



LI.FI SDK v2 Interface

Technical overview about LI.FI's SDK. ``

Copy interfaceLIFI{

getConfig():Config// Get the current configuration of the SDK

getConfigAsync():Promise// Get the SDK configuration after all setup calls are finished

getRpcProvider(chainId:number,archive:boolean):Promise// Get an instance of a provider for a specific chain

setConfig(configUpdate:ConfigUpdate):Config// Set a new confuration for the SDK

getPossibilities(request?:PossibilitiesRequest,options?:RequestOptions):Promise// Get a set of current possibilities based on a request that specifies which chains, exchanges and bridges are preferred or unwanted

getToken(chain:ChainKey|ChainId,token:string,options?:RequestOptions):Promise// Fetch information about a Token

getQuote(request:QuoteRequest,options?:RequestOptions):Promise// Get a quote for a token transfer

getContractCallQuote(request:ContractCallQuoteRequest,options?:RequestOptions):Promise// Get a quote for a destination contract call

getStatus(request:GetStatusRequest,options?:RequestOptions):Promise// Check the status of a transfer. For cross chain transfers, the "bridge" parameter is required

getTools(request:ToolsRequest,options?:RequestOptions):Promise// Get the available tools to bridge and swap tokens

getTokens(request?:TokensRequest,options?:RequestOptions):Promise// Get all known tokens

getChains():Promise// Get all available chains

getRoutes(request:RoutesRequest,options?:RequestOptions):Promise// Get a set of routes for a request that describes a transfer of tokens

getStepTransaction(step:LifiStep,options?:RequestOptions):Promise// Get the transaction data for a single step of a route

getGasRecommendation(request:GasRecommendationRequest,options?:RequestOptions):Promise// Get gas recommendation for a certain chain

getTokenBalance(walletAddress:string,token:Token):Promise// Returns the balances of a specific token a wallet holds across all aggregated chains

getTokenBalances(walletAddress:string,token:Token[]):Promise// Returns the balances for a list tokens a wallet holds across all aggregated chains

getTokenBalancesForChains(walletAddress:string,tokensByChain:{ [chainId:number]:Token[] }):Promise<{ [chainId:number]:TokenAmount[] }>// This method queries the balances of tokens for a specific list of chains for a given wallet.

getTokenApproval(signer:Signer,token:Token,approvalAddress:string):Promise// Get the current approval for a certain token

bulkGetTokenApproval(signer:Signer,tokenData:RevokeTokenData[]):Promise<{token:Token; approval:string|undefined}[]>// Get the current approval for a list of token / approval address pairs

approveToken(request:ApproveTokenRequest[]):Promise// Set approval for a certain token and amount

revokeTokenApproval(request:RevokeApprovalRequest):Promise// Revoke approval for a certain token

getConnections(connectionRequest:ConnectionsRequest):Promise// Get all the available connections for swap / bridging tokens

executeRoute(signer:Signer,route:Route,settings?:ExecutionSettings):Promise// Execute a route.

resumeRoute(signer:Signer,route:Route,settings?:ExecutionSettings):Promise// Resume the execution of a route that has been stopped or had an error while executing.

updateRouteExecution(route:Route,settings:Pick):void// Updates route execution to background or foreground state

```
updateExecutionSettings(settings:ExecutionSettings,route:Route):void// Update the ExecutionSettings for an active route

moveExecutionToBackground(route:Route):void// Executes a route until a user interaction is necessary (signing
transactions, etc.) and then halts until the route is resumed

stopExecution(route:Route):Route// Stops the execution of an active route

getActiveRoutes():Route[]// Get a list of active routes

getActiveRoute(route:Route):Route|undefined// Return the current route information for given route. The route has to be
active

convertQuoteToRoute(step:LifiStep):Route// Converts a quote to Route

}
```

`` Last updated 8 months ago On this page Was this helpful? [Export as PDF](#)