

# How to Use ProProfs API to Capture Learner Data for a Website

Using our API, you can capture quiz data on your website or web-server with ease.

## Example Use Cases of Capture API

- Inserting lead capture into your own database such as name, email, etc. of quiz taker
- Inserting names and information for people that take the test into SugarCRM, Salesforce, or any other CRM system
- Triggering emails or any other process on your website when someone takes the test

This feature requires minor programming at your end to accept the data sent by ProProfs. There are two ways you can capture data through API.

## How it works

Your callback script will be called whenever someone [takes your quiz](#), which will accept the data. After accepting the data, you can write any custom code, such as inserting it into your database or CRM or triggering some business logic process.

## Please follow the steps below to set up the callback URL:

Go to the **Settings** page and click on **Advanced**. Enable "**Notification via API**." Type the callback URL indicating where the script is on your site. This is the URL of the script which will accept the data in the REQUEST method. Select the "**Query String/REQUEST**" or "**JSON**" option from the drop-down menu.

### Notification via API:

Enable: ☒ Yes ☐ No Send lead capture & quiz attempt data to my website via API. [?](#)

Link:

For example: <http://www.xyz.com/abc.php>

Capture Method:

Query String / REQUEST ▾  
Query String / REQUEST  
JSON

Notification via TinCan API:

Enable: ☐ Yes ☒ No Send lead capture & progress data to my LRS via API. [?](#)

This integration can be done either through the **REQUEST** or **JSON methods**.

**Examples for both have been given below. Both methods use the same variable and variable attributes. The variables have been given in the table.**

#	Variable Name	Variable Description	Data Type	Example (PHP)
1	result_id	A unique ID to identify an attempt	INT	<code>\$_REQUEST['result_id']</code>
2	status	<p>New: Refers to a new attempt.</p> <p>Update: Refers to a score update on an existing attempt, such as an instructor grading essay or assigning bonus points to update a student's score. Update the existing score with the help of result_id in such a case.</p>	VARCHAR	<code>\$_REQUEST['status_id']</code>
3	quiz_id	A unique ID that will identify your quiz.	INT	<code>\$_REQUEST['quiz_id']</code>
4	quiz_title	A unique quiz title to identify your quiz.	VARCHAR	<code>\$_REQUEST['quiz_title']</code>
5	quiz_name	Name of your quiz.	VARCHAR	<code>\$_REQUEST['quiz_name']</code>
6	attempt_date	The date and time stamp of the attempt (UNIX TIMESTAMP)	TIMESTAMP	<code>\$_REQUEST['attempt_date']</code>

Quiz Maker FAQ ProProfs Quiz Maker FAQs			API & Integration	
7	total_marks	The total number of marks you set for the quiz. E.g., 100.	VARCHAR	\$_REQUEST['total_marks']
8	user_obtained_marks	The total number of marks obtained by the quiz taker.	VARCHAR	\$_REQUEST['user_obtained_marks']
9	user_percent_marks	Percentage marks of quiz takers.	VARCHAR	\$_REQUEST['user_percent_marks']
10	user_totalcorrect_answers	The total number of correct answers by quiz taker.	INT	\$_REQUEST['user_totalcorrect_answers']
11	user_totalwrong_answers	The total number of wrong answer(s) by the quiz taker.	INT	\$_REQUEST['user_totalwrong_answer']
12	user_Id	Quiz taker's ID	VARCHAR	\$_REQUEST['user_Id']
13	user_Email	Quiz taker's email address.	VARCHAR	\$_REQUEST['user_Email']
14	user_Address	Quiz taker's address	VARCHAR	\$_REQUEST['user_Address']
15	user_City	Quiz taker's city	VARCHAR	\$_REQUEST['user_City']
16	user_State	Quiz taker's state	VARCHAR	\$_REQUEST['user_State']
17	user_Zipcode	Quiz taker's zip code	VARCHAR	\$_REQUEST['user_Zipcode']
18	user_Country	Quiz taker's country	VARCHAR	\$_REQUEST['user_Country']

Quiz Maker FAQ ProProfs Quiz Maker FAQs			API & Integration	
19	time_taken	Total Time taken on Quiz by the Quiz taker in 01:05:10 format	VARCHAR	\$_REQUEST['time_taken']
20	time_taken_in_sec	Total Time taken on Quiz by the Quiz taker in seconds	INT	\$_REQUEST['time_taken_in_sec']
21	user_total_unanswered	Total Unanswered Questions count by the Quiz taker	INT	\$_REQUEST['user_total_unanswered']
22	user_Phone	Quiz taker's phone number	VARCHAR	\$_REQUEST['user_Phone']
23	user_name	Quiz taker's name	VARCHAR	\$_REQUEST['user_name']
24	min_pass_marks	Minimum passing marks	INT	\$_REQUEST['min_pass_marks']
25	token	Unique Notification Token	VARCHAR	\$_REQUEST['token']

## REQUEST Method

To get started, you would need to make a file (your callback script) that will accept the data sent by the quiz through the **REQUEST** method.

### Example:

This example uses PHP & MySQL. Test this feature using PHP for server scripting.

**Step 1: SQL:** Create a table with the following SQL query:

```
CREATE TABLE `quiz_takers`

(
    `test_id` int(11) NOT NULL auto_increment,
    `result_id` int(100) NOT NULL,
    `quiz_id` int(100) NOT NULL,
    `quiz_title` varchar(150) NOT NULL,
    `quiz_name` varchar(150) NOT NULL,
    `attempt_date` timestamp NOT NULL,
```

```
`username` varchar(100) NOT NULL,  
`totalscore` int(100) NOT NULL,  
`obtainedmarks` int(100) NOT NULL,  
`userpercentscore` float NOT NULL,  
`totalcorrectanswer` int(50) NOT NULL,  
`totalwronganswer` int(50) NOT NULL,  
`userId` varchar(100) NOT NULL,  
`userEmail` varchar(100) NOT NULL,  
`userAddress` varchar(150) NOT NULL,  
`userCity` varchar(150) NOT NULL,  
`userState` varchar(150) NOT NULL,  
`userZipcode` int(100) NOT NULL,  
`userCountry` varchar(150) NULL,  
`time_taken` varchar(20) NOT NULL,  
`time_taken_in_sec` int(11) NOT NULL,  
`total_unanswered` int(11) NOT NULL,  
`userPhone` int(100) NOT NULL,  
`min_pass_marks` int(100) NULL,  
PRIMARY KEY (`test_id`)  
)
```

**Step 2: PHP:** Below is the **sample code** you can use to code what data is retrieved.

```
$link = mysql_connect('localhost', 'mysql_user', 'mysql_password');  
$sql_query = "INSERT INTO quiz_takers SET  
  
result_id = '". $_REQUEST['result_id']. "' ,  
quiz_id = '". $_REQUEST['quiz_id']. "' ,  
quiz_name = '". $_REQUEST['quiz_name']. "' ,  
attempt_date = '". $_REQUEST['attempt_date']. "' ,  
username = '". $_REQUEST['user_name']. "' ,  
totalscore = '". $_REQUEST['total_marks']. "' ,  
obtainedmarks = '". $_REQUEST['user_obtained_marks']. "' ,  
userpercentscore = '". $_REQUEST['user_percent_marks']. "' ,  
totalcorrectanswer = '". $_REQUEST['user_totalcorrect_answers']. "' ,  
totalwronganswer = '". $_REQUEST['user_totalwrong_answers']. "' ,  
userId = '". $_REQUEST['user_Id']. "' ,  
userEmail = '". $_REQUEST['user_Email']. "' ,  
userAddress = '". $_REQUEST['user_Address']. "' ,
```

```
userCity = '$_REQUEST['user_City'].',
userState = '$_REQUEST['user_State'].',
userZipcode = '$_REQUEST['user_Zipcode'].',
userCountry = '$_REQUEST['user_Country'].',
time_taken = '$_REQUEST['time_taken'].',
time_taken_in_sec = '$_REQUEST['time_taken_in_sec'].',
total_unanswered = '$_REQUEST['user_total_unanswered'].',
userPhone = '$_REQUEST['user_Phone'].';
```

```
$query_run = mysql_query($sql_query);
```

**Step 3:** Download and implement the following PHP code to retrieve data.

[Click here to download the PHP script \(responsetest.php\)](#)

### JSON Method

You can get JSON data by using the POST request. The sample code below shows how data can be retrieved. The POST request is enough to retrieve the data in the JSON method.

**JSON gets two additional attributes and variables over the REQUEST method. With JSON, you can retrieve data for custom questions and tag-based scores.**

Content-Type : application/json

```
//This code will retrieve the data
```

```
$post_json = file_get_contents("php://input");
```

```
$post_json = json_decode($post_json,true);
```

```
$result_id = $post_json['result_id'];
```

```
$token = $post_json['token']; //Secure notification token, get it from my
account as notification token.
```

```
//This is how data will be retrieved
```

```
{
    "token" : "Your Notification Token",
    "result_id" : "206370034",
    "user_name" : "ProProfs",
    "total_marks" : "10",
    "attempt_date" : 1559632348,
    "user_obtained_marks" : 5,
    "user_percent_marks" : "50",
    "user_totalcorrect_answers" : 1,
    "user_totalwrong_answers" : 1,
```

```
"user_Id" : "",
"user_Email" : "someone@example.com",
"user_Address" : "",
"user_City" : "",
"user_State" : "",
"user_Zipcode" : "",
"user_Country" : "",
"time_taken" : "01:05:10",
"time_taken_in_sec" : 3910,
"user_total_unanswered" : 0,
"user_Phone" : "",
"quiz_id" : "2478348",
"quiz_name" : "API Test",
"quiz_title" : "mjQ3odu5nqo7iv",
"min_pass_marks" : "70",

"tag_based_score" : [
  {
    "tag" : "Maths",
    "score" : "5 / 5"
  },
  {
    "tag" : "Physics",
    "score" : "0 / 5"
  }
],
"custom_ques_ans" : [
  {
    "question" : "How did you find us?",
    "answer" : "Search Engine"
  },
  {
    "question" : "How do you commute?",
    "answer" : "Car"
  }
],
"status" : "new"
```

}

### Related Articles:

- [ProProfs REST API Guide](#)
- [How To Auto-Enroll Learners To Classroom Using API](#)
- [How To Track the Progress Of Learners In My Systems Using API](#)