

Oracle USASCII 이슈

Oracle USASCII 이슈 해결

- MapperConfig.xml
 - typeHandlers 설정 추가

```
<typeHandlers>
    <typeHandler jdbcType="VARCHAR" javaType="java.lang.String" handler="com.humuson.tms.web.util.OracleUsTypeHandler"/>
    <typeHandler javaType="java.lang.String" handler="com.humuson.tms.web.util.OracleUsTypeHandler"/>
</typeHandlers>
```

- handler 지정 경로 아래 class파일 추가

```
package com.humuson.tms.web.util;

import lombok.extern.slf4j.Slf4j;
import org.apache.ibatis.type.JdbcType;
import org.apache.ibatis.type.TypeHandler;
import org.springframework.context.annotation.Configuration;

import java.io.UnsupportedEncodingException;
import java.sql.CallableStatement;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

@Slf4j
@Configuration
public class OracleUsTypeHandler implements TypeHandler {

    @Override
    public void setParameter(PreparedStatement ps, int i, Object parameter,
                            JdbcType jdbcType) throws SQLException {
        String str = null;
        try {
            str = new String(parameter.toString().getBytes("EUC-KR"), "ISO_8859_1");
            System.out.println(str);
        } catch (UnsupportedEncodingException e) {
            log.error("UnsupportedEncodingException : {}", e.getMessage());
        } catch (Exception localException) {
            log.debug("error", localException);
        }
        ps.setString(i, str);
    }

    @Override
    public Object getResult(ResultSet rs, String columnName)
        throws SQLException {
        byte[] str = rs.getBytes(columnName);
        try {
            // System.out.println("getResult String 확인 : \n");
            System.out.println(new String(str, "EUC-KR"));
            return new String(str, "EUC-KR");
        } catch (UnsupportedEncodingException e) {
            log.info("UnsupportedEncodingException : {}", e.getMessage());
        } catch (Exception localException) {
            log.debug("error", localException);
        }
        return "";
    }

    public Object getResult(ResultSet rs, int columnIndex) throws SQLException {
        byte[] arr = rs.getBytes(columnIndex);
        try {
            System.out.println("getResult int 확인 : \n");
        }
    }
}
```

```

        System.out.println(new String(arr, "EUC-KR"));
        return new String(arr, "EUC-KR");
    } catch (UnsupportedEncodingException e) {
        e.printStackTrace();
    }
    return "";
}

@Override
public Object getResult(CallableStatement cs, int columnIndex)
    throws SQLException {
    byte[] arr = cs.getBytes(columnIndex);
    try {
        System.out.println("getResult callable 확인 : \n");
        System.out.println(new String(arr, "EUC-KR"));
        return new String(arr, "EUC-KR");
    } catch (UnsupportedEncodingException e) {
        e.printStackTrace();
    }
    return "";
}
}

```

- **@Configuration 어노테이션 사용**

- **setParameter** : sql mapper에 들어갈 parameter 값을 EUC-KR byte로 받아와 ISO_8859_1 변환
- **getResult** : mapper ResultSet의 String Type 값들을 EUC-KR로 변환

! DB의 VARCHAR 타입만 변환시킴(NUMBER 타입은 Long, DATE 타입은 Mapper에서 String 타입으로 받거나, Date 로 받아야 getResult를 거치지 않고 정상적으로 들어옴)

! NUMBER 타입의 경우 DTO(getter/setter) 부분을 Long 타입으로 세팅해주어야함
 (String으로 받을 경우 getResult를 거치므로 깨지게됨)