

---

# Read the Docs Template Documentation

发行版本 *1.0.0*

**Read the Docs**

2022 年 11 月 25 日



---

## 第一部分

---

<b>1 中国大陆读者:</b>	<b>3</b>
<b>2 我的社交媒体:</b>	<b>5</b>
2.1 Opensea Solidity 合约源码概览 . . . . .	5
2.2 Opensea Solidity Seaport 合约源码解读 . . . . .	7
2.3 网页功能 . . . . .	14
<b>3 索引和搜索</b>	<b>21</b>



你当前看到的分享，是关于 OpenSea Smart Contracts 的源码解读。如果有发现错误和错别字，欢迎在 Github 上修改提交。



# CHAPTER 1

---

中国大陆读者:

---

如果你在中国大陆，欢迎问我的个人网站：阿西河 <https://www.axihe.com/>，那里可以免翻墙浏览本文档，阅读更流畅。还有更多的 Solidity 文章分享。





## CHAPTER 2

---

我的社交媒体:

---

- 我的推特: @anbang\_account (♥ 欢迎关注)
- Discord 合约交流群: Solidity 智能合约交流
- Youtube: Anbang 的 Youtube 频道
- Bilibili: Anbang 的 Bilibili 频道

文档目录:

### 2.1 Opensea Solidity 合约源码概览

这里介绍 Opensea 的源码分析

#### 2.1.1 媒体资料

- 官网: <https://opensea.io/>
- Github: <https://github.com/ProjectOpenSea>
- Github 上所有的合约相关仓库
  - Seaport: <https://github.com/ProjectOpenSea/seaport>
  - operator-filter-registry: <https://github.com/ProjectOpenSea/operator-filter-registry>
  - seadrop: <https://github.com/ProjectOpenSea/seadrop>

- seaport-order-validator: <https://github.com/ProjectOpenSea/seaport-order-validator>
- 0x-fee-wrapper: <https://github.com/ProjectOpenSea/0x-fee-wrapper>
- meta-transactions: <https://github.com/ProjectOpenSea/meta-transactions>
- marketplace-benchmarks: <https://github.com/ProjectOpenSea/marketplace-benchmarks>
- 官方说明文档:
  - Seaport: <https://docs.opensea.io/v2.0/reference/seaport-overview>

## 2.1.2 Seaport 核心合约的资料

- 仓库: <https://github.com/ProjectOpenSea/seaport>
- 文档: <https://github.com/ProjectOpenSea/seaport/blob/main/docs/SeaportDocumentation.md>
- 接口: <https://github.com/ProjectOpenSea/seaport/blob/main/contracts/interfaces/SeaportInterface.sol>
- 完整接口文档: <https://docs.opensea.io/v2.0/reference/seaport-overview>
- 已部署的合约地址:
  - Seaport 1.1: [0x00000000006c3852cbEf3e08E8dF289169EdE581](#)
  - ConduitController: [0x00000000F9490004C11Cef243f5400493c00Ad63](#)
- 部署文档:
  - <https://github.com/ProjectOpenSea/seaport/blob/main/docs/Deployment.md>
- 流程图:
  - 精简版
  - 完整版
- 审计报告:
  - <https://github.com/trailofbits/publications/blob/master/reviews/SeaportProtocol.pdf>
- 相关解读
  - <https://www.theblockbeats.info/news/30718>
  - <https://www.panewslab.com/zh/articledetails/sk465e369t44.html>

## 2.2 Opensea Solidity Seaport 合约源码解读

### 2.2.1 特点

- 买家可以用不同的资产买 NFT
  - ETH/ERC20/ERC721/ERC1155 资产
- 交易特定的 NFT
  - 当交易 NFT 时，也可以设置 NFT 必须具备的特定“条件”。
- 荷兰拍
  - 以荷兰拍的方式进行拍卖

### 2.2.2 核心业务

1. NFT 订单
2. 订单执行
3. 检查余额和批准交易
4. 部分成交
5. 业务关键步骤

#### NFT 订单

每一个订单都包含 11 个关键组件：

```

orderType    订单类型                => 根据两个不同的偏好，指定订单的四种类型之一
            FULL                    : 不支持部分填充
            PARTIAL                  : 允许
-> 允许填充订单中的一部分，注意每个代币必须被提供的分数完全整除（即除法后没有余数）
            OPEN                    : 任意账户都可以提交执行订单的调用
            RESTRICTED              : 需要订单必须由报价者或订单所在区域执行
                                   或者在区域上调用 isValidOrder 或 isValidOrderIncludingExtraData
-> isValidOrderIncludingExtraData
                                   视图函数时返回表示订单被批准的神奇的值。

offer        报价                    => 可以
-> 可以从报价者账户转移的一系列代币，其中每个代币由以下组件组成
    - itemType                : 指定代币类型 有效类型包括：
                                Ether
                                ERC20 / ERC721 / ERC1155
  
```

(续下页)

(接上页)

有条件(criteria)的 ERC721  
 有条件(criteria)的 ERC1155

- token : 指定代币合约的账户地址, 空地址用以代币。
- identifierOrCriteria : ERC721 或 ERC1155

→ 代币标识符, 或者在基于条件的代币类型的情况下,  
 表示由代币的有效代币标识符集合组成的 merkle 根。对于  
 → Ether 和 ERC20 类型, 该值会被忽略,  
 并且对于基于条件的代币类型, 可以将值设置为 0

→ 以允许任何标识符。

- startAmount : 报价生效的开始代币数量
- endAmount : 表示如果在订单到期时执行订单所需要的相关代币的数量。  
 如果此值与 startAmount

→ 不同, 则根据订单激活后经历的时间线性计算出实际的数量。

startTime 开始时间: => 荷兰拍的开始时间

endTime 结束时间: => 荷兰拍的结束时间

→ 该值与 `startTime` 与每个代币的 `startAmount` 和 `endAmount` 一起使用以得出它们的当前数量。

offerer 报价者

zone 区域 => 该区域可以通过调用 `cancel` 来取消命名为该区域的订单  
 报价者仍可以取消他们自己的订单, 可以单独取消,  
 也可以通过调用 `incrementNonce` 立刻取消由其当前 nonce

→ 签名的所有订单

→ 受限订单必须由区域或报价者执行, 或者必须通过调用区域上的 `isValidOrder`  
 或 `isValidOrderIncludingExtraData` 视图函数来获得批准。

zoneHash 区域哈希 =>

→ 当执行受限订单时, 该值将提供给区域, 该区域在确定是否是授权订单时可以使用该值。

consideration =>

→ 包含为完成订单而必须接收的代币数组。它包含所有与所提供代币相同的组件,  
 → 并且还包括一个用于接收每个代币的 `recipient` 组件。该数组可以由执行者在订单执行时进行扩展,  
 以支持“小费”(例如中继费或推荐费)。

conduitKey =>

→ 在执行转移时应将哪个渠道(conduit) (如果有) 用作代币批准的来源。

→ 默认情况下 (即当 `conduitKey` 设置为零哈希时), 报价方将直接向 Seaport

(续下页)

(接上页)

授予 ERC20、ERC721 和 ERC1155。

→ 代币批准，以便它可以在执行期间执行订单指定的任何转移。

→ 相反，选择使用渠道的报价者将授予与提供的渠道密钥相对应的渠道合约的代币批准，  
然后 Seaport 指示该渠道转移相应的代币。

salt                  Salt                  =&gt; 订单的任意熵源

nonce                nonce                =&gt; 必须与给定报价者的当前随机数匹配的值

## 订单执行

订单通过以下 4 种方式中的一种来执行：

- 调用两个“标准”函数 `fulfillOrder` 和 `fulfillAdvancedOrder` 中的一个，并且构造第二个隐含订单，同时其调用者作为报价者 (`offerer`)，已执行订单的对价 (`consideration`) 作为报价 (`offer`)，已执行订单的报价作为对价（使用“高级”订单包含应与一组“条件解析器”一起填写的部分，这些“条件解析器”为已执行订单上的每个基于条件的代币指定一个标识符和相应的包含证明）。所有报价代币将从订单报价者转移到执行者，然后所有对价代币将从执行者转移到指定的接收者。
- 调用“基本”函数 `fulfillBasicOrder`，并提供六种基本路线类型（`ETH_TO_ERC721`、`ETH_TO_ERC1155`、`ERC20_TO_ERC721`、`ERC20_TO_ERC1155`、`ERC721_TO_ERC20` 以及 `ERC1155_TO_ERC20`）中的一种，将从组件子集派生要执行的订单，假设相关订单符合以下条件：
  - 该订单仅包含一个报价代币，并且包含至少一个对价 (`consideration`) 代币。
  - 该订单仅包含一个 ERC721 或 ERC1155 代币，并且该代币不是基于条件的。
  - 订单的报价者是第一个对价代币的接收者。
  - 所有其他代币都具有相同的以太币（或其他原生代币）或 ERC20 项目类型和代币。
  - 该订单不提供以以太币（或其他原生代币）作为其项目类型的项目。
  - 每个项目上的 `startAmount` 必须与该项目的 `endAmount` 匹配（即项目不能有升序/降序数量）。
  - 所有“忽略”的项目字段（即 `token` 和原生代币项目中的 `identifierOrCriteria` 以及 ERC20 项目中的 `identifierOrCriteria`）都设置为空地址或零。
  - 如果订单中有 ERC721 项目，则该项目的数量为 1
  - 如果订单有多个对价 (`consideration`) 项目，且除了第一个对价项目以外的所有对价项目与报价项目的项目类型相同，报价项目数量不小于除了第一个对价项目数量外的所有对价项目数量之和。
- 调用两个“可用执行”函数 (`fulfillAvailableOrders` 和 `fulfillAvailableAdvancedOrders`) 中的一个，并且提供一组订单与一组执行声明，其中的执行声明指定哪些报价项目可以聚合到不同的转移中，相应地哪些对价项目可以聚合在一起，以及其中已经取消的订单是因为时间无效，或者已经

完全成交的订单将被跳过，而不会导致其余可用订单回滚。此外，一旦锁定 `maximumFulfilled` 可用订单，剩余的所有订单都将被跳过。与标准执行函数类似，所有报价项目将从各自的报价者转移到执行者，然后所有对价项目将从执行者转移到指定的接收者。

- 调用两个“匹配”函数（`matchOrders` 和 `matchAdvancedOrders`）中的一个，并且提供一组明确的订单以及一组执行，该执行指定了哪些报价项目应用于哪些对价项目（并且“高级”案例以类似的方式运行标准方法，但支持通过提供的分子 `numerator` 和分母 `denominator` 小数值以及可选的 `extraData` 参数进行部分填充，当执行受限订单类型时，这些参数将作为调用区域上的 `isValidOrderIncludingExtraData` 视图函数的一部分提供）。请注意，以这种方式执行的订单没有明确的执行者；相反，**Seaport** 将简单地确保每个订单的需求一致。

虽然标准方法在技术上可用于执行任何订单，但在某些情况下存在关键的效率限制：

- 与简单的“热路径 (hot path)”的基本方法相比，它需要额外的调用数据。
- 它要求执行者批准每个对价项目，即使对价项目可以使用报价项目来执行（在执行 `ERC721` 或 `ERC1155` 项目提供 `ERC20` 项目并且还包含具有相同的考虑对价的订单时通常是这种情况用于支付费用的 `ERC20` 项目类型）。
- 它可能导致不必要的转移，而在“匹配”情况下，这些转移可以减少到更小的集合。

## 检查余额和批准交易

创建报价时，应检查以下要求以确保订单可以执行：

- 报价者应在所有报价项目中有足够的余额。
- 如果订单未指明使用渠道，则报价者应为所有提供的 `ERC20`、`ERC721` 和 `ERC1155` 项目的 **Seaport** 合约设置足够的批准。
- 如果订单确实指明了使用渠道，则报价者应为所有提供的 `ERC20`、`ERC721` 和 `ERC1155` 项目的相应渠道合约设置足够的批准。

执行基本订单时，需要检查以下要求以确保订单可以执行：

- 需要执行上述检查以确保报价者仍有足够的余额和批准。
- 执行者应该对所有对价项目有足够的余额，除了那些项目类型与订单提供的项目类型相匹配的项目——例如，如果执行的订单提供 `ERC20` 项目，并且要求向报价者提供 `ERC721` 项目并且向另一个接受者提供相同的 `ERC20` 项目，那么执行者需要拥有 `ERC721` 项目，但不需要拥有 `ERC20` 项目，因为它将来自报价者。
- 如果执行者不选择使用渠道，他们需要为已执行订单上所有的 `ERC20`、`ERC721` 和 `ERC1155` 对价项目设置足够的 **Seaport** 合约批准，项目类型与订单提供的项目类型匹配的 `ERC20` 项目除外。
- 如果执行者确实选择使用渠道，则他们需要为已执行订单上的所有 `ERC20`、`ERC721` 和 `ERC1155` 对价项目为其各自的渠道设置足够的批准，项目类型与订单提供的项目类型匹配的 `ERC20` 项目除外。
- 如果已执行的订单将以代币（或其他原生代币）指定为对价项目，则执行者必须能够将这些项目的总金额提供为 `msg.value`

执行标准订单时，需要检查以下要求以确保订单可以执行：

- 需要执行上述检查以确保报价者有足够的余额和批准。
- 在收到所有的报价项目后，执行者应该对所有的报价项目有足够的余额——例如，如果执行的订单提供了 ERC20 项目，并且需要向报价者提供 ERC721 项目，并且向另一个接收者提供相同的 ERC20 项目，其数量小于或等于提供的数量，执行者不需要拥有 ERC20 项目，因为它将最先从报价者处接收到。
- 如果执行者不选择使用渠道，他们需要为已执行订单上的所有 ERC20、ERC721 和 ERC1155 对价项目的 Seaport 合约设置足够的批准。
- 如果执行者确实选择使用渠道，则他们需要为已执行订单上的所有 ERC20、ERC721 和 ERC1155 对价项目其各自的渠道设置足够的批准。
- 如果已执行的订单将以代币（或其他原生代币）指定为对价项目，则执行者必须能够将这些项目的总数量提供为 `msg.value`

在执行一组匹配订单时，需要检查以下要求以确保订单可以执行：

- 作为执行的一部分执行，执行采购 ERC20、ERC721 或 ERC1155 项目的每个帐户必须在触发执行时在 Seaport 或指定的渠道上具有足够的余额和批准。请注意，先前的执行可能会为后续执行提供必要的余额。
- 涉及代币（或其他原生代币）的所有执行的总和必须以 `msg.value` 的形式提供。请注意，提供者和接收者是同一帐户的执行将从最终执行集中被过滤掉。

## 部分成交

在构建订单时，报价者可以选择通过设置适当的订单类型来启用部分成交。然后，支持部分执行的订单可以在相应订单的某一部分中执行，从而允许后续执行绕过签名验证。总结一下部分填充的几个关键点：

- 当创建支持部分成交的订单或确定这些订单要成交的部分时，订单上的所有项目（报价和对价）数量必须能被提供的部分项目数量完全整除（即除法后没有余数）。
- 如果要填写的所需部分会导致要填写的订单数量超过全部订单金额，则该部分将减少为剩余要填写的数量。这适用于部分填充尝试和完全填充尝试。如果不需要这种行为（即填充应该是“全部或无”），则执行者可以使用“基本”订单方法（如果可用）（这需要填写全部订单数量），或使用“匹配”订单方法，并明确提供一个要求收到全部所需金额的订单。
  - 举例来说：如果一个执行者尝试执行订单的 1/2，但另一个执行者首先执行订单的 3/4，则原始执行者最终将执行订单的 1/4。
- 如果部分可成交订单上的任一项目指定了不同的 `startAmount` 和 `endAmount`（例如，它们是递增数量或递减数量的项目），则在确定当前价格之前，该分数将应用于这两个数量。这确保了在构建订单时可以选择完全可分的金额，而不依赖于最终完成订单的时间。
- 部分成交可以与基于条件的项目进行组合，以支持构建提供或接收多个项目的订单，否则这些项目将无法部分成交（例如 ERC721 项目）。

- 举个例子：报价者可以创建一个部分可成交的订单，为给定集合中最多 10 个 ERC721 项目提供最多 10 个 ETH；然后，任何执行者都可以执行该订单的一部分，直到它被完全执行（或取消）。

## 业务关键步骤

### 5.1 执行订单

当通过 `fulfillOrder` 或 `fulfillAdvancedOrder` 来执行订单时：

1. 计算订单哈希值
  - 计算报价项目和对价项目的哈希值
  - 检索报价者的当前计数器
  - 计算订单哈希值
2. 执行初始化校验
  - 确保当前时间在订单有效时间内
  - 确保调用者对于当前订单类型是有效的；如果订单类型收到限制且调用者不是 `offerer` 或者 `zone`，调用 `zone` 判断订单是否有效
3. 检索并更新订单状态
  - 确保订单未被取消
  - 确保订单没有被全部执行
    - 如果订单是部分执行的，如有必要，减少提供的执行数量，以免订单被过度执行
  - 若订单签名尚未验证，则验证订单签名
  - 根据偏好 + 可用金额 (`preference + available amount`) 确定要执行的分数
  - 更新订单状态（已验证 + 执行分数）
4. 确定每个项目的金额
  - 比较初始金额 `startAmount` 和结束金额 `endAmount`
    - 若相等，将执行分数应用于该金额，确保结果是整数，然后使用该结果
    - 若不等，对这两个金额都应用执行分数，确保两个结果都是整数，然后根据当前时间找到这两个结果的现行拟合值
5. 应用条件解析器
  - 确保每一个条件解析器都应用于一个基于条件的订单项目
  - 如果项目具有一个非零的条件根值 (a non-zero criteria root)，确保为每个项目提供的标识符是有效的
  - 更新每个项目的类型和标识符



- 确保所有剩余的项目都不是基于条件的项目
- 6. 触发 `OrderFulfilled` 事件
  - 包括更新的项目（即在金额调整和条件解决之后）
- 7. 将报价项目（代币）由报价者转移到调用者
  - 使用渠道或 `Seaport` 直接获得批准，具体取决于订单的类型
- 8. 将对价项目（代币）有调用者转移到对应的接受者
  - 使用渠道或 `Seaport` 直接获得批准，具体取决于执行者声明的偏好

## 5.2 匹配订单

当通过 `matchOrders` 或者 `matchAdvancedOrders` 来匹配一组订单时，步骤 1 到 6 几乎完全相同，但针对每个提供的订单执行。从这里开始，执行与上面的标准执行不同：

1. 应用执行
  - 确保每次执行都涉及一个或多个报价项目和一个或多个对价项目，所有这些项目都具有相同的类型和代币，并且每个报价项目具有相同的批准源以及每个对价项目具有相同接受者
  - 将每个报价项目和对价项目的金额减少到零，并跟踪其总减少金额
  - 比较每个项目的总金额，并将剩余金额加回相应订单一侧（报价项目或对价项目）的第一个项目
  - 为每个成交返回一个执行
2. 扫描每个对价项目并确保没有一个对价项目仍然有非零的剩余金额
3. 作为每次执行的一部分进行转账
  - 根据原始订单类型，直接使用渠道或 `Seaport` 获得批准
  - 忽略 `to == from` 或 `amount == 0` 时的每次执行（注意：当前实现不执行最后一次优化）

### 2.2.3 接口

1. `fulfillBasicOrder`
2. `fulfillOrder` & `fulfillAdvancedOrder`
3. `fulfillAvailableOrders` & `fulfillAvailableAdvancedOrders`
4. `matchOrders` & `matchAdvancedOrders`

## 2.3 网页功能

- Create a Collection

### 2.3.1 Create a Collection

- 网址: <https://opensea.io/collection/create>
- 属性: 中心化操作
- 结果: <https://opensea.io/collection/anbang-collection>

Logo image	: 合集的LOGO 350 x 350	(核心显示)
Featured image	: 合集精选图 600 x 400	(主页/类别页/活动推广页)
Banner image	: Banner图 1400 x 350	(合集主页顶部Banner区域显示)
Name	: 合集的名字	
URL	: 合集的 URL	在 OpenSea 上自定义合集的 URL。只能包含小写字母、数字、 → 和 -
Description	: 合集的介绍	使用 Markdown 语法，一共 1000 个字符
Category	合集所属分类: Opensea 首页有 Browse by category, 可以按照分类浏览NFT合集 <a href="https://opensea.io/category/collectibles">https://opensea.io/category/collectibles</a>	
Links	<ul style="list-style-type: none"> <li>- 你的网站</li> <li>- Medium</li> <li>- Telegram</li> </ul>	
Creator fees	: 版税	→ 每次交易都会收一定比例的税，可以不设置；如果设置可以多地址瓜分 0xBB83214561E9690ff36639299494726944AA30Fc 1% 0x5dC61e930533136AbE35C1b442823259D6c07694 2%
Blockchain	: 选择合集所在的链	(Polygon)
Payment tokens	: 支持的Token列表	(ETH Polygon / Matic / USDC Polygon)
Display theme	: 显示主题	(合集内，NFT图片的显示方式)
Explicit & sensitive content	: 敏感内容标记	允许色情和敏感内容，但是不会被搜索出来；如果不标记，发现后会被删除

## 2.3.2 Edit My Collection

- 网址: <https://opensea.io/collection/xxxxxxx/edit>
- 属性: 中心化操作

Social Connections	: 链接社交媒体
- Twitter	
- Discord	
- Instagram	
Collaborators	: 添加合作者
- 可以在本合集上传NFT的地址	

## 2.3.3 Create New Item

- 网址: <https://opensea.io/collection/anbang-collection/assets/create>

## 2.3.4 属性页面

- 创作者收益: <https://opensea.io/collection/anbang-collection/payouts>

## 2.3.5 Sell

- Fixed Price : 固定价格
- Timed Auction : 拍卖

### Fixed Price 签名内容

只能设置下面两个属性

- Set a price: 1 Matic
- Set duration: 1 Month

下面是预览

More options		
Summary	上架是免费的。一经售出，将扣除以下费用。	
Listing price	1 MATIC	挂单价格
Service Fee	2.5%	服务费
Creator Fee	3%	创作者费用（常说的版税）
Total potential earnings	0.945 MATIC	预估的最终收益

签名消息如下

```

offerer:      0xBB83214561E9690ff36639299494726944AA30Fc
offer:
  0:
    itemType      :    3
    token          :    0xA604060890923Ff400e8c6f5290461A83AEDACec
    identifierOrCriteria :
→84814189278733858855919475402523828762082208136572947867894469165638202949633
    startAmount    :    1
    endAmount      :    1

consideration:
  0:      预期收益
    itemType:0
    token:0x0000000000000000000000000000000000000000000000000000000000000000
    identifierOrCriteria:0
    startAmount:945000000000000000
    endAmount:945000000000000000
    recipient:0xBB83214561E9690ff36639299494726944AA30Fc
  1:      平台收益
    itemType:0
    token:0x0000000000000000000000000000000000000000000000000000000000000000
    identifierOrCriteria:0
    startAmount:250000000000000000
    endAmount:250000000000000000
    recipient:0x0000a26b00c1F0DF003000390027140000fAa719
  2:      创作者1 的收益
    itemType:0
    token:0x0000000000000000000000000000000000000000000000000000000000000000
    identifierOrCriteria:0
    startAmount:100000000000000000
    endAmount:100000000000000000
    recipient:0xBB83214561E9690ff36639299494726944AA30Fc
  3:      创作者2 的收益
    itemType:0
    token:0x0000000000000000000000000000000000000000000000000000000000000000
    identifierOrCriteria:0
    startAmount:200000000000000000
    endAmount:200000000000000000
    recipient:0x5dC61e930533136AbE35C1b442823259D6c07694

startTime    : 1669364098
endTime      : 1671956098
orderType    : 1

```

(续下页)

(接上页)

```

zone      : 0x0000000000000000000000000000000000000000000000000000000000000000
zoneHash  : 0x0000000000000000000000000000000000000000000000000000000000000000
salt      :  
 24446860302761739304752683030156737591518664810215442929814746464000893001823
conduitKey : 0x0000007b02230091a7ed01230072f7006a004d60a8d4e71d599b8104250f0000
counter    : 0

```

## Timed Auction 拍卖

两种拍卖方式

- Sell to highest bidder : 卖给最高出价者
- Sell with declining price : 以下跌的价格出售 (荷兰拍)

### Sell to highest bidder

只能设置下面两个属性

- Set a price: 1 ETH
- Set duration: 7 Days

下面是预览

```

More options

Summary      上架是免费的。一经售出，将扣除以下费用。
Listing price 1 ETH      挂单价格
Service Fee   2.5%      服务费
Creator Fee   3%        创作者费用（常说的版税）

Total potential earnings 0.945 ETH      预估的最终收益

```

签名消息如下

```

offerer      :0xBB83214561E9690ff36639299494726944AA30Fc
offer:
  0:
    itemType:3
    token:0xA604060890923Ff400e8c6f5290461A83AEDACec
     
 identifierOrCriteria:84814189278733858855919475402523828762082208136572947867894469165638202949633
    startAmount:1

```

(续下页)

(接上页)

```

        endAmount:1
consideration:
  0:
    itemType:1
    token:0x7ceB23fD6bC0adD59E62ac25578270cFf1b9f619
    identifierOrCriteria:0
    startAmount:945000000000000000
    endAmount:945000000000000000
    recipient:0xBB83214561E9690ff36639299494726944AA30Fc
startTime:1669365228
endTime:1670574828
orderType:3
zone:0x110b2B128A9eD1be5Ef3232D8e4E41640dF5c2Cd
zoneHash:0x0000000000000000000000000000000000000000000000000000000000000000
salt:24446860302761739304752683030156737591518664810215442929817021064316783470848
conduitKey:0x0000007b02230091a7ed01230072f7006a004d60a8d4e71d599b8104250f0000
counter:0

```

## Sell with declining price

能设置下面 3 个属性

- Set a price:
  - Starting price: 10 ETH
  - Ending price: 1 ETH
- Set duration: 7 Days

下面是预览

More options

Summary 上架是免费的。一经售出，将扣除以下费用。

Listing price 10 ETH 挂单价格

Service Fee 2.5% 服务费

Creator Fee 3% 创作者费用（常说的版税）

Total potential earnings 9.45 ETH 预估的最终收益

前面消息如下

```

offerer:0xBB83214561E9690ff36639299494726944AA30Fc
offer:

```

(续下页)

(接上页)

```

0:
  itemType:3
  token:0xA604060890923Ff400e8c6f5290461A83AEDACec
  identifierOrCriteria:84814189278733858855919475402523828762082208136572947867894469165638202949633
  startAmount:1
  endAmount:1
consideration:
  0:
    itemType:1
    token:0x7ceB23fD6bC0adD59E62ac25578270cFf1b9f619
    identifierOrCriteria:0
    startAmount:9450000000000000000
    endAmount:9450000000000000000
    recipient:0xBB83214561E9690ff36639299494726944AA30Fc
  1:
    itemType:1
    token:0x7ceB23fD6bC0adD59E62ac25578270cFf1b9f619
    identifierOrCriteria:0
    startAmount:2500000000000000000
    endAmount:2500000000000000000
    recipient:0x0000a26b00c1F0DF003000390027140000fAa719
  2:
    itemType:1
    token:0x7ceB23fD6bC0adD59E62ac25578270cFf1b9f619
    identifierOrCriteria:0
    startAmount:1000000000000000000
    endAmount:1000000000000000000
    recipient:0xBB83214561E9690ff36639299494726944AA30Fc
  3:
    itemType:1
    token:0x7ceB23fD6bC0adD59E62ac25578270cFf1b9f619
    identifierOrCriteria:0
    startAmount:2000000000000000000
    endAmount:2000000000000000000
    recipient:0x5dC61e930533136AbE35C1b442823259D6c07694
startTime:1669365422
endTime:1669970222
orderType:1
zone:0x0000000000000000000000000000000000000000000000000000000000000000
zoneHash:0x0000000000000000000000000000000000000000000000000000000000000000
salt:24446860302761739304752683030156737591518664810215442929815699357387149085795
conduitKey:0x0000007b02230091a7ed01230072f7006a004d60a8d4e71d599b8104250f0000
counter:0

```

### 2.3.6 固定价格上链



## CHAPTER 3

---

### 索引和搜索

---

- `genindex`
- `search`