

하이퍼레저패브릭 : 실습

https://youtu.be/pO0TzVlb_uM

IBM Blockchain Platform 설치 방법

스마트 컨트랙트 생성 및 배포

스마트 컨트랙트 업데이트

실습

IBM Blockchain Platform

- **IBM Blockchain Platform Extension for Visual Studio Code**

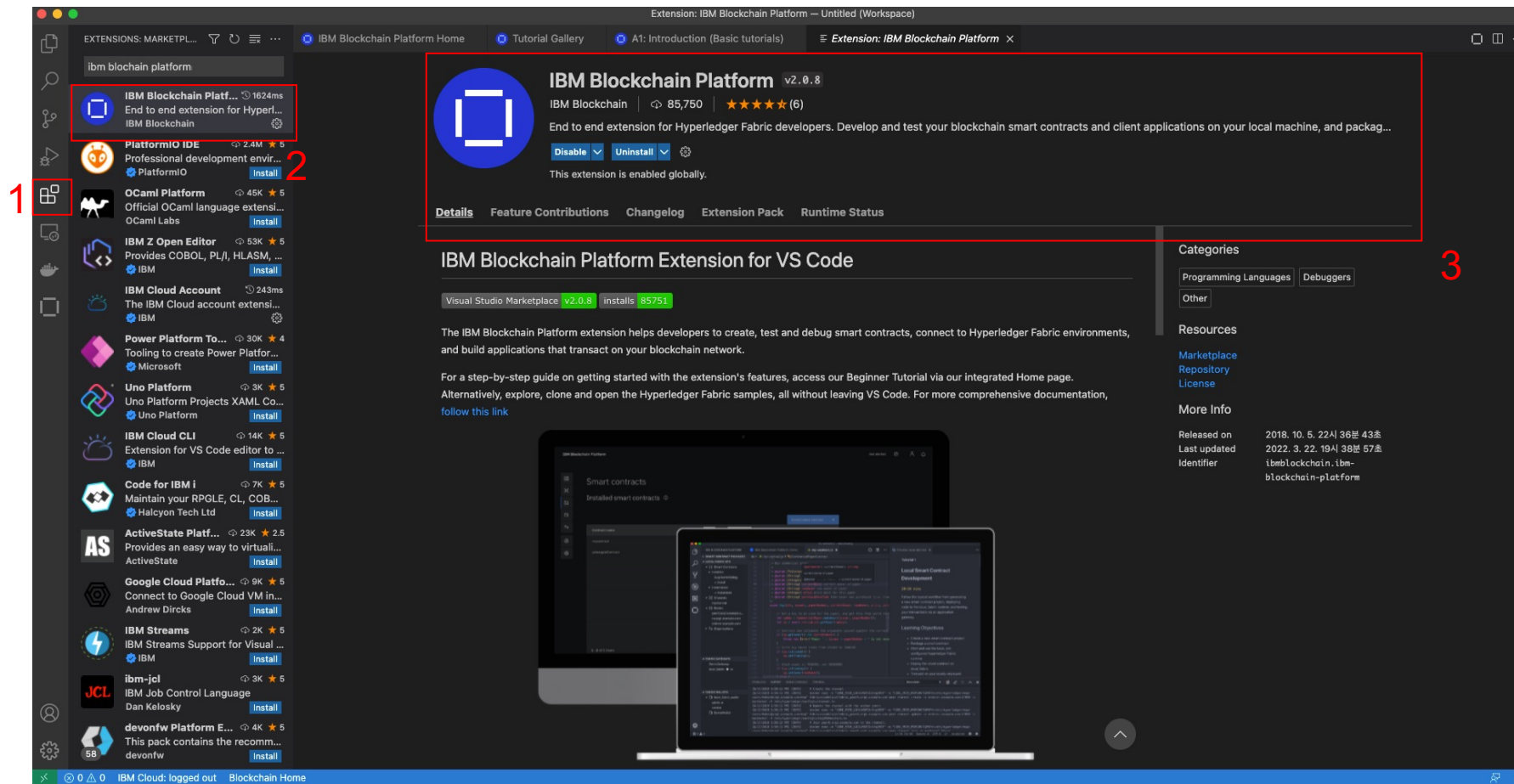
- 체인코드 작성
- 블록체인 네트워크용 어플리케이션 구축
- 내장된 패브릭 인스턴스를 사용하여 어플리케이션을 테스트 및 디버그

- **IBM Blockchain Platform Console**

- 하이퍼레저 패브릭 구성 요소를 생성하고 관리 가능

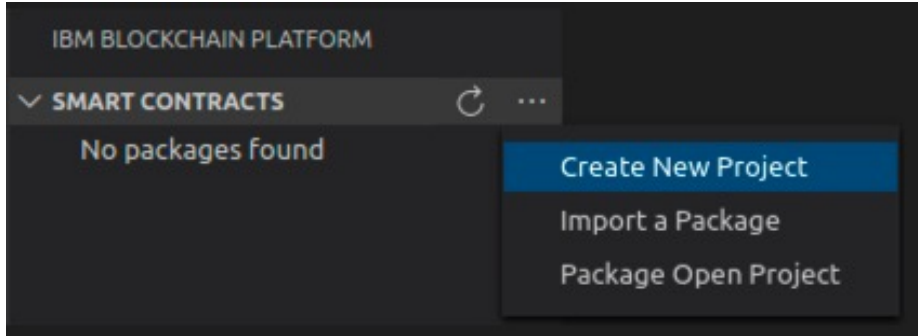
설치 방법

- VS Code 마켓플레이스에서 IBM Blockchain Platform 검색 후 설치



스마트 컨트랙트 생성

- 스마트 컨트랙트 프로젝트 생성



- 스마트 컨트랙트가 포함되어 있는 코드

```
1 export class MyAssetContract extends Contract {  
2
```

스마트 컨트랙트 생성

- **@Transaction 데코레이터**

- 트랜잭션 응답을 생성하기 위한 메소드
- True → 읽기, 쓰기 가능
- False → 읽기만 가능

```
1 @Transaction(false)
2 public async myAssetExists(ctx: Context, myAssetId: string): Promise<boolean> {
3
```

- **Context 객체**

- 데이터를 관리하는데 사용되는 객체
- 이를 통해 World State에 액세스

```
1 await ctx.stub.putState(myAssetId, buffer);
2
```

원장 (Ledger)

- **원장 (Ledger)**

- 하이퍼레저 패브릭 네트워크에서 생성되어 실행되는 모든 트랜잭션들에 대한 기록을 저장하는 객체
- Ledger = blockchain + world state

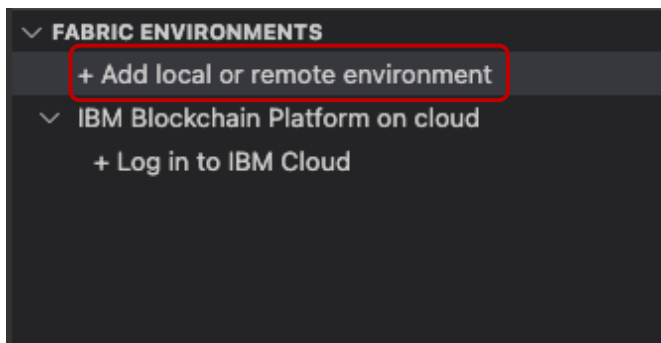
- **BlockChain**

- 트랜잭션들의 모든 블록이 암호학적 해시함수로 연결되어 있는 상태
- Ex) 계좌의 모든 거래 내역이 저장

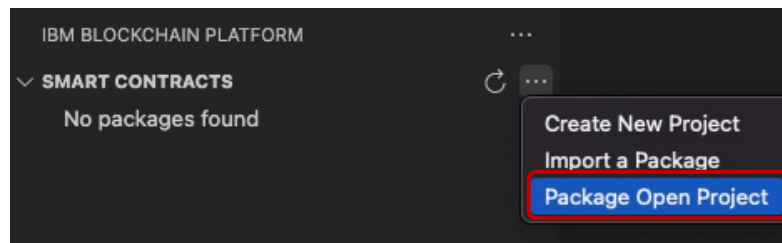
- **World state**

- 트랜잭션들의 로그들로 생성되고, 변경된 모든 키-값들의 최신 상태
- 데이터베이스 형태로 데이터를 저장
- 하이퍼레저 패브릭에서는 World state에 사용할 데이터베이스로 LevelDB, CouchDB 두 가지를 제공
- Ex) 계좌의 현재 남아있는 잔액을 저장

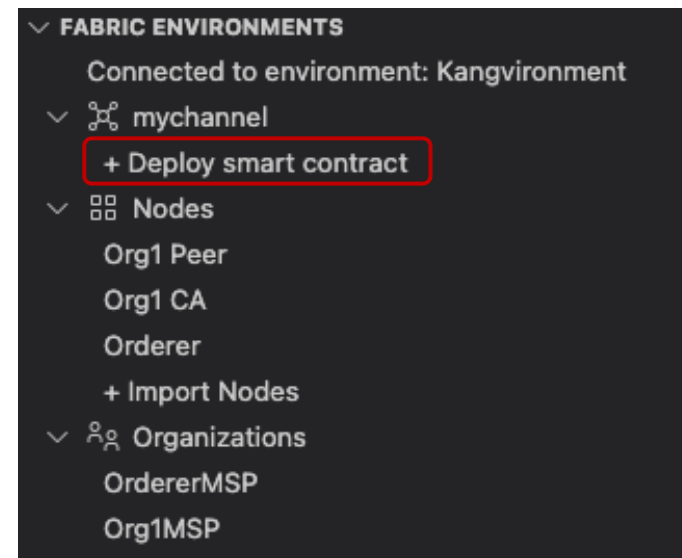
스마트 컨트랙트 배포



패브릭 환경 생성



스마트 컨트랙트 패키지로 만들기



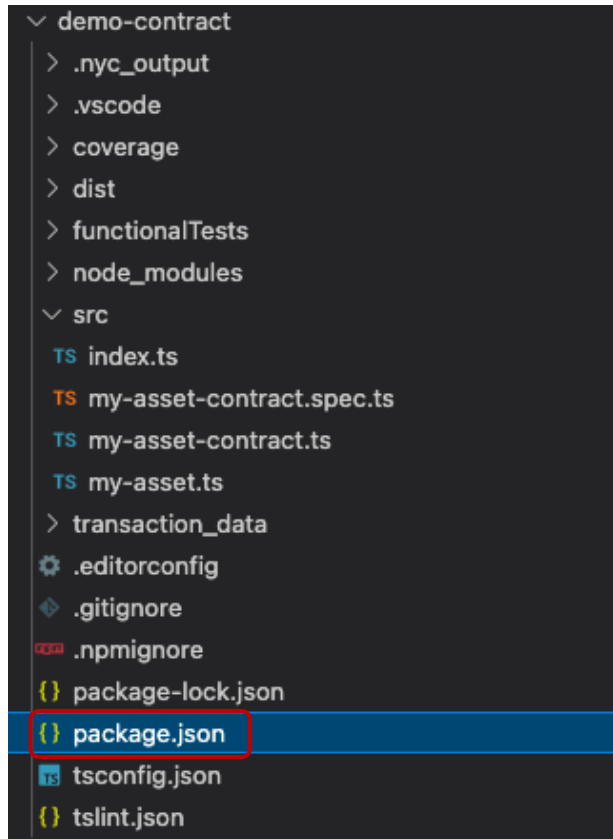
스마트 컨트랙트 배포

Identity, Wallet, Gateway

- Identity
 - 하이퍼레저 패브릭은 허가를 받아야 네트워크 접근 가능
 - 이때 접근할 수 있는 데이터는 사용자의 Identity에 따라 결정
- Wallet
 - 조직에서 발행한 Identity가 X.509 형태로 Wallet에 저장됨
- Gateway
 - 하이퍼레저패브릭 네트워크에 대한 연결
 - Wallet에 Identity가 있다면 게이트웨이 생성 가능
 - 게이트웨이를 통해 트랜잭션 제출

스마트 컨트랙트 업데이트

- 스마트 컨트랙트 파일 수정 후 Package.json 파일에서 version 수정



```
1  {  
2    "name": "demo-contract",  
3    "version": "0.0.1",  
4    "description": "My Smart Contract",  
5    "main": "dist/index.js",  
6    "typings": "dist/index.d.ts",  
7    "engines": {  
8      "node": ">=8",  
9      "npm": ">=5"  
10  },  
}
```

실습

Q & A