

# Php Rest

== Beschreibung == REST baut auf HTTP auf und nutzt seine simpelsten Features für eine einfache API, die jede Programmiersprache beherrscht.

Dabei wird jede Ressource wie mit einem Primärschlüssel über eine URL identifiziert. Und über die HTTP Aktionen GET, POST, PUT, usw. können dann auf dieser Ressource Updates durchgeführt werden.

## Inhaltsverzeichnis

- [1 Befehle](#)
- [2 PHP Codes](#)

Im Gegensatz zu vielen RPC-Architekturen kodiert REST keine Methodeninformation in den URI, da der URI Ort und Namen der Ressource angibt, nicht aber die Funktionalität, die die Ressource anbietet. Es wird versucht, die Funktionalität hauptsächlich über die HTTP-Verben abzubilden.

## Befehle

Der Client kann folgende Befehle absetzen

**GET** fordert die angegebene Ressource vom Server an.

**POST** fügt eine neue (Sub-)Ressource unterhalb der angegebenen Ressource ein. Da die neue Ressource noch keine URI besitzt, adressiert die URI die übergeordnete Ressource. Als Ergebnis wird der neue Ressourcenlink dem Client zurückgegeben. POST kann im weiteren Sinne auch dazu verwendet werden Operationen abzubilden, die von keiner anderen Methode abgedeckt werden.

**PUT** die angegebene Ressource wird angelegt oder geändert.

**DELETE** löscht die angegebene Ressource.

**HEAD** fordert Metadaten zu einer Ressource vom Server an.

**OPTIONS** prüft, welche Methoden auf einer Ressource zur Verfügung stehen

## PHP Codes

Wertvolle Codes findet man unter den folgenden Links

- [gen-x-design.com/archives/making-restful-requests-in-php/](http://gen-x-design.com/archives/making-restful-requests-in-php/)
- [gen-x-design.com/archives/create-a-rest-api-with-php/](http://gen-x-design.com/archives/create-a-rest-api-with-php/)
- [wezfurlong.org/blog/2006/nov/http-post-from-php-without-curl](http://wezfurlong.org/blog/2006/nov/http-post-from-php-without-curl)