

# 한 번에 끝내는 블록체인 개발 A to Z

---

## Chapter 3

### Defi 기초 컨셉 구현

Chapter 3

Defi 기초 컨셉 구현

# 유니스왑v1 CPMM 구현(5)

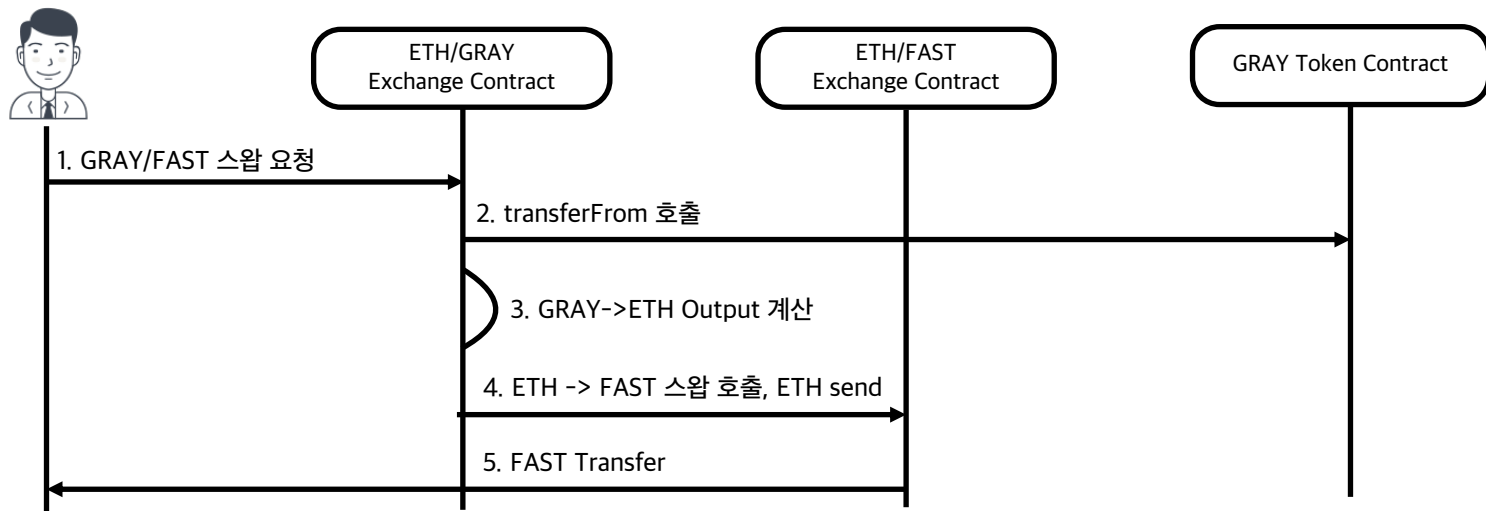
# Goal

---

CPMM 기반의 ERC20<->ERC20 스왑 기능을 구현한다.

# 스왑(ERC20<->ERC20)

유니스왑v1에서 ERC20<->ERC20을 위해서는 ERC20을 ETH로 바꾸고 그 ETH를 ERC20으로 바꾼다.



# Swap

```
function tokenToTokenSwap(  
    uint256 _tokenSold,  
    uint256 _minTokenBought,  
    address _tokenAddress) public {  
  
    address toTokenExchangeAddress = factory.getExchange(_tokenAddress);  
  
    //ERC20 -> ETH  
    uint256 ethOutputAmount = getOutputAmount(_tokenSold, token.balanceOf(address(this)), address(this).balance);  
    require(token.transferFrom(msg.sender, address(this), _tokenSold));  
  
    //ETH -> ERC20  
    IExchange(toTokenExchangeAddress).ethToTokenTransfer(value: ethOutputAmount)(_minTokenBought, msg.sender);  
}
```

1. 이전에 구현한 getOutputAmount 함수로 Input ERC20으로 얻게 되는 ETH양을 구한다.
2. Input ERC20/ETH 풀로 ERC20을 보낸다.
3. Output ERC20/ETH 풀에 ETH->ERC20 스왑을 호출 한다.

# 구현 및 테스트 (스왑)

---

- `git clone https://github.com/GrayWorld-io/lec_fc_defi`
  - `cd lec_fc_defi`
- `git reset --hard 87ee51b696e84b7f864544a13a887ad91c17de6e`

# 다음 강의

---

- 이번 클립으로 유니스왑v1 SmartContract 구현이 완료되었다.
- 실제 유니스왑 코드를 보면서 파라미터 유효성 체크 및 로그 기록 로직을 추가한다.