

Functionalities defined in getProductAndCartDetails.php

Method Name	Description	Input parameters apart from credential tokens	Output on Success	Failure Condition	Output on Failure
getAllProducts	This method returns either all the products or products having available inventory.	onlyAvailableInventory	Returns all products with available inventory with the message products obtained successfully.	when there is no product with available inventory.	Return message - "no products found at this moment"
		No Argument passed	Returns all products irrespective of inventory count.	Occurs when there is no product in system	Return Message - "no products found at this moment"

Method Name	Description	Input parameters apart from credential tokens	Output on Success	Failure Condition	Output on Failure
getProductsByTitle	This method returns all the products that matches the product title(s) given	productTitles. A Json encoded array containing Product titles as values.	Returns the product details matching the Titles given .	When there are no products matching the titles in productTitles argument.	No Products Found by the requested title(s).
				When the productTitles parameter consists of mixed valid and invalid product titles	Returns the product details of valid product titles and returns an json encoded string of invalid product Titles
				When the productTitles parameter is empty or In invalid format	Returns the error message - productTitles parameter is empty or improper.

Method Name	Description	Input parameters apart from credential tokens	Output on Success	Failure Condition	Output on Failure
getProductsById	This method returns all the products that matches the product Id(s) given	Required Parameter - productIds . A Json encoded array containing Product Ids as values.	Returns the product details matching the Ids given.	Occurs when there are no products matching the titles in productIds argument	No Products Found by the requested Id(s).
				When the productIds parameter consists of mixed valid and invalid product Ids	Returns the product details of valid product titles and returns an json encoded string of invalid product Ids
				When the productIds parameter is empty or In invalid format	Returns the error message - productIds parameter is empty or improper.

Method Name	Description	Input parameters apart from credential tokens	Output on Success	Failure Condition	Output on Failure
getCartDetails	This method returns the cart details of a particular user	cartId	Return the cart details of respective cart Id only if it belongs to that user	Cart Id is invalid.	Returns the message cart Id is invalid
		cartStatus Acceptable cart status (1) empty (2) InProcess (3)checkOutCompleted	Return the details of all carts matching the cart status of that particular user	When there is no records matching the cart status for that user.	Returns message- No cart found matching the given cart status
		No Parameter passed	Returns all the details of all the carts of that particular user	When there is no cart found for that user	Returns – “No cart found for this user” message.

Method Name	Description	Input parameters apart from credential tokens	Output on Success	Failure Condition	Output on Failure
authenticate	<p>This method is used to validate the security tokens sent by the user for executing any functions defined in this file</p> <p>Note – This function will be called by the script itself. The user will not be calling this function</p>	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 GET 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 http headers 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. A sample access URL for the functions defined in this file
<http://localhost/api/getProductAndCartDetails.php?action=getAllProducts&OnlyAvailableInventory>.

8. The code for obtaining salt and forming the security token is described in authentication.php
9. Now, the code to get all the products without available inventory constraint is as follows.

```
$url = "http://localhost/api/getProductAndCartDetails.php?action=getAllProducts";
$curl = curl_init($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($curl, CURLOPT_HTTPHEADER,
array('userName:'.$username,'apiKeyId:'.$apiKeyId,'requestSalt':urlencode($requestSalt),'requestToken':urlencode($requestToken)));
$response = curl_exec($curl);
$httpCode = curl_getinfo($curl, CURLINFO_HTTP_CODE);
$array = json_decode($response,true);
if($httpCode==200) {
    echo print_r(json_decode($array['productDetails'],true),true);
}
curl_close($curl);
```

10. Now, the code for getting all products that has available inventory is just same as above with the following change in URL

```
$url = "http://localhost/api/getProductAndCartDetails.php?action=getAllProducts&onlyAvailableInventory";
```

11. Now, the code for getting products by title is as follows.

```
$productTitles = array('LG P880 4X HD','Google Nexus 4','Samsung Galaxy S4');
$productTitles = urlencode(json_encode($productTitles));
$url = "http://localhost/api/getProductAndCartDetails.php?action=getProductsByTitle&productTitles=".$productTitles;
$curl = curl_init($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($curl, CURLOPT_HTTPHEADER,
array('userName:'.$username,'apiKeyId:'.$apiKeyId,'requestSalt':urlencode($requestSalt),'requestToken':urlencode($requestToken)));
$response = curl_exec($curl);
```

```
$httpCode = curl_getinfo($curl, CURLINFO_HTTP_CODE);  
$array = json_decode($resp,true);  
if($httpCode==200) {  
    echo print_r(json_decode($array['productDetails'],true),true);  
}  
curl_close($curl);
```

12. Now the code for getting products by id is same as above except for the following modifications.

```
$productIds = array('1','2','3');  
$productIds = urlencode(json_encode($productIds));  
$url = "http://localhost/api/getProductAndCartDetails.php?action=getProductsById&productIds=".$productIds;
```

13. Now, the url for getting all cart details of the user is as follows.

```
$url = "http://localhost/api/getProductAndCartDetails.php?action=getCartDetails";
```

14. Now, the url for getting cart details of the user by cart Id is as follows.

```
$url = "http://localhost/api/getProductAndCartDetails.php?action=getCartDetails&cartId=1";
```

15. Now, the url for getting all cart details of the user matching a cart status is as follows.

```
$url = "http://localhost/api/getProductAndCartDetails.php?action=getCartDetails&cartStatus=checkOutCompleted";
```