

3

Request a Route

Before any swap or bridging can happen, you need to request the best route from our smart routing API. To request possible routes, please follow the instructions.

Step 1: Build a RoutesRequest object

The RoutesRequest object describes a desired any-to-any transfer and contains all information necessary to calculate the most efficient routes. It is defined by the following interface.

...

```
Copy interface RoutesRequest { fromChainId: number; fromAmount: string; fromTokenAddress: string; fromAddress?: string; toChainId: number; toTokenAddress: string; toAddress?: string; options?: RouteOptions; }
```

...

It is important to note that the fromTokenAddress must be the correct address for the specified fromChainId as the token address changes for a given chain. The same has to be considered for the toTokenAddress and the toChainId. The optional properties fromAddress and toAddress specify where the token comes from and where the SDK will send it to once the transaction is completed.

Further specifications using RouteOptions

Please take a look at [Set Route Options](#) to get access to preference patterns, whitelisting, and blacklisting of cross-chain bridges and exchanges.

Step 2: Call getRoutes

After creating the routesRequest object, you can pass it to the getRoutes functions to start calculating the routes.

...

```
Copy const result = await lifi.getRoutes(routesRequest) const routes = result.routes
```

...

getRoutes returns a RoutesResponse which contains a list of routes for the described transfer. If an order was specified as part of the RouteOptions object, the resulting list would be sorted accordingly.

Step 3: Done, you can now execute the route

Learn [Execute a Route \(Transfer\)](#)

Example code snippet

The following code snippet requests Routes for the following statement:

Bridge and swap 1 USDT on xDai to the maximum amount of USDC on BSC.

...

```
Copy const routeOptions = { slippage: 3 / 100, // 3% order: 'RECOMMENDED' }
```

```
const routesRequest = { fromChainId: 100, fromAmount: '1000000', // 1 USDT
  fromTokenAddress: '0x4ecaba5870353805a9f068101a40e0f32ed605c6', toChainId: 56,
  toTokenAddress: '0x8ac76a51cc950d9822d68b83fe1ad97b32cd580d', options: routeOptions, }
```

```
const result = await lifi.getRoutes(routesRequest) const routes = result.routes
```

...

Last updated 8 months ago On this page * [Step 1: Build a RoutesRequest object](#) * [Step 2: Call getRoutes](#) * [Step 3: Done, you can now execute the route](#) * [Example code snippet](#)

Was this helpful? [Export as PDF](#)