# Apify: Web-Automatisierung clever und effizient meistern

Category: Online-Marketing
geschrieben von Tobias Hager | 6. Februar 2026




# Apify: Web-Automatisierung clever und effizient meistern

Scraping ist tot? Von wegen. Wer heute noch manuell Daten zusammensammelt, lebt digital im Mittelalter. Willkommen im Zeitalter der Web-Automatisierung – und Apify ist dein Schweizer Taschenmesser dafür. Ob du komplexe Websites crawlen, APIs emulieren oder ganze Geschäftsprozesse automatisieren willst: Apify macht’s möglich – schnell, skalierbar und erschreckend effizient. Klingt zu gut, um wahr zu sein? Dann lies weiter. Es wird technisch. Es wird dreckig. Es wird schön.

- Was Apify ist – und warum es mehr als ein Scraping-Tool ist

- Wie du mit Apify komplexe Web-Automatisierungen umsetzt
- Der Unterschied zwischen Actor, Task und Dataset erklärt
- Warum Puppeteer und Playwright das Herzstück moderner Automatisierung sind
- Wie du mit Apify APIs baust, ohne dass eine API existiert
- Best Practices für skalierbare und robuste Web-Scraper
- Wie du mit Apify Workflows orchestrierst und automatisierst
- Apify vs. klassische Scraper – warum du deinen alten Code in Rente schicken solltest
- Step-by-Step: So baust du deinen ersten Actor in unter 20 Minuten
- Fazit: Web-Automatisierung ist kein Hack mehr, sondern Business-Infrastruktur

# Was ist Apify? Web-Automatisierung neu gedacht

Apify ist kein weiteres “Scrape-mir-die-Seite”-Tool. Es ist eine Plattform zur großflächigen Web-Automatisierung und Datenextraktion. Wer heute Daten aus dem Web braucht – sei es für Marktanalysen, Preisvergleiche, Lead-Generierung oder Wettbewerbsbeobachtung – braucht mehr als einen Python-Script mit `requests` und `BeautifulSoup`. Apify liefert dir eine komplette Serverless-Infrastruktur, um maßgeschneiderte Scraper, Bots und Workflows zu bauen – ohne sich mit Deployment, Skalierung oder Proxy-Rotation herumzuärgern.

Der Clou: Apify basiert auf sogenannten “Actors” – kleine, isolierte Anwendungen, die du in JavaScript oder mit No-Code-Tools schreiben kannst. Diese Actors können Websites durchsuchen, Formulare ausfüllen, Daten extrahieren, APIs imitieren oder sogar Browser automatisch steuern. Der Fokus liegt nicht auf billiger Massenextraktion, sondern auf Qualität, Wiederholbarkeit und Skalierbarkeit. Web-Automatisierung auf Enterprise-Niveau, auch wenn du solo arbeitest.

Und während viele Tools bei dynamischen Seiten (hallo, React & Co.) kläglich scheitern, setzt Apify auf Headless-Browsing mit Puppeteer und Playwright. Damit werden auch Single-Page-Applications zuverlässig durchleuchtet – inklusive Content, der erst nach dem Scrollen oder durch JavaScript geladen wird. Willkommen im modernen Web-Scraping.

Anders gesagt: Apify ist der Unterschied zwischen einem Hobby-Scraper und einem professionellen Data Engineer. Und das Beste? Es gibt eine REST-API, CLI, SDKs und eine eigene Integration in GitHub Actions. Wer ernsthaft Web-Automatisierung betreiben will, kommt an Apify 2025 nicht mehr vorbei.

# Actors, Tasks und Datasets:

# Die Apify-Grundlagen

Um Apify effizient zu nutzen, musst du das Vokabular verstehen. Der zentrale Begriff: Actor. Ein Actor ist ein ausführbares Stück Code, das automatisiert eine Aufgabe erledigt – z. B. eine Website scrapen, ein Formular ausfüllen oder einen Workflow starten. Jeder Actor kann parametrisiert, geplant und über die API aufgerufen werden. Du kannst eigene Actors schreiben oder bestehende aus dem Apify Store verwenden.

Ein Task ist eine vorkonfigurierte Ausführung eines Actors mit bestimmten Parametern. Denk an ein Template: Du definierst, welche URL gecrawlt, wie tief gescannt oder welche Selektoren verwendet werden sollen – und speicherst das Ganze als Task. So kannst du dieselbe Actor-Logik mehrfach nutzen, ohne jedes Mal von vorn zu konfigurieren.

Und dann gibt’s Datasets. Jeder Actor kann Daten erzeugen – sei es HTML, JSON, CSV oder strukturierte Felder. Diese Daten landen in einem Dataset, das du über die API abrufen, exportieren oder in Realtime verarbeiten kannst. Die Datasets sind versioniert, persistent und performant abfragbar – auch bei Millionen von Einträgen.

Zusätzlich gibt’s noch Key-Value-Stores (für Configs oder Sessions) und Request-Queues (für komplexe Crawling-Logik). Klingt nach DevOps? Ist es auch. Aber keine Sorge: Apify kapselt die Komplexität so gut, dass du dich auf das Wesentliche konzentrieren kannst – nämlich funktionierende Automatisierung.

# Headless-Browsing mit Puppeteer und Playwright: Apifys Superkräfte

Vergiss `curl`. Vergiss `requests`. Moderne Websites sind keine statischen HTML-Seiten mehr, sondern interaktive JavaScript-Frankensteins, die Content dynamisch nachladen, per AJAX kommunizieren und sich verhalten wie Mini-Apps. Klassisches Scraping versagt hier – weil der Content schlicht noch nicht existiert, wenn der Crawler ihn abruft.

Hier kommt Headless-Browsing ins Spiel – und Apify hat gleich zwei Schwergewichte eingebaut: Puppeteer und Playwright. Beide Tools steuern einen echten Browser (meist Chromium), rendern die Seite komplett und erlauben dann das gezielte Extrahieren von DOM-Elementen. Damit erreichst du Inhalte, die klassische Scraper nie sehen – inklusive Shadow DOM, Lazy Loading und interaktiven Komponenten.

Der Vorteil von Playwright: Es unterstützt mehrere Browser (Chromium, Firefox, WebKit), erlaubt paralleles Browsing in mehreren Tabs und ist robuster beim Umgang mit asynchronem Content. Puppeteer dagegen ist schlanker

und schneller, wenn du nur Chromium brauchst. Beide sind über Apify nahtlos integriert. Du kannst sogar zwischen ihnen wechseln – ohne deine Actor-Logik neu zu schreiben.

Und das Beste: Apify kümmert sich um Proxies, Captcha-Schutz, Session-Rotation und Anti-Bot-Erkennung. Du konzentrierst dich auf den Code – Apify übernimmt den Rest. So sieht moderne Web-Automatisierung aus.

# Datenextraktion, API-Emulation und Workflows: Was mit Apify alles geht

Apify kann mehr als nur Daten auslesen. Du kannst Websites automatisieren, als würdest du selbst davor sitzen. Das bedeutet: Formulare ausfüllen, Captchas lösen (mit externen Services), Login-Prozesse durchlaufen, Cookies speichern, Warenkörbe befüllen, Klickpfade simulieren – alles per Script und ohne manuelle Interaktion. Willkommen im Web-Automatisierungs-Olymp.

Ein besonders spannender Use-Case: API-Emulation. Viele Seiten bieten keine öffentliche API – oder verstecken sie hinter Authentifizierung, Throttling oder Geo-Blocking. Mit Apify kannst du die Website selbst als API "emulieren": Scrape die Infos, strukturiere sie in Datasets, und rufe sie über deine eigene REST-API ab. Du baust dir deine eigene Infrastruktur – ohne die Erlaubnis des Anbieters zu brauchen.

Noch mächtiger wird's mit Workflows. Du kannst mehrere Actors zu einem orchestrierten Ablauf zusammenfügen – inklusive Trigger, Abhängigkeiten und Fehlerbehandlung. Beispiel: Ein Actor crawlt Produkte, ein zweiter prüft Preisänderungen, ein dritter sendet Alerts per Slack oder E-Mail. Vollautomatisch. Täglich. Ohne dein Zutun.

Mit Webhooks kannst du externe Systeme anbinden – z. B. CRM, BI-Tools oder Datenbanken. Oder du nutzt Apifys Integrationen mit Zapier, Make.com oder n8n. Die Möglichkeiten sind endlos – wenn du weißt, was du tust.

# Step-by-Step: Deinen ersten Apify Actor bauen

Genug Theorie. Jetzt wird es praktisch. So baust du dir in unter 20 Minuten deinen ersten funktionierenden Apify Actor – inklusive Datenextraktion und Output als JSON:

1. Apify Account anlegen: Kostenlos auf apify.com registrieren.
2. Neuen Actor erstellen: In der Plattform auf "Create new Actor" klicken. Wähle Node.js als Runtime.
3. Code schreiben: Im integrierten Editor folgenden Code einfügen:

```
const { Actor } = require('apify');

Actor.main(async () => {
  const input = await Actor.getInput();
  const { url } = input;

  const browser = await Actor.launchPuppeteer();
  const page = await browser.newPage();
  await page.goto(url);

  const title = await page.title();
  await Actor.pushData({ title });

  await browser.close();
});
```

4. Input definieren: Lege ein JSON-Input-Feld an mit `{ „url“: „https://example.com“ }`.
5. Run starten: Actor ausführen – und im Dataset erscheint der Seitentitel von example.com.

Das war’s. Dein erster Actor läuft. Und du kannst ihn jederzeit erweitern – um DOM-Selektoren, Screenshots, Login-Flows oder API-Aufrufe. Willkommen im Club.

# Fazit: Apify ist die Zukunft der Web-Automatisierung

Web-Automatisierung war lange ein dunkles Kapitel voller kaputter Python-Scripts, gesperrter IPs und fragiler Cronjobs. Apify hat das Spiel neu definiert: Mit strukturierter Infrastruktur, skalierbaren Workflows und einer Entwicklererfahrung, die ihresgleichen sucht. Wer heute noch mit Selenium auf localhost herumdoktort, hat den Anschluss verpasst.

Ob du Marketplaces analysieren, Wettbewerber beobachten oder interne Prozesse automatisieren willst – Apify ist dein Toolkit. Und zwar nicht als Hack, sondern als professionelles Fundament. Web-Scraping ist nicht tot. Es ist erwachsen geworden. Und Apify ist der Beweis.