**Practical – 5**

**Aim: -** Create PHP code that utilizes the commonly used API library functions built in to PHP.

**API: - Application Programming Interface (API)** is a software interface that allows two applications to interact with each other without any user intervention. API is a collection of software functions and procedures. In simple terms, API means a software code that can be accessed or executed. API is defined as a code that helps two different software’s to communicate and exchange data with each other. It is very much similar to a web service used in developing web pages or mobile apps. One application can call other programs API to utilize some functionality. APIs get requests and return the result in the programmer’s software system. If the system communicates with databases, then the APIs are exposed by PHP extensions.

Examples of API are: -

1. Google Maps API
2. YouTube API

**Prerequisites: -**

1. PHP
2. PHP cURL Library

<?php

$url = 'RequiredLink';

$data = [ 'collection' => 'RequiredAPI' ];

$curl = curl\_init($url);

function APIcall($method, $url, $data)

{

        $curl = curl\_init();

            switch ($method)

                {

                    case "POST": curl\_setopt($curl, CURLOPT\_POST, 1);

                        if ($data) curl\_setopt($curl, CURLOPT\_POSTFIELDS, $data);

                        break;

                    case "PUT": curl\_setopt($curl, CURLOPT\_CUSTOMREQUEST, "PUT");

                        if ($data) curl\_setopt($curl, CURLOPT\_POSTFIELDS, $data);

                        break;

                }

curl\_setopt($curl, CURLOPT\_URL, $url);

curl\_setopt($curl, CURLOPT\_HTTPHEADER, array( 'APIKEY: RegisteredAPIkey', 'Content-Type: application/json', ));

curl\_setopt($curl, CURLOPT\_RETURNTRANSFER, 1);

curl\_setopt($curl, CURLOPT\_HTTPAUTH, CURLAUTH\_BASIC);

$result = curl\_exec($curl);

if(!$result)

    {

                 echo("Connection failure!");

    }

curl\_close($curl); return $result; }

?>

**Types of Web APIs:** Web APIs are those which are accessible over the internet.

1. **Open APIs:** These APIs are publicly available as there is no restrictions.
2. **Partner APIs:** The user needs license and special rights to access this type of APIs.
3. **Private APIs:** Owned by companies for internal systems.
4. **Composite APIs:** Its a combination of data and service APIs for speeding up the execution process.

Other than the above four, many other APIs are also available. Some APIs are available on the internet, you need not install the software.

**Types of Web Service APIs:** A very common example, use of payment process API rather than developing our own payment process. Web service APIs are platform-independent methods that are accessed by a network connection.

1. **SOAP:** Simple Object Access Protocol which uses web service definition language or XML for data transfer. It is very robust. These are used in integrating APIs.
2. **JSON-RPC:** For data transfer it uses JSON.
3. **REST:** The set of rules includes some standard architectural principles for data exchange. For making a request, it uses HTTP methods of getting, PUT, POST, PATCH, DELETE for all CRUD operations. It consumes less bandwidths and also comfortable accessing cloud services.

**REST output is in the form of JSON:**

* GET: Read or retrieves information.
* POST: Creates new record.
* PUT: Update a record.
* DELETE: Deletes a record.

Whenever an application uses all four database operations for create, read, update, delete. It is said to have utilized the REST API. All APIs are not web services but all the web services are APIs. A very common example is the REST APIs. REST APIs are backbones of the internet and web services. The .htaccess file is used for mapping the request URI to the REST API service.

<?php

require\_once("MobileRestHandler.php");

$mobileRestHandler = new MobileRestHandler();

$mobileRestHandler->getAllMobiles();

?>

* **GET:** Fetch information or data collection. For example product details from a table.
* **POST:** Adds or creates new information such as feedback or reviews of some restaurant.
* **PUT:** Updates existing data.
* **DELETE:** Deletes existing data.

**Using API with PHP cURL:**

1. **Get an API key:** When you start with APIs, you should register and get a key which is a string of letters and numbers. This API key is only used while making a request so that the server API recognizes the user as a registered user.
2. **Test API with PHP applications:** This process checks if everything works as expected.
3. Develop the required PHP app by using API. Create or develop the application by using necessary APIs.

**Steps to create REST API:**

1. Create a database and DB table.
2. Establish Database Connection.
3. Create a REST API file
4. Create index or HTML file
5. Fetch the records from Database via cURL