### Methods Defined on purchaseProducts.php

| Method Name | Description | Input Parameters apart from security credentials | Output On Success | Failure Condition | Output on Failure |
|---|---|---|---|---|---|
| **createCart** | This method is used to create a cart for the user. At a time only one cart exists for a user. The user cannot create a new cart until that user checkout the previous cart. | No parameters required | Returns the message that cart has been created successfully with the corresponding cart Id. | when an unused cart already exists for that particular user | Returns the message- the cart already exists with that corresponding cart Id. |

| Method Name | Description | Input Parameters apart from security credentials | Output On Success | Failure Condition | Output on Failure |
|---|---|---|---|---|---|
| **checkOutCart** | This method checkouts the cart updating the inventory count on product database | No parameters required | Returns the success message “checkout completed” with the total amount. | When there is no cart in process i.e. the cart with products added to it. | Returns the failure message “No cart in progress is found”. |

| Method Name | Description | Input Parameters apart from security credentials | Output On Success | Failure Condition | Output on Failure |
|---|---|---|---|---|---|
| **addProductsToCart** | This method is used to add products to the cart.<br>. | **productList** –<br>A Json encoded associative array containing **Product title** as index and **quantity** needed as value. | Returns the message products added to cart, The total price and the products that have been added to cart. | when there is no existing cart is found for that user | Returns the error message “No empty cart is found create a cart to add products”. |
| | | | | When the **productList** parameter is invalid or empty | Returns the error message Invalid Product List format. |
| | | | | When the requested products are completely out of stock or not available in the system | Returns the message “All the requested items are not available at the moment” |
| | | | | When the **productList** parameter consists of products that are in any combination of available, not available and no inventory | Returns the message “Some or all of the requested products are not added to cart due to no stock or unavailability at the |

|  |  |  |  |  |  |
|---|---|---|---|---|---|
|  |  |  |  |  | moment”. Returns the list of products that are added and not added to the cart. |

| Method Name | Description | Input Parameters apart from security credentials | Output On Success | Failure Condition | Output on Failure |
|---|---|---|---|---|---|
| **emptyCart** | This method is used to clear the products added to cart | No parameter required | Returns the success message “cart emptied successfully” | When there is no cart in process i.e. the cart with products added to it | Returns the error message “No cart currently in process of checkout to empty it”. |

| Method Name | Description | Input Parameters apart from security credentials | Output On Success | Failure Condition | Output on Failure |
|---|---|---|---|---|---|
| **authenticate** | This method is used to validate the security tokens sent by the user for executing any functions defined in this file. | The security tokens sent by the user | Returns true on success | When the security credentials are invalid. | Returns Authentication failed message to the user. |

**General Comments** -

1. All outputs are json encoded
2. When sending a json encoded string, please do url encode the json encoded string before sending it as request
3. Only POST method is allowed for functions defined in this file.
4. The security tokens are ‘userName’,’apiKeyId’,’requestSalt’ and requestToken. How it is obtained and who has what security token is described in authentication document.
5. All the security tokens should be sent through post request body for the calls accessing the methods described in this file.
6. The method name should be assigned as a value to action parameter for sending a request.
7. The authenticate function will be called by the script itself. The user will not be calling this function.
8. Calling the addProductsToCart function again and again simply edit the cart details with the ones specified in productList parameter until the request is valid otherwise last successful operation of adding products to cart will be maintained.
9. A sample access URL for the functions defined in this file http://localhost/api/purchaseProducts.php
10. The code for getting system salt and forming a request token can be found in authentication.php file.
11. Now, the code for creating a cart is as follows.

```
        $url = "http://localhost/api/purchaseProducts.php";
         $post =
array('action'=>'createCart','userName'=>urlencode($userName),'apiKeyId'=>urlencode($apiKeyId),’requestSalt'=>urlencode($requestSalt),'requestToken'=>urlencode($requestToken));
          $curl = curl_init($url);
        curl_setopt($curl, CURLOPT_TIMEOUT, 3);
        curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
        curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, 0);
        curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, 0);
        curl_setopt($curl, CURLOPT_POST, 1);
         curl_setopt($curl, CURLOPT_POSTFIELDS, $post);
        $resp = curl_exec($curl);
        $httpCode = curl_getinfo($curl, CURLINFO_HTTP_CODE);
         echo $resp;
        echo $httpCode;
        if ($httpCode == 200) {
                $array = json_decode($resp,true);
                 echo print_r($array,true);
         }
        curl_close($curl);
```

12. Now to check out or empty the cart, it is the same code above just change the action method to corresponding method.
13. To add Products to code, the code will be as follows.

```
        $productList = urlencode(json_encode(array("LG P880 4X HD"=>2,"Google Nexus 4"=>2,"Bench Shirt"=>1)));
        $url = "http://localhost/api/purchaseProducts.php";
         $post=
array('action'=>'addProductsToCart','productList'=>$productList,'userName'=>$userName,'apiKeyId'=>$apiKeyId,'requestSalt'=>urlencode($requestSalt),'requestToken'=>urlencode($requestToken));
        $curl = curl_init($url);
         curl_setopt($curl, CURLOPT_TIMEOUT, 3);
         curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
        curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, 0);
```

```
 curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt($curl, CURLOPT_POST, 1);
 curl_setopt($curl, CURLOPT_POSTFIELDS, $post);
$resp = curl_exec($curl);
$httpCode = curl_getinfo($curl, CURLINFO_HTTP_CODE);
echo $resp;
echo $httpCode;
if ($httpCode == 200) {
               $array = json_decode($resp,true);
              echo print_r($array,true);
 }
curl_close($curl);
```