Simple MarketPlace

We are going to implement a marketplace to buy NFT (ERC721) with ERC20 tokens. We are going to need these functions:

1. Sell : you have items to sell and receive ERC20 payment. Items are put in escrow.
2. Buy : you have selected items to buy, pay in ERC20 and receive items.

We are going to start with the data we need:

```
struct Trade {
   address owner;
   uint256 item;
   uint256 price;
   uint256 status; // 0=Available, 1=Sold
}
```

Trade tracks the item available for sale. We keep track of the Trade data in a mapping called trades.

```
mapping(uint256 => Trade) public trades;
```

The mapping maps the trade id to the Trade data. The items available for sale are digital items and they are represented by ERC721 tokens.

So the contructor of our marketplace contract would need to obtain the token contract addresses:

```
constructor ( address _erc20, address _erc721) {
        currency = IERC20(_erc20);
         item = IERC721(_erc721);
         counter=0;
}
```

We need to implement the **sell** function:

```
function sell(uint256 _item, uint256 _price)   public {
 //transfer the item from seller to the contract as escrow
 items.transferFrom(msg.sender, address(this), _item);
 //keep track of trade
 trades[counter] = Trade({
                owner: msg.sender,
                item: _item,
                price: _price,
                status: 0
 });
 counter += 1; //trade id
}
```

The **buy** function:

```
function buy(uint256 _trade)   public {
  //retrieve the item for sale
  Trade memory trade = trades[_trade];
  require(trade.status == 0, "Trade is not available.");

  //transfer erc20 token from buyer=msg.sender to owner
  currency.transferFrom(msg.sender, trade.owner, trade.price);

  //transfer nft item to buyer=msg.sender
  item.transferFrom(address(this), msg.sender, trade.item);

  //mark item as sold
  trades[_trade].status = 1;

}
```

Put all of these in a contract, and fix any bugs or issues. Remember to import the correct ERC implementations. Can you test the deployment?

It would be more fun to test this in a networked scenario.