다중 주식 시스템 구현 TODO

1. 데이터베이스 구조 변경 ✅

1.1 테이블 삭제 및 생성 ✅

- [x] 기존 STOCK 테이블 삭제
- [x] STOCK_LIST 테이블 생성
- [x] CHARACTER_STOCK 테이블 생성 (복합키: CHARACTER_ID, STOCK_ID)

1.2 LIFE_CHARACTER 테이블 수정

- [ ] STOCK_COUNT 컬럼 제거 ⚠️ **누락됨 - 추가 필요**
- [ ] STOCK_VALUE 컬럼 제거 ⚠️ **누락됨 - 추가 필요**

1.3 초기 데이터 삽입 ✅

- [x] STOCK_LIST에 주식 목록 INSERT
  - 삼선전자 (9,000만원)
  - KH로우닉스 (27,000만원)
  - 롯뚜기식품 (1,300만원)
  - KIG넥스트 (800만원)
  - 차이슬라 (400만원)

2. DAO 계층 변경

2.1 StockDao.java 완전 재작성

- [ ] 기존 메서드 모두 삭제
- [ ] `getAllStocks(int characterId)` 구현
- [ ] `getStockPrice(int characterId, int stockId)` 구현
- [ ] `updateStockPrice(int characterId, int stockId, int newPrice)` 구현
- [ ] `buyStock(int characterId, int stockId, int count, int price)` 구현
- [ ] `sellStock(int characterId, int stockId, int count)` 구현
- [ ] `initializeCharacterStocks(int characterId)` 구현 ⭐ **주식별 초기 가격 설정**
- [ ] `getTotalStockValue(int characterId)` 구현

2.2 query.xml 수정

- [ ] 기존 주식 관련 쿼리 삭제
- [ ] `getAllStocks` 쿼리 추가
- [ ] `getStockPrice` 쿼리 추가
- [ ] `updateStockPrice` 쿼리 추가
- [ ] `buyStock` 쿼리 추가

- [ ] `sellStock` 쿼리 추가
- [ ] `initializeCharacterStocks` 쿼리 추가 ⭐

```sql
INSERT INTO CHARACTER_STOCK (CHARACTER_ID, STOCK_ID, STOCK_CNT, PRICE)
VALUES
(?, 1, 0, 9000),  -- 삼선전자
(?, 2, 0, 27000),  -- KH로우닉스
(?, 3, 0, 1300),  -- 롯뚜기식품
(?, 4, 0, 800),  -- KIG넥스트
(?, 5, 0, 400)  -- 차이슬라
```

- [ ] `getTotalStockValue` 쿼리 추가

2.3 LifeCharacterDao.java 수정

- [ ] `createNewCharacter`에서 주식 초기화 로직 추가
- [ ] `executeStockTrade` 메서드 파라미터 수정 (stockId 추가)
- [ ] `execute` 메서드에서 STOCK_COUNT, STOCK_VALUE 관련 코드 제거

3. Service 계층 변경

3.1 StockService.java 수정

- [ ] `getCurrentStockPrice(int characterId, int stockId)` 시그니처 변경
- [ ] `updateStockPrice(int characterId)` → 모든 주식 가격 업데이트로 변경
- [ ] `updateAllStockPrices(int characterId)` 새 메서드 추가
- [ ] `getTotalStockValue(int characterId)` 새 메서드 추가
- [ ] `getAllStocks(int characterId)` 새 메서드 추가

3.2 LifeCharacterService.java 수정

- [ ] `executeStockTrade` 메서드 파라미터 수정
- [ ] `createNewCharacter`에서 주식 초기화 호출 추가

4. Model 계층 변경

4.1 LifeCharacter.java 수정

- [ ] `stockCount` 필드 제거
- [ ] `stockValue` 필드 제거
- [ ] `buyStock()` 메서드 제거
- [ ] `sellStock()` 메서드 제거
- [ ] `updateStockValue()` 메서드 제거
- [ ] `getTotalAsset()` 메서드 수정 (Service를 통해 주식 가치 계산)

4.2 Stock.java VO 재작성

- [ ] `stockId` 필드 추가
- [ ] `stockName` 필드 추가
- [ ] `price` 필드 유지
- [ ] `stockCount` 필드 추가
- [ ] 생성자, getter/setter 재작성

5. Controller 계층 변경

5.1 StockController.java 수정

- [ ] `updateStockPrice(int characterId)` → 모든 주식 업데이트로 변경
- [ ] `getCurrentStockPrice(int characterId, int stockId)` 파라미터 수정
- [ ] `getAllStocks(int characterId)` 새 메서드 추가

5.2 LifeCharacterController.java 수정

- [ ] `updateStockTrade` 호출 시 stockId 파라미터 전달
- [ ] `updateCharacter`에서 주식 가격 업데이트 로직 수정

6. View 계층 변경

6.1 StockMenu.java 대폭 수정

- [ ] `showMenu()` - 주식 목록 표시 로직 추가
- [ ] `displayStockList()` 새 메서드 추가
- [ ] `selectStock()` 주식 선택 메서드 추가
- [ ] `buyStock()` 메서드에 stockId 파라미터 추가
- [ ] `sellStock()` 메서드에 stockId 파라미터 추가
- [ ] 메뉴 UI 개선 (5개 주식 목록 표시)

6.2 InvestMenu.java 수정

- [ ] 주식 메뉴 진입 시 다중 주식 정보 표시

6.3 UiTemplate.java 수정

- [ ] `investmentStat()` 메서드 수정 - 다중 주식 정보 표시
- [ ] 주식 포트폴리오 표시 로직 추가

7. 게임 로직 수정

7.1 캐릭터 생성 시 초기화 ⭐

- [ ] 새 캐릭터 생성 시 5개 주식을 각각의 초기 가격으로 설정
  - 삼선전자: 9,000만원
  - KH로우닉스: 27,000만원

- 롯뚜기식품: 1,300만원
- KIG넥스트: 800만원
- 차이슬라: 400만원

- [ ] 보유량은 모두 0주로 설정

7.2 주식 가격 변동

- [ ] 일하기 완료 후 모든 주식 가격 변동 적용
- [ ] 각 주식별로 독립적인 변동률 적용 (-10% ~ +11%)

7.3 자산 계산

- [ ] `getTotalAsset()` 호출 시 모든 주식의 총 가치 계산
- [ ] 랭킹 시스템에서 주식 가치 포함

8. 테스트 및 검증

8.1 기능 테스트

- [ ] 캐릭터 생성 시 5개 주식 초기화 확인
- [ ] 주식 구매/판매 정상 작동 확인
- [ ] 5개 주식 각각의 가격 변동 확인
- [ ] 총 자산 계산 정확성 확인 (다중 주식 합계)
- [ ] 랭킹 시스템 정상 작동 확인

8.2 데이터 무결성 확인

- [ ] 외래키 제약조건 정상 작동 확인
- [ ] 가격 양수 제약조건 확인
- [ ] 복합키 중복 방지 확인

9. 추가 개선사항 (선택)

9.1 UI/UX 개선

- [ ] 주식별 수익률 표시
- [ ] 포트폴리오 비중 표시
- [ ] 주식 정렬 기능 (가격순, 보유량순, 수익률순)
- [ ] 고가주/저가주 구분 표시

9.2 게임 밸런스

- [ ] 주식별 변동성 차별화 (고가주 = 낮은 변동성, 저가주 = 높은 변동성)
- [ ] 배당 시스템 추가
- [ ] 주식 분할/합병 이벤트

⚠️ 긴급 수정 필요사항

```sql
-- init.sql에 추가 필요
ALTER TABLE LIFE_CHARACTER DROP COLUMN STOCK_COUNT;
ALTER TABLE LIFE_CHARACTER DROP COLUMN STOCK_VALUE;

-- INSERT 구문 오류 수정
INSERT INTO STOCK_LIST VALUES(1, '삼선전자');
INSERT INTO STOCK_LIST VALUES(2, 'KH로우닉스');
INSERT INTO STOCK_LIST VALUES(3, '롯뚜기식품');
INSERT INTO STOCK_LIST VALUES(4, 'KIG넥스트');
INSERT INTO STOCK_LIST VALUES(5, '차이슬라');
```

진행 순서 권장사항

1. **긴급 수정사항 적용** ⚠️
2. **DAO 계층 구현** (2단계) - 특히 `initializeCharacterStocks` 중요
3. **Service 계층 수정** (3단계)
4. **Model 계층 수정** (4단계)
5. **Controller 계층 수정** (5단계)
6. **View 계층 수정** (6단계)
7. **테스트 및 검증** (8단계)

각 단계를 완료한 후 다음 단계로 진행하여 오류를 최소화하세요.