A5

JavaScriptCore ist die JavaScript-Engine, die von Apple entwickelt wird und in Safari-Browsern verwendet wird. Es unterstützt Set.prototype und Array.prototype.flat(depth). Es unterstützt jedoch nicht Static Blocks in Klassen zur Initialisierung statischer Variablen, Array.prototype.group oder das neue Top-Level-Namespace-Objekt "Temporal".

V8 ist die JavaScript-Engine, die von Google entwickelt wird und in Chrome-Browsern verwendet wird. Es unterstützt alle fünf Sprachkonstrukte: Set.prototype, Static Blocks in Klassen zur Initialisierung statischer Variablen, Array.prototype.flat(depth), Array.prototype.group und das neue Top-Level-Namespace-Objekt "Temporal".

SpiderMonkey ist die JavaScript-Engine, die von Mozilla entwickelt wird und in Firefox-Browsern verwendet wird. Es unterstützt Set.prototype, Array.prototype.flat(depth) und das neue Top-Level-Namespace-Objekt "Temporal". Es unterstützt jedoch nicht Static Blocks in Klassen zur Initialisierung statischer Variablen oder Array.prototype.group.

A6

Amazon Product Advertising API (PA-API) Die Amazon Product Advertising API ist eine RESTful-API, die es Entwicklern ermöglicht, Amazon-Produktdaten in ihre eigenen Anwendungen zu integrieren. Die API ermöglicht den Zugriff auf Produktinformationen, Kundenbewertungen und Kaufmöglichkeiten auf Amazon. Entwickler können mit der API Produkte durchsuchen, Produktdetails und -bilder abrufen und Kundenbewertungen abrufen. Die API unterstützt auch die Erstellung von Partnerlinks, die es Entwicklern ermöglichen, eine Provision für den Verkauf von Amazon-Produkten über ihre Anwendungen zu verdienen.

REST-Prinzipien:

Die PA-API implementiert die REST-Prinzipien der Ressourcenidentifikation, der Zustandslosigkeit und der einheitlichen Schnittstelle. Die API verwendet HTTP-Methoden wie GET, POST und DELETE und gibt Daten im JSON-Format zurück.

Level nach dem Richardson Maturity Model:

Die Amazon Product Advertising API ist auf Stufe 2 des Richardson Maturity Models. Obwohl es die REST-Prinzipien der Ressourcenidentifikation, der Zustandslosigkeit und der einheitlichen Schnittstelle implementiert, fehlt es der API an Hypermedia-Kontrollen, die es Benutzern ermöglichen würden, durch die API zu navigieren und ihre Funktionen zu entdecken.

Versionierung:

Die API verwendet eine URI-basierte Versionierung, bei der die Versionsnummer in der URI enthalten ist. Zum Beispiel kann ein Entwickler die Version 5 der API mit dem folgenden Endpunkt aufrufen: <https://webservices.amazon.com/paapi5>.

2. Amazon Web Services (AWS) API

Die Amazon Web Services (AWS) API ist eine RESTful-API, die es Entwicklern ermöglicht, auf die verschiedenen Cloud-Dienste von Amazon zuzugreifen, darunter Computing, Speicherung, Datenbanken, Analysen und künstliche Intelligenz. Mit der API können Entwickler virtuelle Maschinen starten, Datenbanken erstellen, Dateien speichern und vieles mehr.

REST-Prinzipien:

Die AWS API implementiert die REST-Prinzipien der Ressourcenidentifikation, der Zustandslosigkeit und der einheitlichen Schnittstelle. Die API verwendet HTTP-Methoden wie GET, POST, PUT und DELETE und gibt Daten im JSON-Format zurück.

Level nach dem Richardson Maturity Model:

Die Amazon Web Services API ist auf Stufe 3 des Richardson Maturity Models. Die API implementiert nicht nur die REST-Prinzipien der Ressourcenidentifikation, der Zustandslosigkeit und der einheitlichen Schnittstelle, sondern auch Hypermedia-Kontrollen, die es Benutzern ermöglichen, durch die API zu navigieren und ihre Funktionen zu entdecken.

Versionierung:

Die API verwendet eine URI-basierte Versionierung, bei der die Versionsnummer in der URI enthalten ist. Zum Beispiel kann ein Entwickler die Version 2 der API mit dem folgenden Endpunkt aufrufen: <https://api.aws.amazon.com/v2>.