

# Anleitung für die Nutzung der Bahn-API in PHP

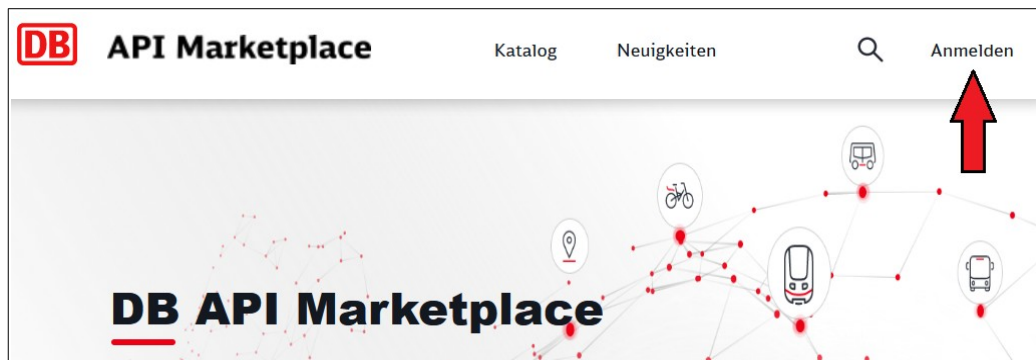
Sie sollen in einem Projekt mit der Bahn-API arbeiten und eine kleine Web-App dafür erstellen. Diese Anleitung soll Ihnen bei der Einrichtung der API und der Nutzung mit PHP helfen.

## Bei Bahn registrieren und die APIs abonnieren

Unter folgendem Link finden Sie die Entwicklerseite der Deutschen Bahn:

<https://developers.deutschebahn.com>

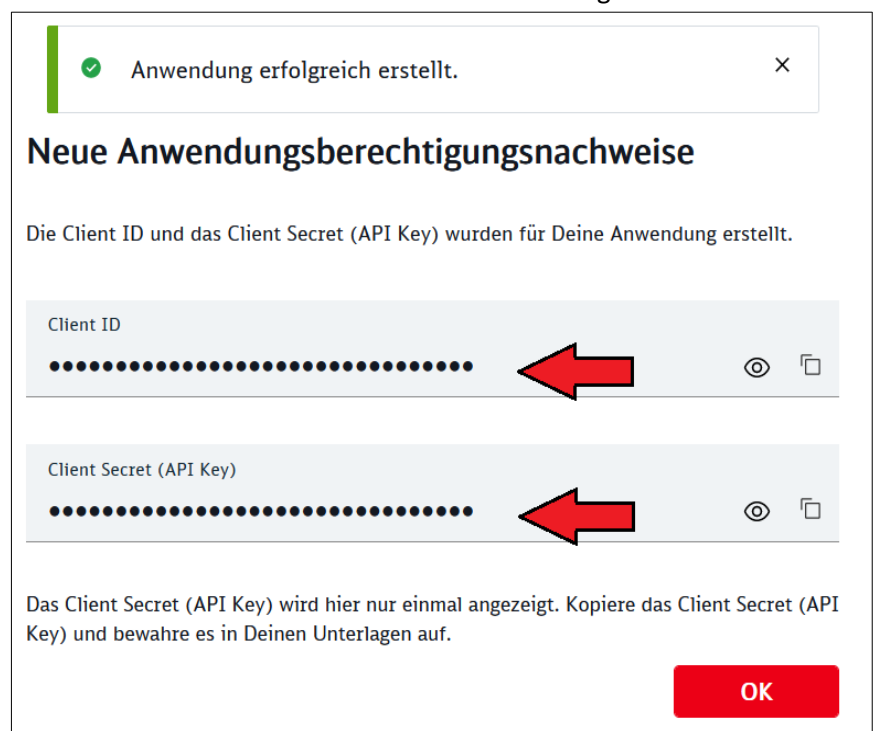
Auf der Startseite finden Sie oben rechts in der Ecke einen Button zum Registrieren.



Füllen Sie das Formular aus und bestätigen Sie die Emailbenachrichtigung. Anschließend können Sie sich auf dem Portal einloggen. Nach dem Login können Sie sich nun aus dem API-Pool die für Sie notwendigen APIs aussuchen. Im Normalfall sollten dies die **StaDa – Stations Data Free4all API** und die **Fahrplan - Free4all API** sein. Klicken Sie dafür einfach in der API-Übersicht auf die gewünschte API und dann im Fenster auf "Abonnieren". Evtl. müsse Sie vorher noch eine Anwendung auswählen/erstellen. Nach der Auswahl "Neue Anwendung erstellen" dürfte der Weg selbsterklärend sein. Zu dieser werden dann die beiden API's hinzugefügt.

**Achtung:** Die von Ihnen erstellte Anwendung ist Ihr Zugang zu den API's und Sie erhalten hierfür 2 wichtige ID's.

Lassen Sie sich diese sichtbar machen und speichern Sie sie ab. Sie benötigen diese später!



Diese beiden free4all API-Zugänge sind beschränkt in der Anzahl der Aufrufe pro Zeiteinheit. Behalten Sie dies im Hinterkopf.

Nun sind Sie mit der Einrichtung der API-Zugänge fertig und können Anfragen stellen. Dies kann entweder über die Nutzeroberfläche dieser Webseite erfolgen oder auch über ein PHP-Programm.

## Anfragen über die Webseite stellen

Folgen Sie folgendem Pfad:

Anwendungen → Ihre Anwendung → Auswahl des Produktabbonements (z. B.: Fahrplan (1.1.688))  
→ Zugehörige API (z. B.: Fahrplan Version: 1.0.2)

Hier sehen Sie nun auf der linken Seite die möglichen Abfragefunktionen.

Filter

**Überblick**

- GET /arrivalBoard/{id}
- GET /departureBoard/{id}
- GET /journeyDetails/{id}
- GET /location/{name}
- Definitionen

**Überblick**

A RESTful webservice to request a railway journey

Typ

REST **OAS3**

Protokoll

HTTPS

Wenn Sie nun eine der Abfragefunktionen auswählen öffnet sich eine Beschreibung zur Nutzung samt Rückgabetypen und Sie können auch eine Testanfrage senden. Hierfür klicken Sie auf „Versuch' es“. Nun werden die ID's Ihrer Anwendung, sowie die notwendigen Übergabewerte für die Abfrage an die API abgefragt. Wenn Sie alles korrekt eingetragen haben, sollte eine positive Rückmeldung erfolgen.

Filter

**Überblick**

- GET /arrivalBoard/{id}**
- GET /departureBoard/{id}
- GET /journeyDetails/{id}
- GET /location/{name}
- Definitionen

**Get arrival board of a location** **Fahrplan**

Details **Versuch' es!**

GET

https://apis.deutschebahn.com/db-api-marketplace/apis/fahrplan/v1/arrivalBoard/{id}

Sicherheit

^ Identifikation

Client ID **Client-ID kopieren**

Sonstige

Geben Sie Ihre Client ID ein.

Nur abonnierte Berechtigungsnachweise können ausgewählt werden.

## Einbindung und Nutzung in PHP

Um die API nun codeseitig nutzen zu können, beschreibe ich Ihnen dies einmal an einer Beispieldatei (API\_Beiispiel.php). Sie müssen in Ihrem Code einen Stream zu der API-URL starten und dann die Daten schicken und empfangen. Dies geschieht mit cURL.

```
1  <?php
2  $ch = curl_init('https://apis.deutschebahn.com/db-api-marketplace/apis/fahrplan/v1/location/Magdeburg%20Hbf');
3  curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
4  curl_setopt($ch, CURLOPT_HTTPHEADER, array(
5      'Accept: application/json',
6      'DB-Client-Id: 676b64cd3619162dba9d6396e1ecc92a',
7      'DB-API-Key: 105d05e3d9c4df4bb2a6d4676089f5e7'
8  ));
9
10 // execute!
11 $response = curl_exec($ch);
12
13 // close the connection, release resources used
14 curl_close($ch);
15
16 // do anything you want with your response
17 $response = json_decode($response, true);
18 echo $response[0]['name'];
19
20 //var_dump($response);
21 ?>
```

In Zeile 2 sehen Sie die Initialisierung des cURL. Hier müssen Sie die URL der API eingeben.

In Zeile 3 erlauben Sie eine Antwort. Die Funktion "curl\_setopt" beschreibt dabei das festlegen optionaler Parameter.

In den Zeilen 4 bis 8 legen wir den Header des cURL fest. Dieser ist wichtig, damit die API weiß, dass wir berechtigt sind. Fügen Sie hier Ihre beiden ID's ein. Den Rest können Sie einfach so übernehmen.

In Zeile 11 wird der cURL nun ausgeführt und das Ergebnis in "response" gespeichert. Das Ergebnis kommt als ein String in JSON-Codierung. Dies ist einfach ein langer String, der Arrays und Objekte nach festem Schema beschreibt. Wenn Sie sich das Rohergebnis anschauen möchten, einfach Zeile 17/18 auskommentieren und Zeile 20 dekommentieren (mir fiel kein anderes Wort ein ^^).

In Zeile 14 wird der Stream wieder geschlossen. Wichtig!

Zeile 17 dekodiert den JSON-String dann in ein Array (in PHP eine gegebene Funktion) und Zeile 18 gibt dann einen Teil des Arrays aus.