .05 | |
| 4.65 | 10.00 | |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | |
| 15 | 15.00 | |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 3.20 | 50.00 | |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 0 | 2 | 0 | 1 | 0 | 0 | 3 | 0 | 1 | |
| 31.90 | 100.00 | |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 0 | 1 | 1 | 2 | 0 | 0 | 1 | 0 | |
| 6.60 | 10.00 | |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | |
| 112.00 | 200.00 | |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | |
| 22.70 | 85.00 | |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | |
| 3.56 | 5.00 | |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 2 | 0 | |
| 17.84 | 50.00 | |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | |