Angelia migration guideline

1. 關掉網路連線，避免改動到 qat/product 的 db 資料

2. spin up migration db

確定db是空的 (砍掉 db內容，或砍掉重建container)

```
    docker run -d --name angelia-db-migration -p 5432:5432 -e
POSTGRES_USER="angelia" -e POSTGRES_PASSWORD="password" -e
POSTGRES_DB="angelia" -d postgres:9.6

    # go inside and create schema `angelia`
    psql -h localhost -U angelia angelia
    CREATE SCHEMA angelia;
```

3. stash ur code and check out to migration branch

```
    git stash --include-untracked
    git checkout migration0
```

4. add `ormconfig.json`in root folder

```
    {
      "type": "postgres",
      "host": "localhost",
      "port": 5432,
      "username": "angelia",
      "password": "password",
      "database": "angelia",
      "schema": "angelia",
      "entities": ["src/entities/*.ts"],
      "migrations": ["tmp/migrations-local/*.ts"],
      "cli": {
        "migrationsDir": "tmp/migrations-local"
      }
    }
```

5. run migration

```
    # gen migration
    npx ts-node ./node_modules/.bin/typeorm migration:generate -n
existed_db_entities

    # 把 migration run 出來的那個檔案 執行在 目前連線上的 migration db 上
    npx ts-node ./node_modules/.bin/typeorm migration:run;

    # 進入 db 確認東西有進去
```

6. pop/applied the stashed code and gen migration for what we need

```
    # gen migration
    npx ts-node ./node_modules/.bin/typeorm migration:generate -n
createXXXEntity

    # 看 gen 出來的 migration file, 確認正確性

    # 如果需要，可以把 migration run 出來的那個檔案 執行在 目前連線上的
migration db 上
    npx ts-node ./node_modules/.bin/typeorm migration:run
```

7. Other

- You can just copy and paste the generated migration file into src/migrations folder, take care the naming
- To applied newly added db entity change to qat, update `ormconfig.json` to point to the qat db to run the migration