

API integration

The following tutorials provide a detailed, step-by-step guide on how to integrate Account Abstraction using APIs. If you're developing in TypeScript, it's recommended to utilize the SDK. Refer to these tutorials for comprehensive insights.

[## User sponsored](#) This tutorial represents the API integration flow where smart contract wallet has the native tokens to sponsor transaction. [## Paymaster sponsored](#) This tutorial represents the API integration flow where paymaster is used to sponsor transactions. [## Sponsor using ERC20 tokens](#) This tutorial represents the API integration flow where ERC20 tokens in smart wallet are used for the transactions. paymaster will cover the gas fees and equivalent ERC20 tokens will be deducted from user's smart account. [## UserOp](#) UserOp signature binds the userOpHash, preventing any changes to the userOp once the signature is generated. If any of the userOp fields is changed, it will require re-signing. Signature is generated based on the validation scheme used. [Previous](#) [Migrating to V4](#) [Next](#) [User sponsored](#)