Properties 설정 정리

1. Properties 목록

* application.properties
* application-local.properties(로컬)
* application-dev.properties(개발)
* application-production.properties(운영)

1. 상황에 맞는 properties 참조 방법

* application.properties 에서 spring.profiles.active 속성에 설정으로 원하는 properties를 참조할 수 있다.
* 로걸환경(application-local.properties)

spring.profiles.active=local

* 개발환경(application-dev.properties)

spring.profiles.active=dev

* 운영환경(application-production.properties)

spring.profiles.active=production

1. Properties 설정 속성

# 파일 업로드 directory

file.upload.directory=/data/apps/ebid/uploads/

# tradeSign 설정파일 위치

certificate.setting.path=/data/apps/ebid/data/tradesign3280.properties

# 서버용(계열사) 인증서 위치

certificate.file.path=/data/apps/ebid/data/config/ServerCert/

# 파일 암호화 키(입찰 첨부파일 업로드)

file.encrypted.key=626974637562656669676874696E6721

#스케줄러 사용여부

common.schedule.flag=false

#메일 관련 설정

#Mail host

mail.host=197.200.1.81

#form Mail sender

mail.sender.address=ithelpdesk@iljin.co.kr

#mail reply

mail.replyto.address=ithelpdesk@iljin.co.kr

#DB 접속정보

spring.datasource.driver-class-name=net.sf.log4jdbc.sql.jdbcapi.DriverSpy

spring.datasource.url=jdbc:log4jdbc:mariadb://168.126.74.44:3310/ebiddb

spring.datasource.username=ebid

spring.datasource.password=Ebid!!00

#SMS 전송시 필요한 Oracle DB 접속을 위한 정보

oracle.datasource.driver-class-name=oracle.jdbc.driver.OracleDriver

oracle.datasource.url=jdbc:oracle:thin:@197.200.16.37:1584:EBID

oracle.datasource.username=ebid

oracle.datasource.password=ebid

#t\_co\_user use\_yn schedule update DB connection

oracle.ehr.datasource.driver-class-name=oracle.jdbc.driver.OracleDriver

oracle.ehr.datasource.url=jdbc:oracle:thin:@197.200.11.158:1581:EHR

oracle.ehr.datasource.username=ebid

oracle.ehr.datasource.password=ebid

# JPA 엔티티와 데이터베이스 테이블 매핑 설정

# physical-strategy는 entity가 실제 DB이름이랑 어떻게 매핑되는지 설정

spring.jpa.hibernate.naming.physical-strategy=org.hibernate.boot.model.naming.PhysicalNamingStrategyStandardImpl

# Hibernate가 실행한 SQL 쿼리를 콘솔에 출력할지 여부.

spring.jpa.show-sql=true

# JPA 스키마 자동생성 옵션

# create : 애플리케이션이 시작될 때마다 이미 존재하는 테이블은 삭제하고 새로 만든다.

# update : 변경된 스키마를 반영

# validate : 현재 데이터베이스 스키마가 JPA 엔티티와 일치하는지 확인한다. 일치하지 않으면 오류발생

# create-drop : 애플리케이션이 시작될 때마다 테이블을 생성하고 종료될 때마다 테이블을 삭제한다.

# none : Hibernate가 자동으로 데이터베이스 스키마를 변경하지 못하도록 설정한다.

# 운영시에는 항상 none으로 설정해야한다. \*\*\*\*\*\*\*\*\*\*중요\*\*\*\*\*\*\*\*\*\*

spring.jpa.hibernate.ddl-auto=none

# SQL 쿼리에 주석 추가 여부

spring.jpa.properties.hibernate.use\_sql\_comments=true

# SQL 쿼리문 포맷팅 여부(키워드 대문자로 변환, 각 구문 새줄 표시, 들여쓰기..)

spring.jpa.properties.hibernate.format\_sql=true

# 파일업로드

# 파일 업로드와 같은 multipart 요청에 대한 활상화 여부

spring.servlet.multipart.enabled=true

# 파일의 서버로 전송 가능한 최대 크기

spring.servlet.multipart.max-request-size=100MB

# 개별 파일의 크기가 100MB 를 초과하면 거부됨

spring.servlet.multipart.max-file-size=100MB

#서버 포트 설정

server.port=18500

#서버에서 발생한 오류 응답에 에러 메시지를 포함할지 여부

#always를 사용하면 항상 에러 메시지가 포함되며, 클라이언트에게 전달

#never로 설정하면 에러 메시지가 포함되지 않음

server.error.include-message=always

#ssl 설정

#ssl 키파일 경로

server.ssl.key-store=/etc/nginx/ssl/cert.p12

#ssl 키파일 비밀번호

server.ssl.key-store-password=DU61A7F2

#ssl 키파일 형식

server.ssl.keyStoreType=PKCS12

#세션 관리 방법

# JDBC를 이용해 세션을 저장하여 세션 데이터를 데이터베이스에 저장(SPRING\_SESSION)

spring.session.store-type=jdbc

# JDBC 세션 저장소를 초기화할 때 스키마를 생성할지 여부.

# never는 스키마가 이미 생성되어 있다고 가정하고 초기화하지 않음

spring.session.jdbc.initialize-schema=never

#spring.session.timeout.seconds=86400

# Actuator 엔드포인트

# 외부 시스템이나 관리도구에 모든 Actuaotr 엔드포인트 노출(애플리케이션의 health, info, environment, thread, memory 사용량 등의 정보)

management.endpoints.web.exposure.include=\*

# shutdown 엔드포인트를 활성화하여 외부에서 애플리케이션 종료 가능

management.endpoint.shutdown.enabled=true

#sso 관련설정

server.servlet.context-parameters.RTSSO\_SP\_CONFIG=classpath:rathon\_sso\_sp\_dev.properties

opagent.app.ptk.use=N

opagent.sso.server.context=

opagent.sso.server.login.url=index.html

opagent.sso.server.logout.url=logout.html

opagent.sso.server.ip=197.200.11

opagent.sso.server.port=

opagent.cookieUse=Y

opagent.validate.request.timeout=5000

opagent.sso.client.login=N

opagent.login.view.emate.apps=APP\_155

opagent.sso.server.url=http://wingsso.iljin.co.kr/

opagent.apps.login.url=http://localhost:18500/login/sso

opagent.apps.logout.url=http://localhost:18500/logout/sso

# 로그 관련 설정

#logging.level.org.springframework=DEBUG

logging.level.iljin.framework=DEBUG

logging.file.path=./logs

logging.level.org.springframework.jdbc.core = DEBUG

logging.level.org.springframework.web=DEBUG

#logging.level.org.hibernate.type=TRACE