### 1、安装工具和准备工作

下载安装nodejs17

<https://nodejs.org/download/release/v17.9.1/node-v17.9.1-x64.msi>

安装yarn：

**npm install yarn -g**

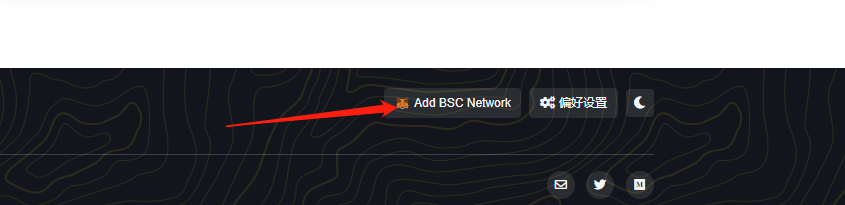
部署过程中实际提示缺少什么再安装什么

谷歌应用商店安装metamask

<https://chrome.google.com/webstore/detail/metamask/nkbihfbeogaeaoehlefnkodbefgpgknn>

之后打开币安链测试网，为metamask添加网络

<https://testnet.bscscan.com/>



导入以下钱包到metamask：

私钥：0xe4d30f19888cde4786cfc8900af6850864d63a26e125c5c9cdc7466cc024fb87

### 2、编译、部署合约

进入contract目录

查看hardhat.config.js配置，里面有一些相关信息

执行命令：**yarn**

执行编译命令：**npx hardhat compile**，成功后会提示编译的数量

部署工厂、WETH、Multicall、MockERC20合约：**npx hardhat run scripts/deploy1.js**，执行后需要等待几十秒，得到所有合约地址，**备份它**：

-----------------合约部署结果------------------

工厂合约: 0xDda6a3645BdDB0C11369B207eC504aC62b2da218

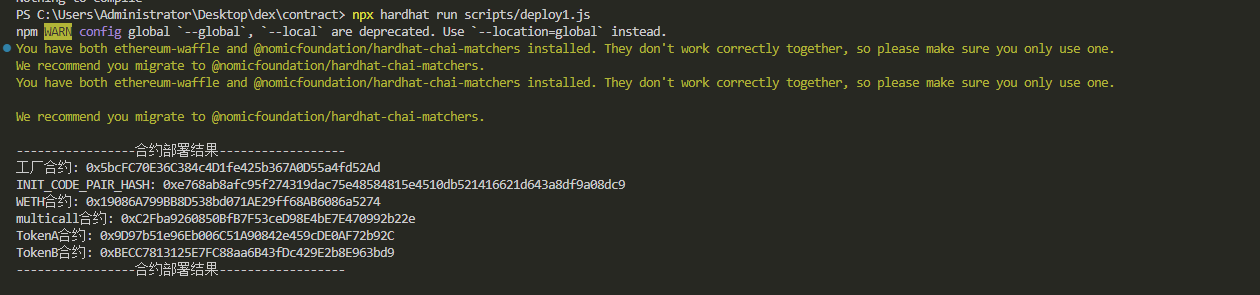
INIT\_CODE\_PAIR\_HASH: 0x07de0cb5b800b79646c0c07a5820e21692255238a43c6f35d00404ca614abcff

WETH合约: 0xF2C1A22AcEd2eDE0ede9fE130278Ec2d91edD817

multicall合约: 0x3079bc3911357086D1Cc9c612bdBF8cFC91009E4

TokenA合约: 0xE41353b14f5f79A15aCbFb59e4AECA619d25D7C4

TokenB合约: 0xfca31B1B19B0Ef144583A53217Cac84daFD8A204

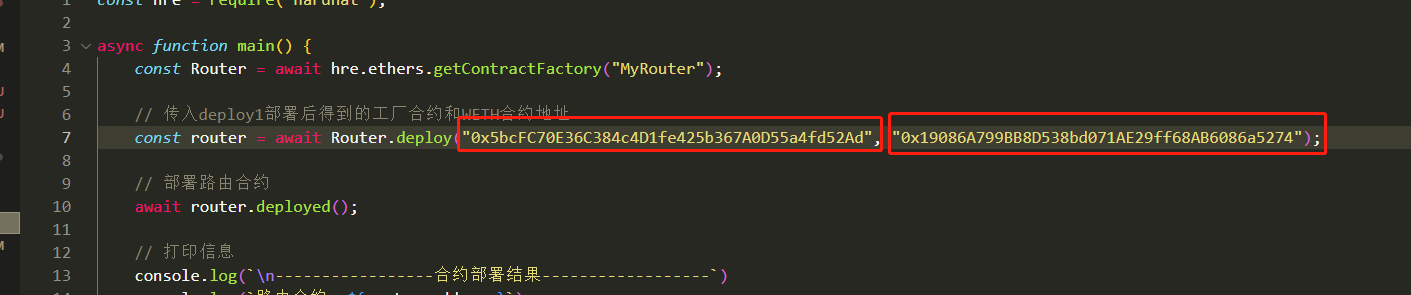
-----------------合约部署结果------------------

**修改contracts/MyRouter.sol路由合约文件init code hash（697行）：将单引号内的字符替换为上一步得到的INIT\_CODE\_PAIR\_HASH**：e768ab8afc95f274319dac75e48584815e4510db521416621d643a8df9a08dc9（注意**去掉0x前缀**），记得保存



再次编译合约：**npx hardhat compile**，提示成功编译1个

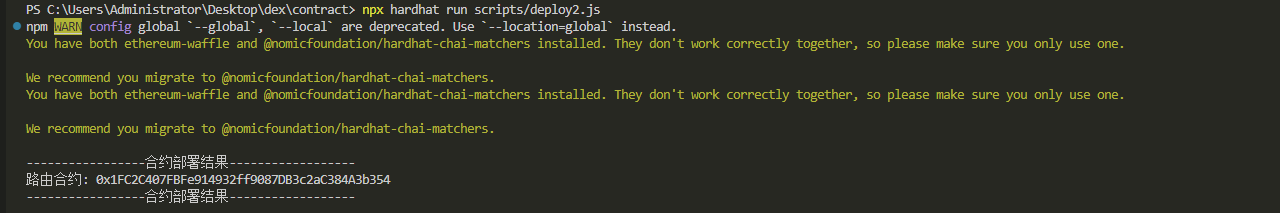
修改scripts/deploy2.js文件第7行，填入2个参数，分别是上一步得到的**工厂合约地址**和**WETH合约地址**，记得保存



部署路由合约：**npx hardhat run scripts/deploy2.js**，需要等待几十秒，得到路由合约地址，**备份它**：

-----------------合约部署结果------------------

路由合约: 0xceBf2F2542E651CfD27fc4453458a9c4B2e01523

-----------------合约部署结果------------------

### 3、注册npmjs账号

地址：<https://www.npmjs.com/signup>

这步我就不演示了 就是注册一个账号

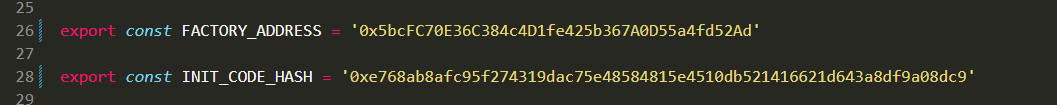
合约已经部署完了，接下来修改sdk

### 4、发布dex-sdk包

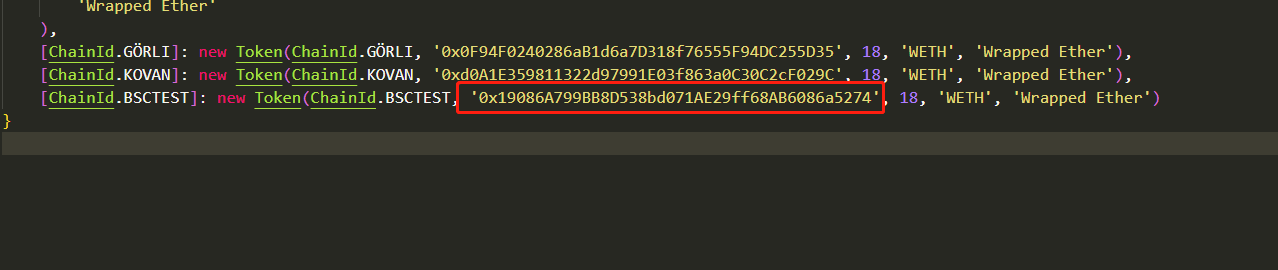
进入dex-sdk目录

执行命令yarn

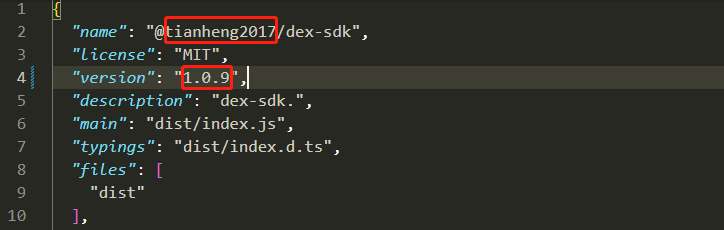
修改src/constants.ts的FACTORY\_ADDRESS和INIT\_CODE\_HASH为上一步得到的工厂合约与INIT\_CODE\_PAIR\_HASH值，保存



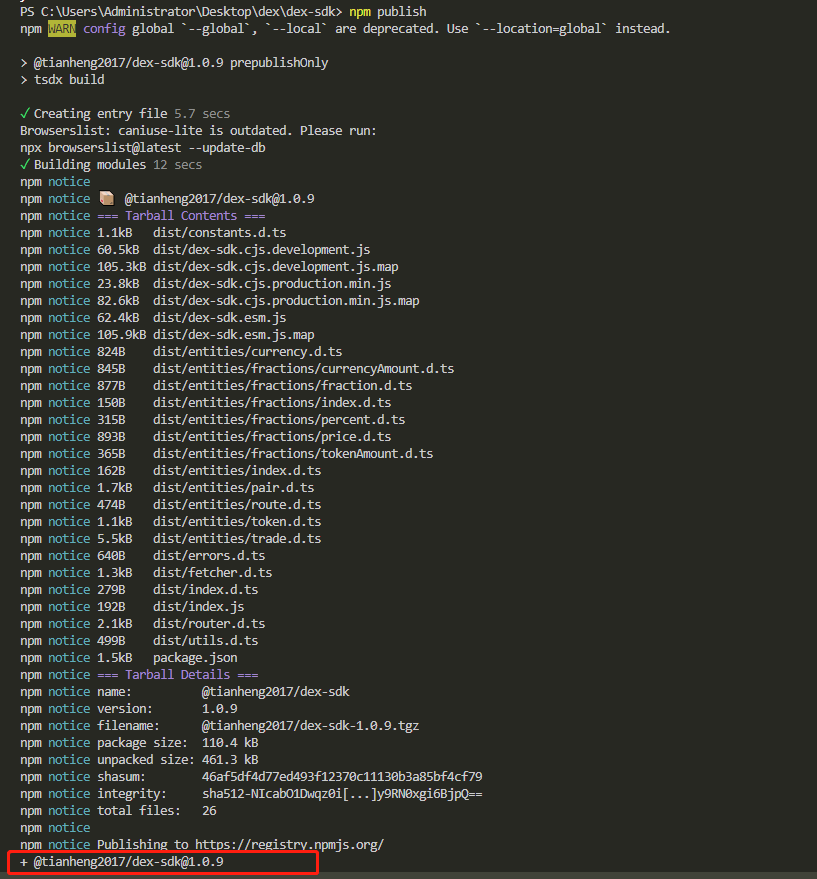
修改src/entities/token.ts的BSC的WETH地址为上一步得到的WETH合约地址



修改package.json的name和version，将name中的tianheng2017替换为为您的npmjs账号，修改version为您的版本号，新账号可以从1.0.0开始，之后每次发布包需要**递增**



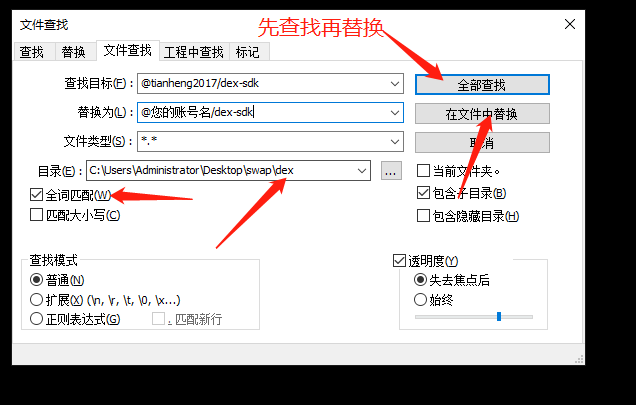
发布npm包：**npm publish –access public**



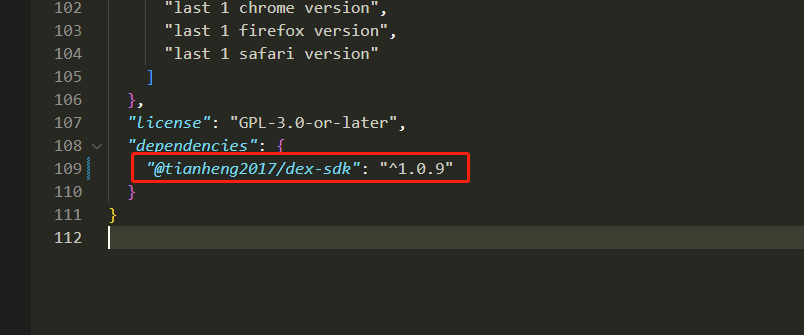
### 替换dex目录的所有npm包名

这步我不演示具体了，大概演示下

下载notepad++，进入notepad++，ctrl+h弹出替换界面，限定路径为dex目录，全局替换@tianheng2017/dex-sdk为你的npm信息，比如@xxxx/dex-sdk



进入dex目录，修改package.json，"@你的npx信息/dex-sdk": "^您的版本号"，如"@tianheng2017/dex-sdk": "^1.0.9"，保存

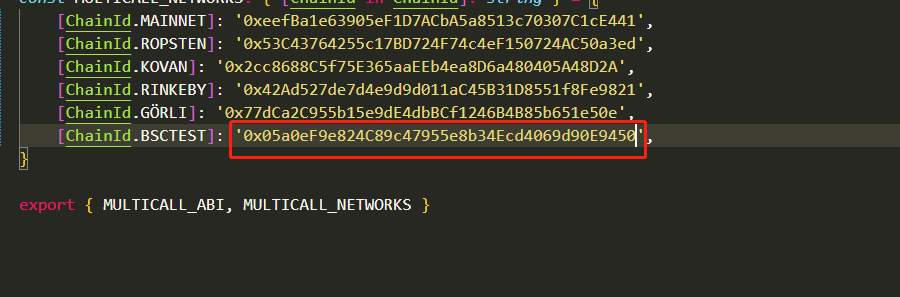


6、运行前端

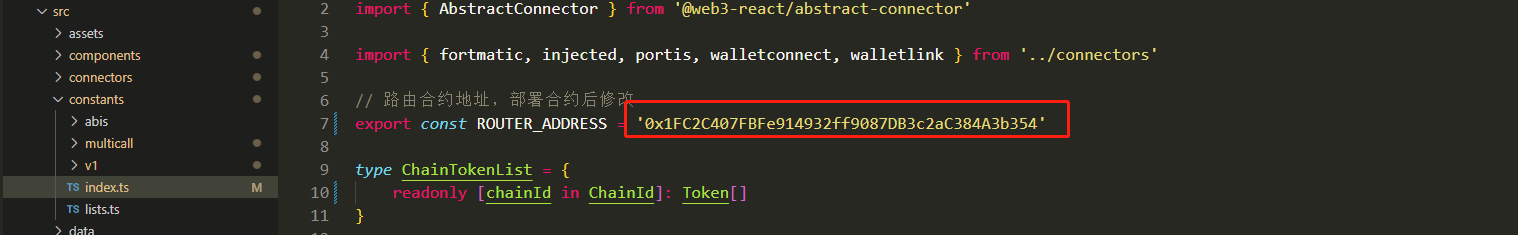
进入dex目录

执行命令yarn

修改文件multicall：dex\src\constants\multicall\index.ts，为上一步得到的multicall地址，保存

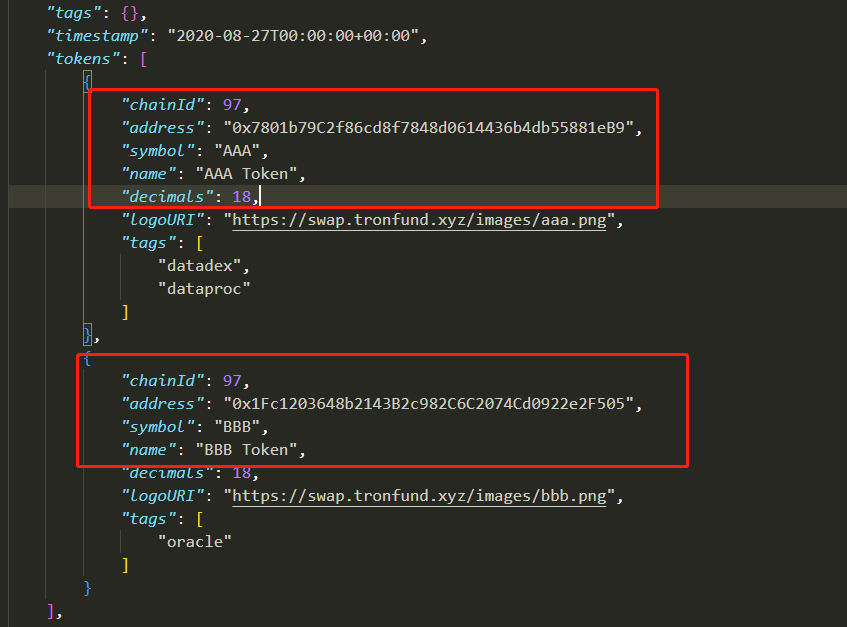


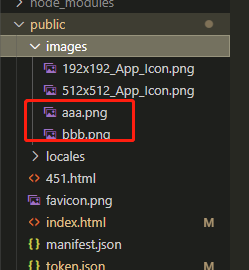
修改文件dex\src\constants\index.ts，将路由合约修改为上一步得到的路由合约地址，保存

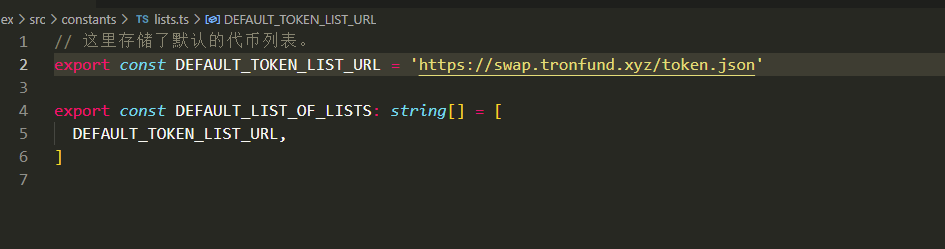


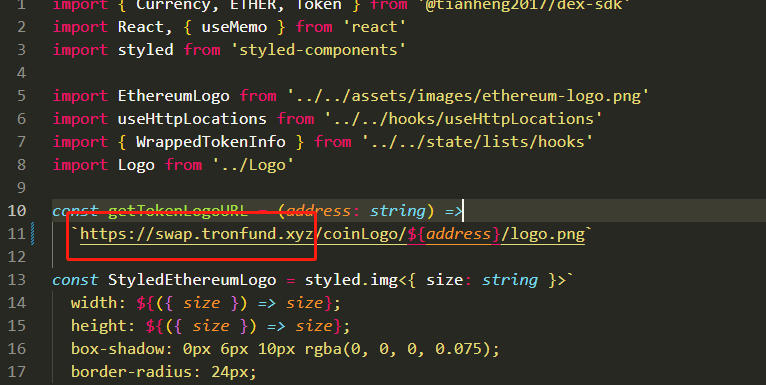
修改public/token.json，分别是上一步得到的AAA和BBB代币信息

且图标放到dex\public\images下





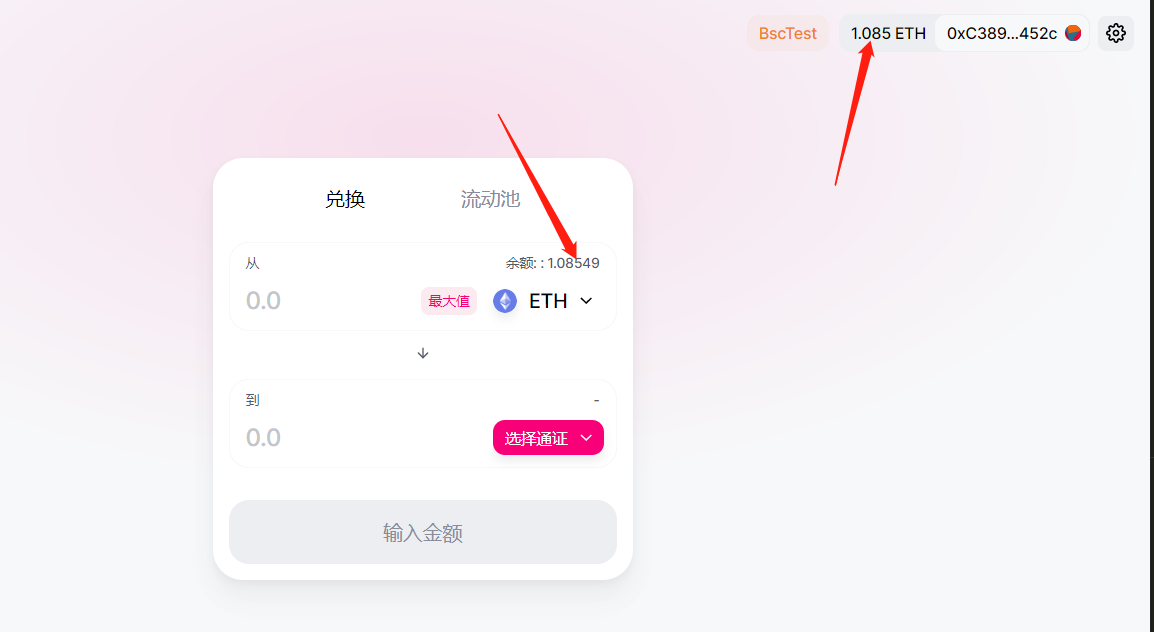
修改dex\src\constants\lists.ts的json地址指向token.json(我这里是线上web)，里面包含了AAA和BBB，这里储存了默认的代币列表，如图

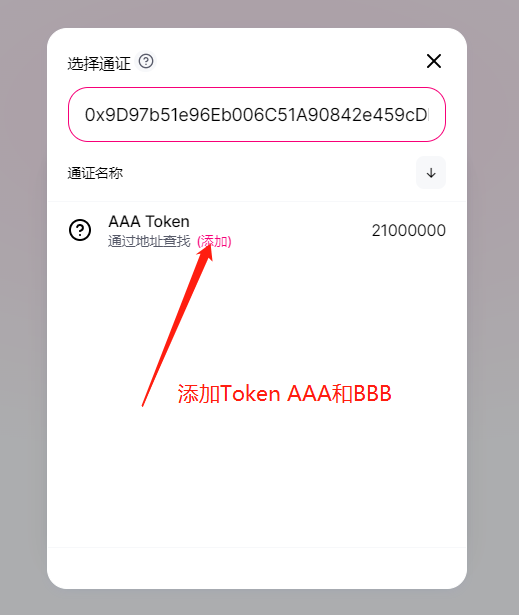
修改域名：dex\src\components\CurrencyLogo\index.tsx，否者仓位交易对要显示为？？

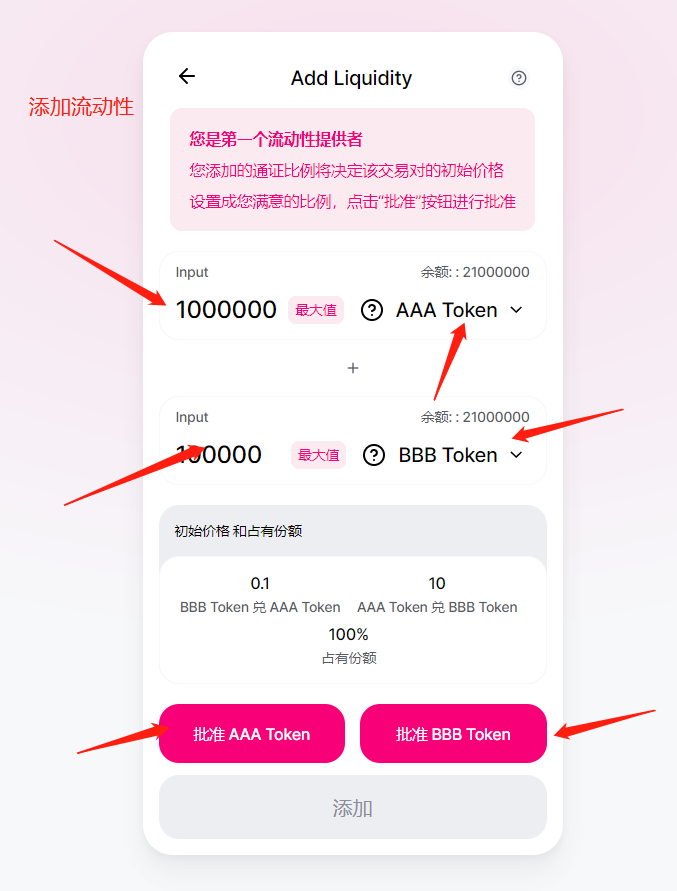
最后执行本地运行命令：yarn start

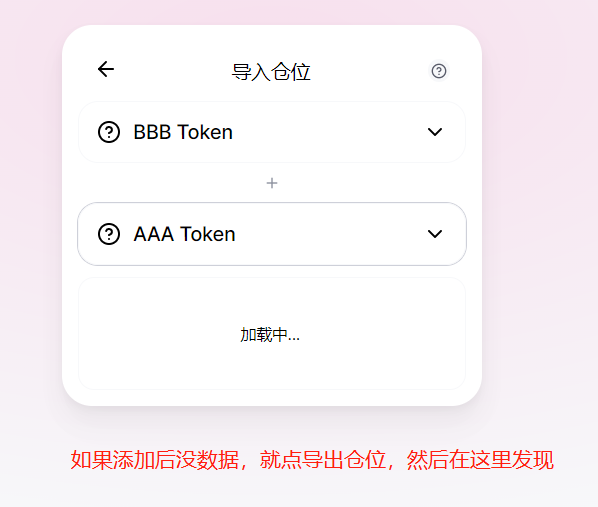
或者打包后到传到web服务器：yarn build

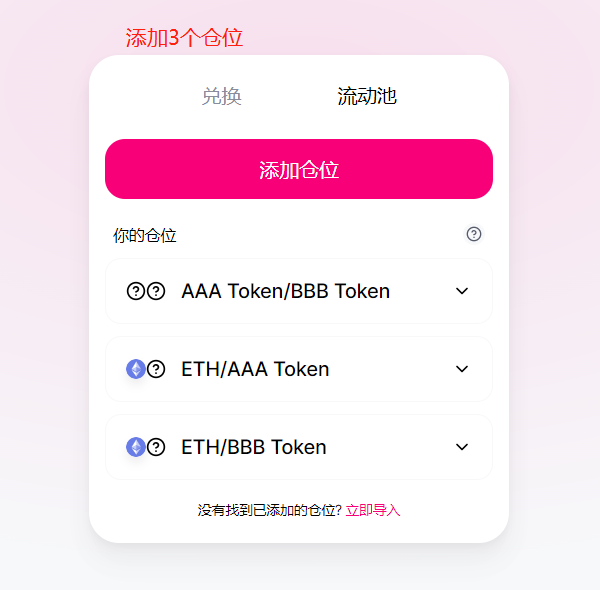
### 添加流动性和兑换测试

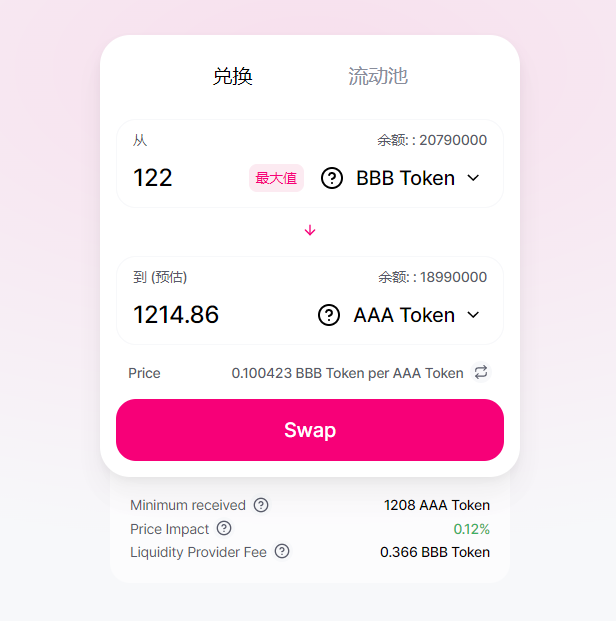














### BSC测试网代币领取

登陆BSC测试网水龙头：<https://testnet.bnbchain.org/faucet-smart>

每个钱包每天可以领0.1个，如果不够要想领很多，就要使用多个钱包（文件有提供100个测试的） + VPN切换IP + 谷歌浏览器无痕模式，领取后转账汇总到主要钱包

VPN：http://miao.sdmone.pro/auth/login