

Application Programming Interface

Eine API ist eine Schnittstelle für mehrere Programmierer (Kurz gesagt eine Anwendung-Programmierschnittstelle) die an einem Projekt arbeiten.

Die API ist quasi nur eine Anbindung für das aktuelle System bzw. aktuelle Programm um das programmieren zu vereinfachen.

Der Begriff [API](#):

Der Begriff [API](#) ist nur eine Kurzfassung von "*Application Programming Interface*". Übersetzt heißt das ganze einfach nur Anwendung-Programmierschnittstelle. Das Wort [API](#) wird in jeder Programmiersprache eine Rolle spielen sobald man von einem externen Projekt bzw. einem eingebundenen Projekt etwas erbt oder benutzt.

Ein kurzes Beispiel wird **jetzt** hier erläutert:

> Stell dir vor du möchtest ein Spigot-Plugin programmieren, dies geht allerdings nur sofern du die **Spigot-[Api](#)** einbindest.

Sobald du die Spigot-[Api](#) eingebunden hast kannst du direkt loslegen mit deinem Plugin. Du machst das übliche; baust die

`#onEnable()` und die `#onDisable()` Methoden ein, erstellst externe Klassen und tust dort deine EventHandler's rein packen

Damit diese aber funktionieren musst du diese registrieren mit `#registerEvents(---,---);`. Alles schön und gut.

> Nun stell dir nun den Teil vor wo du die Spigot-[Api](#) nicht eingebunden hättest. Es gäbe kein `#onEnable()`, kein `#onDisable()`;

Keine EventHandler's und selbstverständlich keine `#registerEvents(---,---);` Methode.

Merke: Sobald du etwas einbindest in dein Projekt was du nicht selber geschrieben/programmiert hast und nicht in Form von einem Code

Schnipsel ist, wird als [API](#) bezeichnet. Eine Schnittstelle für andere Programmierer um das programmieren zu vereinfachen bzw. als Hilfsmittel bereit zu stellen.

Weiteres Beispiel:

> Nehmen wir an du hast ein Java Projekt angelegt und benutzt dabei Maven. Jedes Maven-Projekt beinhaltet eine Datei namens "*pom.xml*".

In dieser pom.xml kannst du den Ausgabe-Pfad angeben für deine Jar-Datei, die Install-Methode auswählen und vieles mehr. Du kannst auch

Dependencies einbinden. Sobald du eine **Dependency** einfügst in dem `<dependencies> ... </dependencies>` Tag/Bereich ist sofort die Sprache

von einer [API](#).

---> Nochmal als Erklärung: Du bindest ein Projekt ein was andere geschrieben haben, dies wird dann automatisch als [API](#) bezeichnet da dass eine Schnittstelle/Erweiterung für dich zum programmieren ist.

Hier ein Nachtrag von Janhektor [Danke an der Stelle]:



[Zitat von Janhektor](#)

Besonders bei C oder C++ als Programmiersprache wird das noch deutlich wichtiger: Hier ist es sogar vorgesehen, als Schnittstelle Header-Files zur Verfügung zu stellen und die Binärdistribution nicht als [API](#) zu benutzen (das geht zumeist auch gar nicht, weil bereits kompiliert wurde). So gibt es z.B. Dynamic Linked Libarys (DLL) oder statische Bibliotheken. Auch ein paar Klassen, die einfach in das eigene Projekt kopiert werden, können eine Schnittstelle darstellen