Sui SDK and Dapp-Kit

Justa from Bucket Protocol

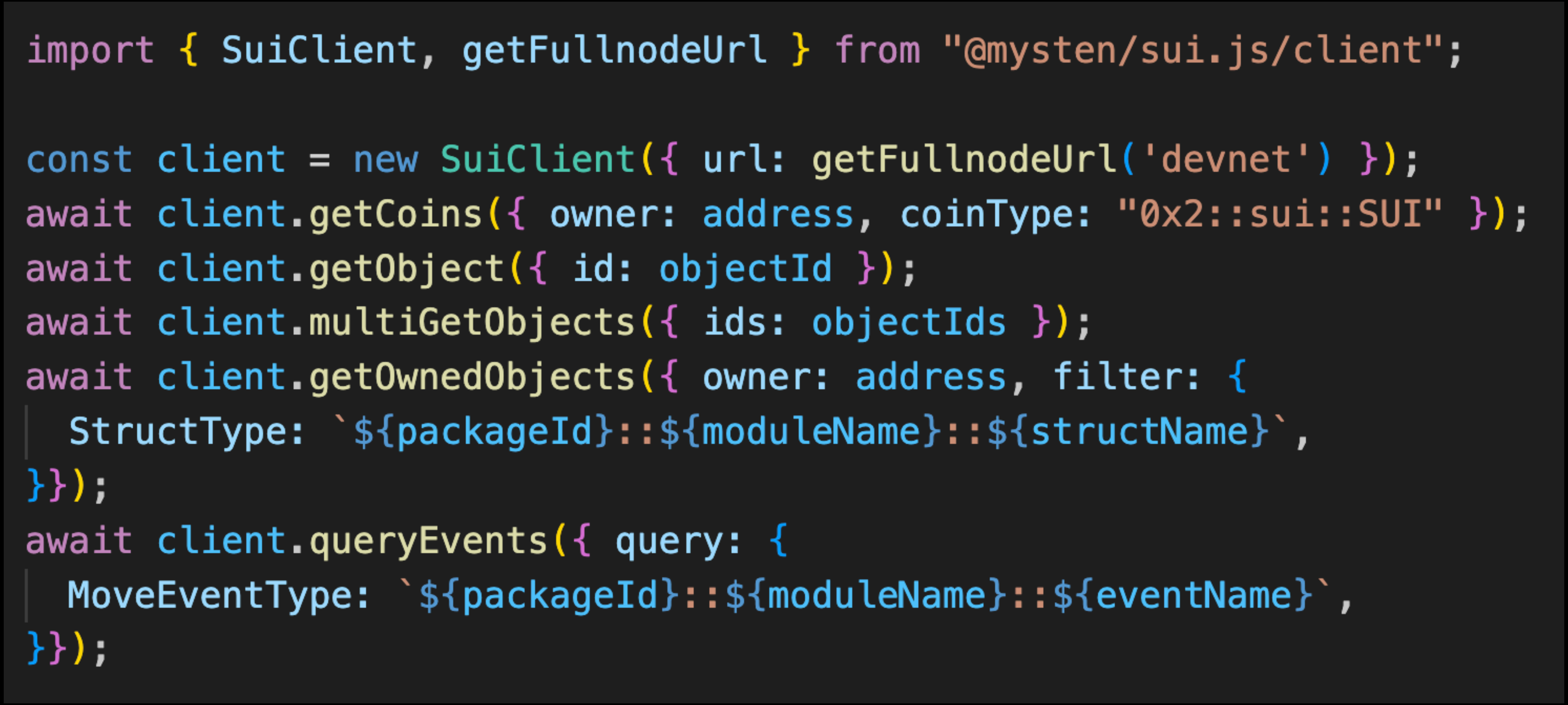
Bucket

Packages

- @mysten/sui.js

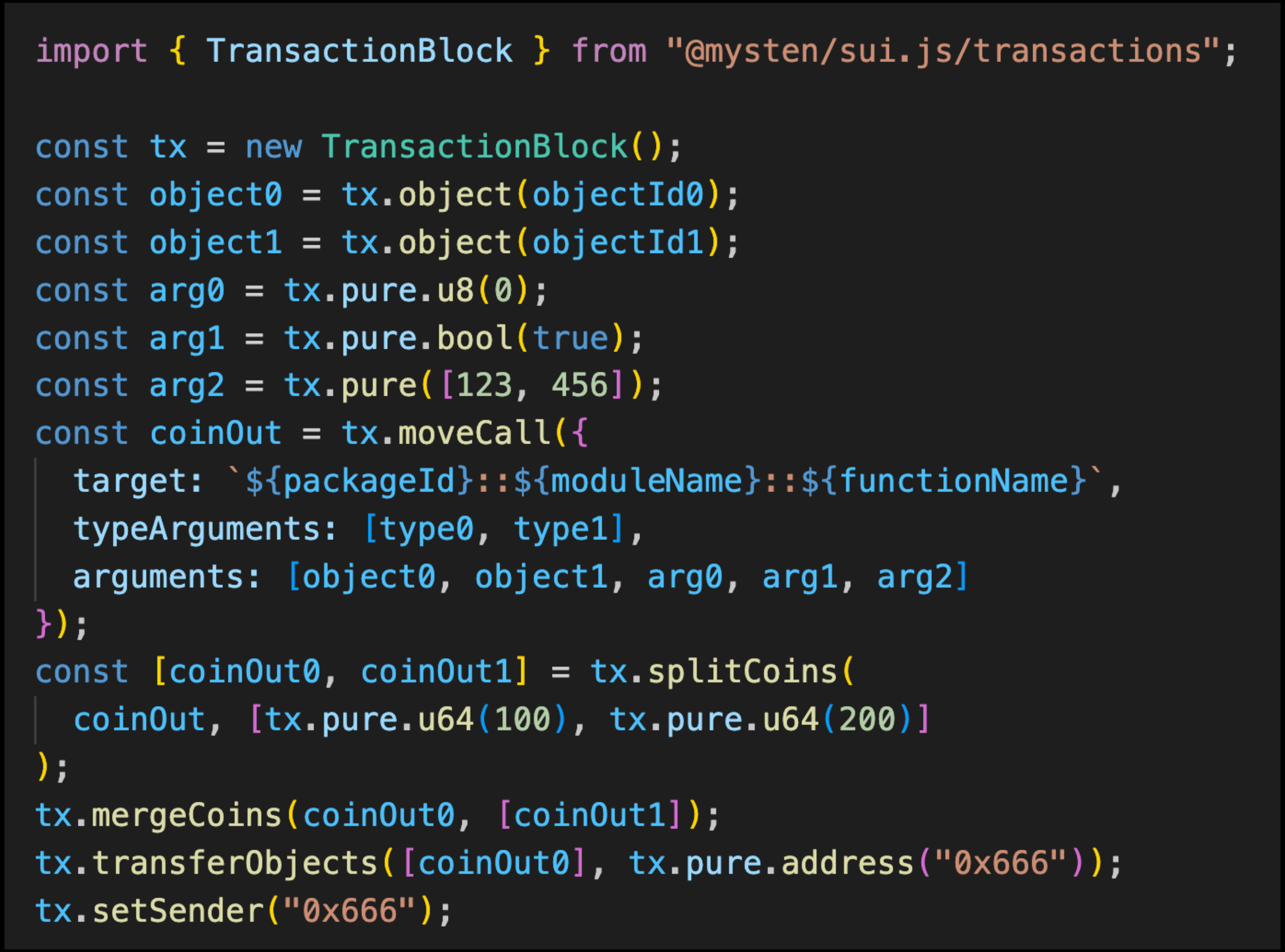
- @mysten/dapp-kit

Sui Client



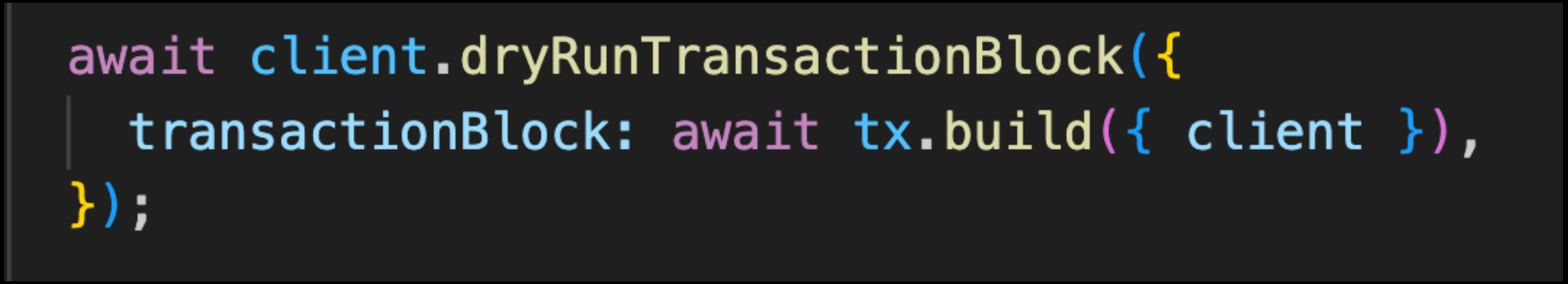
```
import { SuiClient, getFullnodeUrl } from "@mysten/sui.js/client";

const client = new SuiClient({ url: getFullnodeUrl('devnet') });
await client.getCoins({ owner: address, coinType: "0x2::sui::SUI" });
await client.getObject({ id: objectId });
await client.multiGetObjects({ ids: objectIds });
await client.getOwnedObjects({ owner: address, filter: {
  StructType: `${packageId}::${moduleName}::${structName}`,
}});
await client.queryEvents({ query: {
  MoveEventType: `${packageId}::${moduleName}::${eventName}`,
}});
```

Transaction Block



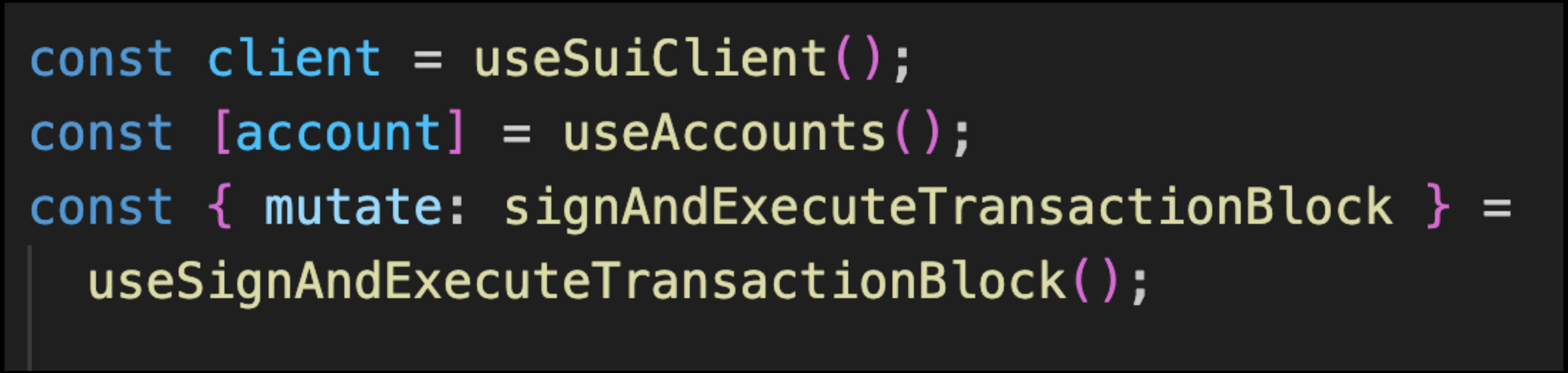
```
import { TransactionBlock } from "@mysten/sui.js/transactions";

const tx = new TransactionBlock();
const object0 = tx.object(objectId0);
const object1 = tx.object(objectId1);
const arg0 = tx.pure.u8(0);
const arg1 = tx.pure.bool(true);
const arg2 = tx.pure([123, 456]);
const coinOut = tx.moveCall({
  target: `${packageId}::${moduleName}::${functionName}`,
  typeArguments: [type0, type1],
  arguments: [object0, object1, arg0, arg1, arg2]
});
const [coinOut0, coinOut1] = tx.splitCoins(
  coinOut, [tx.pure.u64(100), tx.pure.u64(200)]
);
tx.mergeCoins(coinOut0, [coinOut1]);
tx.transferObjects([coinOut0], tx.pure.address("0x666"));
tx.setSender("0x666");
```

Dry Run



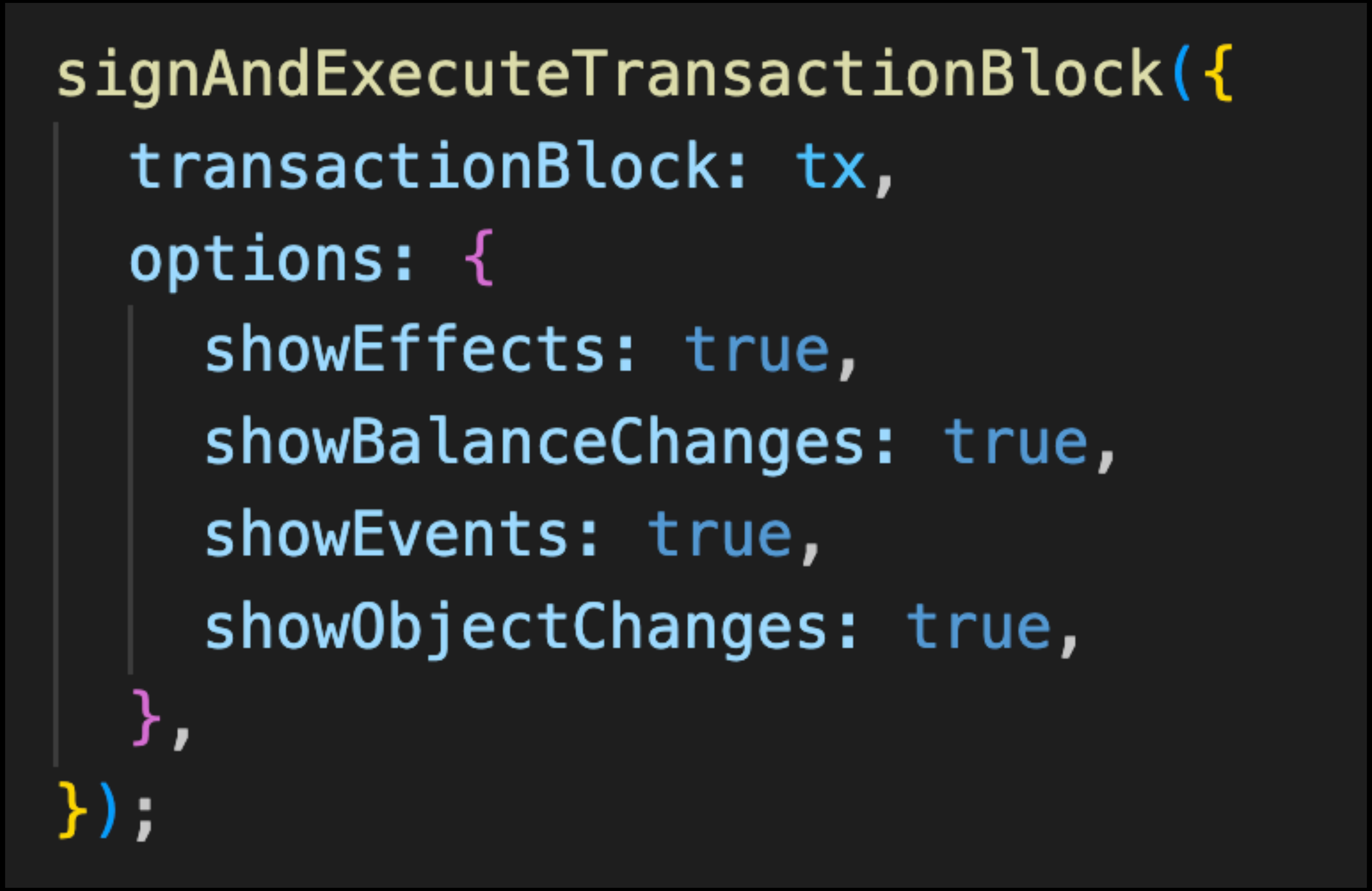
```
await client.dryRunTransactionBlock({
  transactionBlock: await tx.build({ client }),
});
```

Hooks



```
const client = useSuiClient();
const [account] = useAccounts();
const { mutate: signAndExecuteTransactionBlock } =
  useSignAndExecuteTransactionBlock();
```

Sign and Execute



```
signAndExecuteTransactionBlock({
  transactionBlock: tx,
  options: {
    showEffects: true,
    showBalanceChanges: true,
    showEvents: true,
    showObjectChanges: true,
  },
});
```