

# Full working PHP example

Below is a full working example in PHP. You can copy & paste this code into a web page on your web server, insert your API key, and the code will work.

```
<?php
$httpApiUrl = "https://api.rescuegroups.org/http/v2.json";
$data = array(
    "apikey" => "XXXXXX",
    "objectType" => "animals",
    "objectAction" => "publicSearch",
    "search" => array (
        "calcFoundRows" => "Yes",
        "resultStart" => 0,
        "resultLimit" => 20,
        "resultSort" => "animalID",
        "fields" => array("animalID", "animalOrgID", "animalName", "animalSpecies", "animalBreed", "
animalThumbnailUrl"),
        "filters" => array (
            array(
                "fieldName" => "animalStatus",
                "operation" => "equals",
                "criteria" => "Available"
            )
        )
    )
);
$result = postToApi($data);
if (!$result) {
    echo "login issue with the API.";
    exit;
}
print_r($result);
exit;

function postJson($url, $json) {
    // create a new cURL resource
    $ch = curl_init();
    // set options, url, etc.
    curl_setopt($ch, CURLOPT_HTTPHEADER, array("Content-Type: application/json"));
    curl_setopt($ch, CURLOPT_URL, $url);
    curl_setopt($ch, CURLOPT_POSTFIELDS, $json);
    curl_setopt($ch, CURLOPT_POST, 1);
    //curl_setopt($ch, CURLOPT_VERBOSE, true);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
    // grab URL and pass it to the browser
    $result = curl_exec($ch);
    if (curl_errno($ch)) {
        // TODO: Handle errors here
        return array(
            "result" => "",
            "status" => "error",
            "error" => curl_error($ch)
        );
    } else {
        // close cURL resource, and free up system resources
        curl_close($ch);
    }
    return array(
        "status" => "ok",
        "error" => "",
        "result" => $result,
    );
}

function postToApi($data) {
    $resultJson = postJson($GLOBALS["httpApiUrl"], json_encode($data));
    if ($resultJson["status"] == "ok") {
        $result = json_decode($resultJson["result"], true);
        $jsonError = getJsonError();
        if (!$jsonError && $resultJson["status"] == "ok") {
            return $result;
        }
    }
}
```

```

        } else {
            return array (
                "status" => "error",
                "text" => $result["error"] . $jsonError,
                "errors" => array()
            );
        }
    } else return false;
}

function getJsonError() {
    switch (json_last_error()) {
        case JSON_ERROR_NONE:
            return false;
            break;
        case JSON_ERROR_DEPTH:
            return "Maximum stack depth exceeded";
            break;
        case JSON_ERROR_STATE_MISMATCH:
            return "Underflow or the modes mismatch";
            break;
        case JSON_ERROR_CTRL_CHAR:
            return "Unexpected control character found";
            break;
        case JSON_ERROR_SYNTAX:
            return "Syntax error, malformed JSON";
            break;
        case JSON_ERROR_UTF8:
            return "Malformed UTF-8 characters, possibly incorrectly encoded";
            break;
        default:
            return "Unknown error";
            break;
    }
}

```