

SOLIDITY CODE

// SPDX-License-Identifier: MIT

pragma solidity ^0.8.0;

```
contract BookRegistry {  
    address public owner;
```

```
  
    constructor() {  
        owner = msg.sender;  
    }
```

```
  
    modifier onlyOwner() {  
        require(msg.sender == owner, "Only the owner can perform this action");  
        _;  
    }
```

```
  
    struct Book {  
        string title;  
        string author;  
        address currentOwner;  
    }
```

```
  
    mapping(uint256 => Book) public books;  
    uint256 public bookCount;
```

```
  
    event BookAdded(uint256 indexed bookId, string title, string author, address indexed owner);  
    event OwnershipTransferred(uint256 indexed bookId, address indexed previousOwner, address indexed newOwner);
```

```
  
    function addBook(uint256 registration, string memory _title, string memory _author) external onlyOwner {
```

```
    books[registration] = Book(_title, _author, owner);
    bookCount++;
    emit BookAdded(registration, _title, _author, owner);
}
```

```
function transferOwnership(uint256 registrationId, address _newOwner) external {
    require(_newOwner != address(0), "Invalid address");
    require(_newOwner != books[registrationId].currentOwner, "The new owner is the same as the current owner");
    require(msg.sender == books[registrationId].currentOwner, "Only the current owner can transfer ownership");
```

```
    address previousOwner = books[registrationId].currentOwner;
    books[registrationId].currentOwner = _newOwner;
```

```
    emit OwnershipTransferred(registrationId, previousOwner, _newOwner);
}
```

```
function getBookDetails(uint256 registrationId) external view returns (string memory, string memory, address) {
```

```
    Book memory book = books[registrationId];
    return (book.title, book.author, book.currentOwner);
}
```

```
}
ABI
```

```
[
    {
        "inputs": [],
        "stateMutability": "nonpayable",
        "type": "constructor"
    },
    {
        "anonymous": false,
```

```
"inputs": [  
  {  
    "indexed": true,  
    "internalType": "uint256",  
    "name": "bookId",  
    "type": "uint256"  
  },  
  {  
    "indexed": false,  
    "internalType": "string",  
    "name": "title",  
    "type": "string"  
  },  
  {  
    "indexed": false,  
    "internalType": "string",  
    "name": "author",  
    "type": "string"  
  },  
  {  
    "indexed": true,  
    "internalType": "address",  
    "name": "owner",  
    "type": "address"  
  }  
],  
"name": "BookAdded",  
"type": "event"  
},  
{  
  "anonymous": false,  
  "inputs": [  
    {  
      "indexed": true,  
      "internalType": "uint256",  
      "name": "bookId",
```

```
"type": "uint256"
},
{
  "indexed": true,
  "internalType": "address",
  "name": "previousOwner",
  "type": "address"
},
{
  "indexed": true,
  "internalType": "address",
  "name": "newOwner",
  "type": "address"
}
],
"name": "OwnershipTransferred",
"type": "event"
},
{
  "inputs": [
    {
      "internalType": "uint256",
      "name": "registration",
      "type": "uint256"
    },
    {
      "internalType": "string",
      "name": "_title",
      "type": "string"
    },
    {
      "internalType": "string",
      "name": "_author",
      "type": "string"
    }
  ],
```

```
"name": "addBook",
"outputs": [],
"stateMutability": "nonpayable",
"type": "function"
},
{
"inputs": [],
"name": "bookCount",
"outputs": [
{
"internalType": "uint256",
"name": "",
"type": "uint256"
}
],
"stateMutability": "view",
"type": "function"
},
{
"inputs": [
{
"internalType": "uint256",
"name": "",
"type": "uint256"
}
],
"name": "books",
"outputs": [
{
"internalType": "string",
"name": "title",
"type": "string"
},
{
"internalType": "string",
"name": "author",
```

```
"type": "string"
},
{
  "internalType": "address",
  "name": "currentOwner",
  "type": "address"
}
],
"stateMutability": "view",
"type": "function"
},
{
  "inputs": [
    {
      "internalType": "uint256",
      "name": "registrationId",
      "type": "uint256"
    }
  ],
  "name": "getBookDetails",
  "outputs": [
    {
      "internalType": "string",
      "name": "",
      "type": "string"
    },
    {
      "internalType": "string",
      "name": "",
      "type": "string"
    },
    {
      "internalType": "address",
      "name": "",
      "type": "address"
    }
  ]
}
```

```
],
"stateMutability": "view",
"type": "function"
},
{
"inputs": [],
"name": "owner",
"outputs": [
{
"internalType": "address",
"name": "",
"type": "address"
}
],
"stateMutability": "view",
"type": "function"
},
{
"inputs": [
{
"internalType": "uint256",
"name": "registrationId",
"type": "uint256"
},
{
"internalType": "address",
"name": "_newOwner",
"type": "address"
}
],
"name": "transferOwnership",
"outputs": [],
"stateMutability": "nonpayable",
"type": "function"
}
]
```

BYTECODE

608060405234801561001057600080fd5b50336000806101000a81548173ff021916908373ff1602179055