**Dex实战课程**

* 1. 基础篇
  + 问题1：如何创建交易？
    - 创建你的第一笔交易？
      * 钱包
      * 代币
      * 购买
      * 签名
      * 上链
  + 问题2：钱包在交易中的角色？
    - 任务
      * 如何创建一个钱包？
      * 如何用钱包进行交易？
      * 交易做了什么事情？
      * 交易发到哪里去了？
    - 涉及概念
      * web3的入口
      * 账户
        + 公
      * 签名
        + 私钥

椭圆曲线加密

* + - * 余额
  + 问题3：代币是什么？
    - 类比
    - spl token
  + 问题4: 如何查看我的交易？
    - 区块链浏览器
      * 下节课的钩子
        + 怎么这么多地址？都代表什么意思?
  + 问题5：地址和账户的关系？
    - 账户模型
      * hook
        + 学完了账户模型，我们来看owner到底什么意思？它有什么作用？
  + 问题6：钱包账户和代币的关系？
    - 持有关系怎么表示
      * ATA账户
  + 问题7：如何发送交易到链上？
    - 节点：测试网，开发网，主网
      * hook
  + 问题8：合约是什么？在我们的交易中，它起了什么作用？
    - 合约和传统类的类比
    - 代币合约，system合约，dex合约等等
  + 问题9： 如何创建并部署合约？
  + 问题10： 如何与合约交互？
    - IDL 与合约的关系？
      * hook
        + 基于IDL我们能做什么？
  + 问题11： 交易上链后发生了什么？
    - 交易状态
      * hook
        + processing+confirmed+finalized 状态变化发生了什么？
  + 问题12： 区块链？什么是区块？链接的什么？
    - 重点：交易上链上的哪里去了？ 交易和区块的关系？
  + 问题13： 交易的三种状态：被谁process了？处理了什么？什么时候被处理？
    - hook
      * 处理之后呢？
  + 问题14：交易状态2：confirmed: when, what, who,how...
    - 共识系统
      * 达成共识了confirmed 就安全了吗？
      * 对什么达成了共识？
  + 问题15： 交易状态3：finalized （wehn, what,who,how, why?）
    - 区块重置攻击
      * pos,pow原理
  + 问题16：语言基础---golang在项目中用到的核心特性：channel,goroutine 及GMP并发模型讲解
    - 并发管理
* 2. 数据解析篇
  + 问题16：交易解析是做什么的？
    - 应用场景
      * 跟单
      * 狙击
      * 套利
      * 找工作
      * 攻击
        + 实时冻结攻击
        + 钓鱼攻击
        + ...
      * ...
  + 问题17：要解析什么？
    - 看你的需求，交易结构里有的你又可以解
  + 问题18：数据源从哪来？
    - 官方数据接口
    - 第三方数据服务商
  + 问题20：区块中交易怎么解析？
    - 遍历交易
  + 问题21： 交易数据结构
    - hook： 指令数据结构
  + 问题22：指令数据结构
  + 问题23： 如何解析指令
    - 分账户 和 data
  + 问题24：账户放入交易前后的变化？
  + 问题25：IDL文件从哪里获取？
  + 问题26： 如何查看data布局？
    - IDL文件
  + 问题27：如何实时计算代币价格？
    - 链上没有的数据，如何基于链上有的信息，进行聚合处理，来获取？
  + 问题28：如何获取代币的holder数？
    - 账户模型
  + 问题29：如何检查代币的安全性？
    - 代币数据结构
  + 问题30：如何解析pumpfun / AMM事件？
    - swap: buy / sell
    - pool creation
  + 问题31：如何解析raydium / launchpad事件?
    - swap: buy /sell
    - pool creation
  + 问题32： 如何解析 token/token2022 事件？
    - mint
    - burn
  + 问题：如何计算top10？
  + 问题33： 如何解析代币元数据？
  + 问题：如何获取用户持仓？
* 3. 交易构造篇
  + 问题34：交易是什么？
    - 构造交易就是构造指令
  + 问题35：回顾交易浏览器，我们一笔交易涉及哪些指令？
  + 问题36：computeunit/compunitprice指令作用？
  + 问题37：ata指令作用？
  + 问题38：sync指令作用？
  + 问题39： transfer指令作用？
  + 问题40：swap指令作用？
  + 问题41：close指令作用？
    - 账户生命周期
  + 问题42：system program合约作用？
  + 问题43：token program合约作用？
  + 问题44：custom program合约作用？
  + 问题45：什么是交易对？
  + 问题46：什么是AMM？
  + 问题47：什么是流动性？
  + 问题48：什么是滑点？
  + 问题49：如何计算最小输出数量？
  + 问题50：什么是CLMM？与AMM区别？
  + 问题51：如何在开发网部署pumpfun合约？
  + 问题52：如何在开发网部署raydium合约?
  + 问题53：什么是PDA账户?
  + 问题：如何实现交易构造并发布上链？
* 7. 项目综合篇
  + 问题：go-zero框架如何搭建？
  + 问题：consumer服务核心点？
  + 问题：trade服务核心点？
  + 问题：market服务核心点？
  + 问题：gateway服务核心点？
  + 问题：data-flow服务核心点？
  + 问题：websocket服务核心点？
  + 问题：前端实现核心点？
  + 问题：kafka服务器部署及在项目中的核心点？
    - kafka partition
  + 问题：redis服务器部署及其在项目中的角色与应用？
  + 问题：mysql服务器部署及其在项目中的角色与应用？
  + 问题：grpc在项目中的角色与应用？
* 6. 核心原理篇
  + 【涉及源码】和基础一下区分
  + 问题：什么是拜占庭攻击？
  + 问题：什么是POW？
  + 问题：什么是POS?
  + 问题：POH在优化solana共识速度中的作用？
  + 问题：为什么区块一旦finalized后，很难被重置？
  + 问题：solana最新共识
  + 问题：solana 虚拟机是什么？
* 5. 性能优化篇
  + 问题：如何保障交易成功率？
  + 问题：如何提高交易上链速度？
    - 节点
    - computelimit
    - priority fee
  + 问题：如何保障购买代币安全？
  + 问题：如何优化上亿行的大表？
  + 问题：如何提高查询速度？
* 4. 核心业务篇
  + 问题54：如何保障解析的实时性？
    - 生产者，消费者协程
    - block表如何设计?
  + 问题55：解析失败了怎么办？不久丢块了吗？
    - 兜底策略
  + 问题59：如何推送实时数据给前端展示？
    - redis 发布订阅模式
    - websocket推送
  + 问题60：限价单如何设计?
  + 问题61：翻倍出本如何设计？
  + 问题62：市价单如何设计？
  + 问题63：如何判断当前交易已经成功上链？
    - trade表
  + 问题64：kline如何计算？
  + 问题：表结构设计
    - token表如何设计？
    - pair表如何设计？
    - trade表
* 问题19：怎么解析区块？
  + sdk
* 自由主题