Dutch Auction

Steps:

S1 pragma
S2 interface IERC721
S3 contract
S4 variables
S5 constructor (Assigning initialising vars)
S6 functions

S1: Pragma

```
pragma solidity 0.8.26;
```

S2: Interface

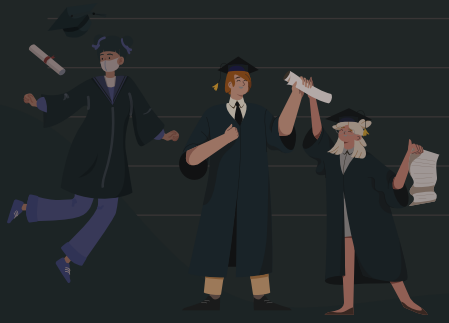
```
interface IERC721 {
    function transferFrom(
        address _from,
        address _to,
        address _nftId
    ) external;
}
```

S3: Contract

Create contract DutchAuction { }

mention DURATION = 7 days;

IERC721 nft; } public immutable
uint nftId; }



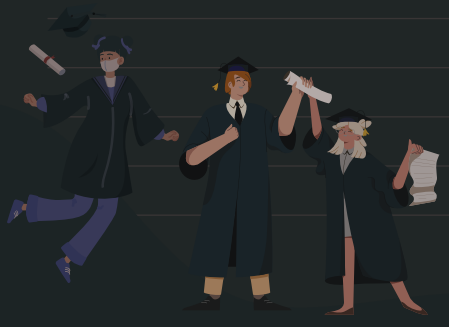
S4: Variables

seller, } address, payable, public
StartingPrice, } uint
StartAt, } public
EndAt, } immutable
DiscountRate }

S5: Constructor

```
constructor (
        uint _StartingPrice,
        uint _DiscountRate,
        address _nft,
        uint _nftId,
) {
    seller = payable(msg.sender);
    SP = _SP;
    SA = b.t;
    EA = b.t + DURATION;
    DR = _DR;

    require()
    nft = IERC721(_nft)
    nftId = _nftId;
}
```



S6: Functions

1> getPrice() public view returns (uint) {

```
timeElapsed = b.t. - SA;
discount = DR * timeElapsed;
return SP - discount;
```

2> buy() external payable {

```
require(b.t. < EA, "Auction Expired");

uint price = getPrice();

require(msg.value >= price, "ETH < price");
```