# Setup

Vpn: big-id

My shortName computer: \\Cscvieai374839

<https://www.traveloka.com/vi-vn/>

<https://www.airbnb.com/>

<https://archive.org/details/Tropangnj8815hp01>

User and pass cho 20.194.10.15 để lấy code:

cscguest/8S3Pt24s~

tnguyen443/Tn9yY4ne@E

remote:

.\cscv

P@ssword123

|  |  |  |
| --- | --- | --- |
| Nguyen Thien Thuat | tnguyen443 | k7r77ka7af |

**Meeting:**

Call: 912010288 / 12010288

-> 8885674484

pass: 5285386#

95 CSCVIEAI374818

People manager hien tai: Le Van Hung (1119340)

Địa chỉ IBM để tìm hiểu lệnh:

<http://www.ibm.com/support/knowledgecenter/ssw_ibm_i_71/cl/call.htm>

Địa chỉ a.Hoang kiu tìm hiểu:

<http://www.go4as400.com/>

Trong file doc anh Hoang Dang co gui 1 chương trình cài đặt sau:

New svn: http://20.0.2.132

## - 2 tool trong STS

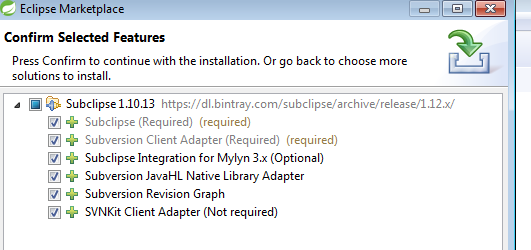
16.687.477

A Hoàng kiu setup 2 cái tool này để làm việc (hình như là 1 dạng SVN cho eclipse):



## Subclipse

Nên cài đặt cái này: gõ svn search. Sau đó có Subclipse 1.10.13 (hiện tại)



## - Tìm hiểu FileZilla

**FileZilla Client** (còn được gọi tắt **là FileZilla**) **là** [trình khách FTP] tự do, mã nguồn mở, đa nền tảng. Sẵn dùng cho Windows, Linux, và Mac OS X. Hỗ trợ FTP, SFTP, và FTPS. Tính đến 5 tháng 3 2009, **FileZilla**đứng thứ 5 trong số những phần mềm được tải về nhiều nhất từ SourceForge.net.

FTP là File Transfer Protocol: giao thức chuyển nhượng tập tin. Port mặc định là 21. Tuy Control Panel có giao thức transfer khác, nhưng người ta cũng dùng giao thức này, thường dùng để transfer data giữa user và host.

## Cài đặt BeyondCompare 4:



## Cài đặt lại jdk7 (jdk rộng hơn jre, vì trong máy có sẵn jre nên mình muốn setup lại jdk):



## Lỗi workbench

Vào STS tăng dung lượng maximum lên và xóa luôn cái minimum



## SVN

Your svn userid/password is:

tnguyen443/Tn9yY4ne@E

# 2. Các địa chỉ học Spring:

<https://www.tutorialspoint.com/spring/spring_hello_world_example.htm>

# 3. HelloSpring:

## 3.1 Giới thiệu:

Đây là project mình thực hiện mô phỏng đầy đủ các chức năng của Spring Framework, từ đây về sau, có gì thắc mắc thì lên Project HelloSpring kiếm trước.

<https://codersontrang.com/2013/01/20/gioi-thieu-ve-spring-framework/>

Spring nổi bật ở tính năng liên kết các thành phần lại với nhau theo cách riêng và dễ quản lý. Spring tạo ra 1 framework dễ xây dựng ứng dụng hơn J2EE.

## 3.2 Thư viện

Các thư viện cài đặt cho ví dụ Spring:



Lib dành cho cache:



Lib dành cho log:



## 3.3 Tìm hiểu về file Beans.xml - beanwiring

Beans.xml có tác dụng liên kết các thành phần trong Spring.

<https://codersontrang.com/2013/01/25/bean-wiring-cac-cach-de-lien-ket-cac-thanh-phan-trong-spring/>

Như đã biết, Spring dùng thẻ **<bean>** để khởi tạo một thành phần. Trong một thành phần này, có thể có nhiều thuộc tính tham chiếu đến khác thành phần khác được quản lý bởi Spring container. Bình thường, nếu như không dùng Spring, một thuộc tính sẽ được khai báo để tham chiếu đến một giá trị/đối tượng nào đó qua contructor hoặc phương thức setXXX() tương ứng với thuộc tính đó. Tương tự, Spring cũng cung cấp đầy đủ các cách khai báo để liên kết các thành phần qua Contructor hoặc các phương thức setter.

…

## 3.4 Properties trong java

Với những thông tin như jdbc driver, URL, port, schema…, thay vì viết trực tiếp như vậy, hoặc đặt thành những hằng số trong code thì ta có thể lưu chúng vào file cấu hình (chẳng hạn databaseConfig.properties).

Properties cũng có key và value tương tự như map, nhưng quan trọng nhất của properties là có thể năng save thành file text và load file text lên.

## 3.5 Tìm hiểu về file Beans.xml – autowiring

<https://codersontrang.com/2013/02/02/bean-autowiring-co-che-tu-dong-lien-ket-cac-thanh-phan-trong-spring/>

Qua bài viết “[Bean Wiring – các cách để liên kết các thành phần trong Spring](http://wp.me/p2A80w-5E)“, chúng ta đã được giới thiệu về các cách cơ bản mà các thành phần được khởi tạo và liên kết với nhau trong Spring thông qua việc khai báo ở file cấu hình (.xml) như thế nào. Như đã biết, chúng ta phải trực tiếp dùng thẻ <constructor-arg> hoặc <property> để chỉ ra thành phần nào sẽ được tham chiếu đến. Spring còn có một cơ chế để giảm bớt đi việc khai báo này mà vẫn đảm bảo các thành phần có thể liên kết với nhau, đó là cơ chế Bean Autowiring. Việc liên kết giữa các thành phần sẽ diễn ra hoàn toàn tự động. Cơ chế tự động được thực hiện bởi Spring container dựa vào một trong 4 cách thức sau:

* **by Name**: Tự động liên kết một thuộc tính đến một thành phần có tên trùng với tên của thuộc tính đó
* **by Type**: Tự động liên kết một thuộc tính đến một thành phần có kiểu khớp với kiểu của thuộc tính đó
* **by Constructor**: Giống như byType, nhưng thay vì tìm đến một thành phần có kiểu khớp với kiểu của tham số trong các phương thức setter thì ở đây sẽ tìm đến một thành phần có kiểu khớp với kiểu của tham số trong constructor.
* **by AutoDetect**: Tự động thực hiện cơ chế tự động liên kết *by Constructor* trước, nếu không có sự liên kết thỏa mãn, sẽ thực hiện tiếp cơ chế tự động liên kết *by Type*

## 3.6 Cơ chế Cache trong Spring Framework:

### Spring 3

<https://codersontrang.com/2013/03/04/co-che-cache-trong-spring-framework/>

Lưu ý cách này không dùng được spring 4, chỉ có thể áp dụng cho spring 3.

Các thư viện sử dụng:



File ehcache.xml – file này éo biết làm mẹ gì, nhưng có nó sẽ gây ra lỗi:

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<ehcache xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xsi:noNamespaceSchemaLocation=*"http://ehcache.sf.net/ehcache.xsd"*>

<diskStore path=*"java.io.tmpdir"* />

<cache name=*"products"*

maxEntriesLocalHeap=*"100"*

maxEntriesLocalDisk=*"1000"*

eternal=*"false"*

timeToIdleSeconds=*"300"*

timeToLiveSeconds=*"600"*

memoryStoreEvictionPolicy=*"LFU"*

transactionalMode=*"off"*>

<persistence strategy=*"localTempSwap"* />

</cache>

</ehcache>

Lỗi mà cái file ngu ehcache.xml gây ra:



### Spring 4: chuyển qua sử dụng annotation

<http://websystique.com/spring/spring-4-cache-tutorial-with-ehcache/>

Spring4CachingExample2.zip: file này nằm trong doc của project HelloSpring, đây là ví dụ của trang web bên trên, có sử dụng down để load thư viện.

There are many caching solutions available in market today namely EhCahe, Guava Cache, Caffeine Cache, OScache, JBoss Cache,etc.

Caching Annotations:

* @Cacheable : triggers cache population
* @CacheEvict : triggers cache eviction
* @CachePut : updates the cache without interfering with the method execution
* @Caching : regroups multiple cache operations to be applied on a method
* @CacheConfig : shares some common cache-related settings at class-level
* @EnableCaching : Configuration level annotation which enables Caching

#### Các package liên quan trong ví dụ này:

com.websystique.spring.model

com.websystique.spring.service

com.websystique.spring.configuration

**File ProductServiceImpl khi không sử dụng cache:**

**package** com.websystique.spring.service;

**import** org.springframework.stereotype.Service;

**import** com.websystique.spring.model.Product;

@Service("productService")

**public** **class** ProductServiceImpl **implements** ProductService{

@Override

**public** Product getByName(String name) {

showLookupOperation();

**return** **new** Product(name, 100);

}

**public** **void** showLookupOperation(){

**try** {

**long** time = 5000L;

Thread.*sleep*(time);

} **catch** (InterruptedException e) {

**throw** **new** IllegalStateException();

}

}

}

**AppConfig khi chưa xài cache của Spring:**

**package** com.websystique.spring.configuration;

**import** java.util.Arrays;

**import** org.springframework.cache.CacheManager;

**import** org.springframework.cache.concurrent.ConcurrentMapCache;

**import** org.springframework.cache.support.SimpleCacheManager;

**import** org.springframework.context.annotation.Bean;

**import** org.springframework.context.annotation.ComponentScan;

**import** org.springframework.context.annotation.Configuration;

@Configuration

@ComponentScan(basePackages = "com.websystique.spring")

**public** **class** AppConfig {

@Bean

**public** CacheManager cacheManager() {

// configure and return an implementation of Spring's CacheManager SPI

SimpleCacheManager cacheManager = **new** SimpleCacheManager();

cacheManager.setCaches(Arrays.*asList*(**new** ConcurrentMapCache("products")));

**return** cacheManager;

}

}

**SampleApplication khi chưa xài cache của Spring:**

**package** com.websystique.spring.configuration;

**import** org.slf4j.Logger;

**import** org.slf4j.LoggerFactory;

**import** org.springframework.context.annotation.AnnotationConfigApplicationContext;

**import** org.springframework.context.support.AbstractApplicationContext;

**import** com.websystique.spring.service.ProductService;

**public** **class** SampleApplication {

**private** **static** **final** Logger ***logger*** = LoggerFactory.*getLogger*(SampleApplication.**class**);

**public** **static** **void** main(String[] args) {

AbstractApplicationContext context = **new** AnnotationConfigApplicationContext(AppConfig.**class**);

ProductService service = (ProductService) context.getBean("productService");

***logger***.info("Iphone ->"+service.getByName("Iphone"));

***logger***.info("Iphone ->"+service.getByName("Iphone"));

***logger***.info("Iphone ->"+service.getByName("Iphone"));

((AbstractApplicationContext) context).close();

}

}

Nếu không có cache, phải 5 giây thì chương trình mới lấy được data:



Lib dành cho log:



Lib dành cho cache:



Không hiểu sao cache dùng cho ví dụ này vẫn k xài được.

Trong lúc sử dụng trên trang chủ Spring có nói đến Gradle, cải tiến từ Ant và Maven, khuyên mình đọc sách này: Gradle in Action

### Spring 4: Ví dụ trên trang chủ Spring:

<https://spring.io/guides/gs/caching>

Giới thiệu khái niệm cache trong spring:

<http://dangtritue.blogspot.com/2008/09/xy-dng-h-thng-cache-vi-spring-aop.html>

Thư viện mình dùng maven để down được, chạy được luôn:



Tới đây thì mình đã hoàn thành được thêm và xóa cache trong spring cơ bản. Lưu ý là mình cần xóa cái ehcache.xml đi vì nó cấu hình cái cache. Có lẽ nên tìm hiểu file này sau.

Tới đây thì có các thư viện sau:



## 3.7 autowired sử dụng annotation @autowired:

<https://huongdanjava.com/bean-autowiring-su-dung-autowired-annotation.html>

Mẹ cái địa chỉ ở trên vô dụng vkl, tiếng Việt viết như cờ cờ

Địa chỉ ở đây mới đúng nè:

<https://www.tutorialspoint.com/spring/spring_autowired_annotation.htm>

Sử dụng package com.autowired.annotation

Cái annotation @autowired theo mình thấy nó autowired thành phần nào đó trong 1 class, chứ k phải autowired nguyên cả cái class.

Bonus thêm 1 số annotation:

<http://dodangquan.blogspot.com/2016/01/mot-so-annotation-trong-spring.html>

## 3.8 Bên lề: Slf4j

Đây là cải tiến của log4j. Mình làm ví dụ này trong package slf4j.log

<https://nmhblog.wordpress.com/2010/08/21/java-logging/>

Mặc định là logger.info hay logger.debug sẽ print ra màn hình console. Để có file log ta phải chỉnh cấu hình logback.xml như sau:

<configuration>

<appender name=*"FILE"* class=*"ch.qos.logback.core.FileAppender"*>

<file>slf4j-example.log</file>

<encoder>

<pattern>%date %level [%thread] %logger{10} [%file:%line] %msg%n</pattern>

</encoder>

</appender>

<root level=*"debug"*>

<appender-ref ref=*"FILE"* />

</root>

</configuration>

Sau đó file slf4j-example.log sẽ nằm trong project HelloSpring

Result:



Vì root level mình để debug nên nó sẽ in tất cả mọi thứ ra (debug+info+…)

Nếu mình để root level là info thì nó chỉ log cái logger.info, cái logger.debug thì k được log ra.

Nếu root level là warn thì chương trình hiện tại không print gì cả, vì không có error hay warning gì ở chương trình HelloWorld này cả.

Cách sử dụng file log trong Daito:

<configuration>

<property name=*"thuat.log.file"* value=*"thuat-test.log"*/>

<property name=*"message.pattern"* value=*"%d [%thread] %level %mdc %logger{0} %C %M %ex - %msg%n"*/>

<appender name=*"FILE"*

class=*"ch.qos.logback.core.rolling.RollingFileAppender"*>

<File>${thuat.log.file}</File>

<encoder>

<Pattern>${message.pattern}</Pattern>

</encoder>

<rollingPolicy class=*"ch.qos.logback.core.rolling.TimeBasedRollingPolicy"*>

<FileNamePattern>${thuat.log.file}.%d{yyyy-MM-dd-HH}

</FileNamePattern>

</rollingPolicy>

</appender>

<appender name=*"CONSOLE"* class=*"ch.qos.logback.core.ConsoleAppender"*>

<encoder>

<pattern>${message.pattern}</pattern>

</encoder>

</appender>

<!-- Ví dụ này là version 1, cách làm thông thường -->

<!-- <appender name="FILE" class="ch.qos.logback.core.FileAppender">

<file>slf4j-example.log</file>

<encoder>

<pattern>%date %level [%thread] %logger{10} [%file:%line] %msg%n</pattern>

</encoder>

</appender> -->

<root level=*"debug"*>

<appender-ref ref=*"FILE"* />

</root>

</configuration>

Với TimeBasedRollingPolicy quy định ở đây tối đa cho 1 file log là 1 tiếng.

ConsoleAppender thì để in ra console, mà hình như ở đay chỉ sử dụng command line mới thấy được.

Kết quả:



Trang web rất hay để tìm hiểu về slf4f:

<http://logback.qos.ch/manual/appenders.html>

Trong ví dụ trên ở Daito có xài cái ${project.build.directory}, cái này được define trong file maven pom.xml. Nhưng chưa hiểu lắm cách define

## 3.9 @Component

Ta có thể sử dụng @Component thay thế cho 3 annotation còn lại là @Repository, @Service và @Controller. Spring sẽ cho phép scan tất cả với @Component, nó hoạt động tốt nhưng sẽ không tối ưu, nó sẽ không catch exception liên quan tới DAO thay vì Repository

<https://huongdanjava.com/auto-component-scan-trong-spring.html>

## 3.10 Assert trong Spring

Cái này dùng để kiểm tra nhanh dữ liệu đầu vào, ví dụ như notNull, hasText…

**public** **class** AssertUtils **extends** Assert {

**public** **static** **void** notNull(**final** Object o, **final** String message, **final** Object... args) {

**if** (o == **null**) {

**throw** **new** IllegalArgumentException(String.*format*(message, args));

}

}

}

## 3.11 Singleton Pattern in Java

**package** com.test;

**public** **class** Database {

// private constructor của Database để không thể khởi tạo nó ở ngoài

**private** Database() {

}

// tạo ra 1 instance duy nhất, phải để static để method này thành method của lớp.

**private** **static** Database *INSTANCE*;

**public** **static** Database getInstance(){

**if** (*INSTANCE* == **null**) {

*INSTANCE* = **new** Database();

}

**return** *INSTANCE*;

}

**private** **boolean** offSet = **false**;

**public** **static** **boolean** isOffset(){

**return** *INSTANCE*.offSet;

}

}

Chạy hàm main Test:

**package** com.test;

**public** **class** Test {

**public** **static** **void** main(String[] args) {

System.***out***.println(Database.*getInstance*().*isOffset*());

}

}

Kết quả:



## 3.12 Sử dụng InitializingBean và DisposableBean

2 interface này bắt buộc user phải overide lại 2 method là:

1. For bean implemented InitializingBean, it will run afterPropertiesSet() after all bean properties have been set.
2. For bean implemented DisposableBean, it will run destroy() after Spring container is released the bean.

LegacyDataConfig.java:

**package** com.initializingbean;

**import** org.springframework.beans.factory.DisposableBean;

**import** org.springframework.beans.factory.InitializingBean;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Component;

@Component

**public** **class** LegacyDataConfig **implements** InitializingBean, DisposableBean {

String message;

**public** String getMessage() {

**return** message;

}

@Autowired

**public** **void** setMessage(String message) {

**this**.message = message;

}

@Override

**public** **void** afterPropertiesSet() **throws** Exception {

System.***out***.println("Init method after properties are set");

}

@Override

**public** **void** destroy() **throws** Exception {

System.***out***.println("Spring Container is destroy! Customer clean up");

}

}

Spring-Customer.xml:

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans*

*http://www.springframework.org/schema/beans/spring-beans-4.0.xsd"*>

<bean id=*"legacyDataConfig"* class=*"com.initializingbean.LegacyDataConfig"*>

<property name=*"message"* value=*"i'm property message"* />

</bean>

</beans>

App.java:

**package** com.initializingbean;

**import** org.springframework.context.ConfigurableApplicationContext;

**import** org.springframework.context.support.ClassPathXmlApplicationContext;

**public** **class** App {

**public** **static** **void** main( String[] args )

{

ConfigurableApplicationContext context =

**new** ClassPathXmlApplicationContext("Spring-Customer.xml");

LegacyDataConfig legacy = (LegacyDataConfig)context.getBean("legacyDataConfig");

System.***out***.println(legacy.message);

context.close();

}

}

## 3.13 Khác biệt giữa ApplicationContext và BeanFactory:

<https://huongdanjava.com/beanfactory-va-applicationcontext-trong-sping.html>

Cơ bản nhất là ApplicationContext extends từ interface Beanfactory nhưng ApplicationContext sẽ khởi tạo mọi đối tượng trong khung chứa bean ngay khi gọi đến, trong khi BeanFactory chỉ tạo ra đối tượng cần khi gọi getBean().

## 3.14 Sử dụng ApplicationContextAware và BeanNameAware

Cái ApplicationContextAware này trong project Daito nó có ghi một câu:

The class is created to reference Spring ApplicatinContext in a static way. This should be used in those classes that can't be configured as Spring beans, but require accessing some Spring beans.

Thấy câu này không đúng lắm. Đại khái cái thằng ApplicationContextAware này dùng để tạo ra ApplicationContext, tạo một hay nhiều cái đều được, tạo bất cứ đâu. Nếu có một cái class nào đó k được cấu hình trong khung Bean, nhưng lại cần sử dụng mấy cái instance trong Bean thì cứ lôi thằng này ra sử dụng.

Ví dụ này nằm trong project org.arpit.javapostsforlearning

Ví dụ này mình lấy trên trang này:

<http://www.java2blog.com/2012/08/spring-applicationcontext.html>

## 3.15 Junit:

<http://laptrinh.vn/d/4178-tim-hieu-ve-testing-junit.html>



Class Money:

**package** com.junit;

**public** **class** Money {

**private** **double** amount;

**private** String currency;

**public** Money(**double** amount, String currency) {

**this**.amount = amount;

**this**.currency = currency;

}

**public** Money add(Money money){

Money m = **new** Money(money.amount+amount, currency);

**return** m;

}

}

Class MoneyTest:

**package** com.junit;

**import** junit.framework.~~Assert~~;

**import** junit.framework.TestCase;

**public** **class** MoneyTest **extends** TestCase{

**public** **void** testAdd(){

Money m1 = **new** Money(200, "VND");

Money m2 = **new** Money(1000, "VND");

Money result = m1.add(m2); // đối tượng lưu kết quả tính toán

Money expected = **new** Money(1200, "VND"); // kết quả dự kiến

~~Assert~~.~~assertEquals~~(result, expected);

}

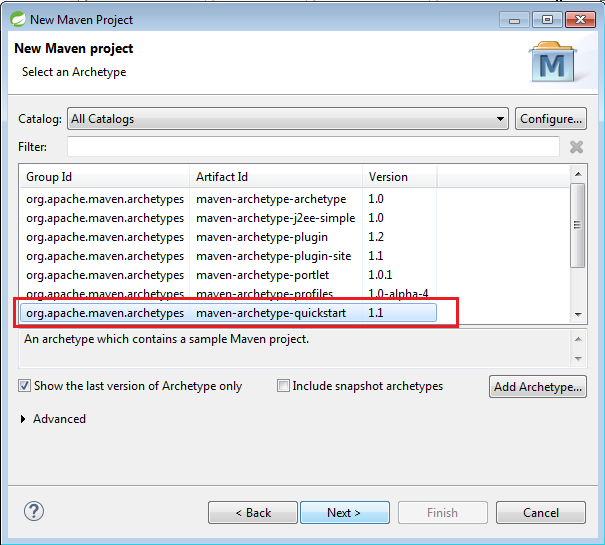
}

Right click vào testAdd, run Junit t được fail vì 2 object này không giống nhau:



# 4. Maven Project

<http://o7planning.org/vi/10127/huong-dan-lap-trinh-spring-cho-nguoi-moi-bat-dau#a720908>



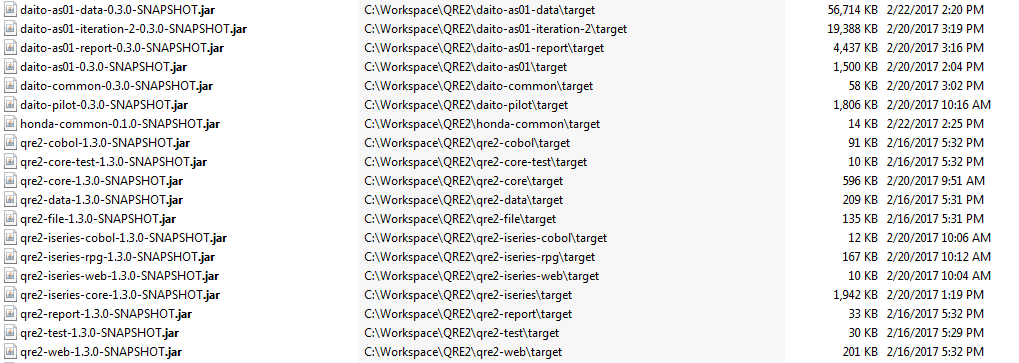
Dùng cái hướng dẫn này mình sử dụng được file pom để down các thư viện về

Có 1 thủ thuật mentor làm mà k biết để làm gì:

Right click -> Maven -> Update project:



Sau khi build project ra được các file SNAPSHOT sau đây:



Remote System explorer Operation cứ chạy hoài, k biết có nên tắt k

# 5. Các lưu ý cho Project Daito:

\_ Column nếu là String thì k được NULL, nếu NULL thì thêm ký tự single space vào.

\_ Column nếu là numberic thì k được NULL, nếu NULL thì thêm số 0 vào.

## EBCDIC:

(Extended Binary Coded Decimal Interchange Code). Nó là mã có 8 cột, không có cột kiểm tra (như bảng 1.4).

|  |  |  |
| --- | --- | --- |
| **Table of EBCDIC character set Document Produced by Jim Plaxco** | | |
| decimal value | hex value | Character |
| 000 | 00 | NUL |
| 001 | 01 | SOH |
| 002 | 02 | STX |
| 003 | 03 | ETX |
| 004 | 04 | PF |
| 005 | 05 | HT |
| 006 | 06 | LC |
| 007 | 07 | DEL |
| 008 | 08 | GE |
| 009 | 09 | RLF |
| 010 | 0A | SMM |
| 011 | 0B | VT |
| 012 | 0C | FF |
| 013 | 0D | CR |
| 014 | 0E | SO |
| 015 | 0F | SI |
| 016 | 10 | DLE |
| 017 | 11 | DC1 |
| 018 | 12 | DC2 |
| 019 | 13 | TM |
| 020 | 14 | RES |
| 021 | 15 | NL |
| 022 | 16 | BS |
| 023 | 17 | IL |
| 024 | 18 | CAN |
| 025 | 19 | EM |
| 026 | 1A | CC |
| 027 | 1B | CU1 |
| 028 | 1C | IFS |
| 029 | 1D | IGS |
| 030 | 1E | IRS |
| 031 | 1F | IUS |
| 032 | 20 | DS |
| 033 | 21 | SOS |
| 034 | 22 | FS |
| 035 | 23 |  |
| 036 | 24 | BYP |
| 037 | 25 | LF |
| 038 | 26 | ETB |
| 039 | 27 | ESC |
| 040 | 28 |  |
| 041 | 29 |  |
| 042 | 2A | SM |
| 043 | 2B | CU2 |
| 044 | 2C |  |
| 045 | 2D | ENQ |
| 046 | 2E | ACK |
| 047 | 2F | BEL |
| 048 | 30 |  |
| 049 | 31 |  |
| 050 | 32 | SYN |
| 051 | 33 |  |
| 052 | 34 | PN |
| 053 | 35 | RS |
| 054 | 36 | UC |
| 055 | 37 | EOT |
| 056 | 38 |  |
| 057 | 39 |  |
| 058 | 3A |  |
| 059 | 3B | CUB |
| 060 | 3C | DC4 |
| 061 | 3D | NAK |
| 062 | 3E |  |
| 063 | 3F | SUB |
| 064 | 40 | BLANK |
| 065 | 41 |  |
| 066 | 42 |  |
| 067 | 43 |  |
| 068 | 44 |  |
| 069 | 45 |  |
| 070 | 46 |  |
| 071 | 47 |  |
| 072 | 48 |  |
| 073 | 49 |  |
| 074 | 4A | ¢ |
| 075 | 4B | . |
| 076 | 4C | < |
| 077 | 4D | ( |
| 078 | 4E | + |
| 079 | 4F | | |
| 080 | 50 |  |
| 081 | 51 |  |
| 082 | 52 |  |
| 083 | 53 |  |
| 084 | 54 |  |
| 085 | 55 |  |
| 086 | 56 |  |
| 087 | 57 |  |
| 088 | 58 |  |
| 089 | 59 |  |
| 090 | 5A | ! |
| 091 | 5B | $ |
| 092 | 5C | \* |
| 093 | 5D | ) |
| 094 | 5E | ; |
| 095 | 5F | ¬ |
| 096 | 60 | - |
| 097 | 61 | / |
| 098 | 62 |  |
| 099 | 63 |  |
| 100 | 64 |  |
| 101 | 65 |  |
| 102 | 66 |  |
| 103 | 67 |  |
| 104 | 68 |  |
| 105 | 69 |  |
| 106 | 6A | ¦ |
| 107 | 6B | , |
| 108 | 6C | % |
| 109 | 6D | \_ |
| 110 | 6E | > |
| 111 | 6F | ? |
| 112 | 70 |  |
| 113 | 71 |  |
| 114 | 72 |  |
| 115 | 73 |  |
| 116 | 74 |  |
| 117 | 75 |  |
| 118 | 76 |  |
| 119 | 77 |  |
| 120 | 78 |  |
| 121 | 79 | ` |
| 122 | 7A | : |
| 123 | 7B | # |
| 124 | 7C | @ |
| 125 | 7D | ' |
| 126 | 7E | = |
| 127 | 7F | " |
| 128 | 80 |  |
| 129 | 81 | a |
| 130 | 82 | b |
| 131 | 83 | c |
| 132 | 84 | d |
| 133 | 85 | e |
| 134 | 86 | f |
| 135 | 87 | g |
| 136 | 88 | h |
| 137 | 89 | i |
| 138 | 8A |  |
| 139 | 8B |  |
| 140 | 8C |  |
| 141 | 8D |  |
| 142 | 8E |  |
| 143 | 8F |  |
| 144 | 90 |  |
| 145 | 91 | j |
| 146 | 92 | k |
| 147 | 93 | l |
| 148 | 94 | m |
| 149 | 95 | n |
| 150 | 96 | o |
| 151 | 97 | p |
| 152 | 98 | q |
| 153 | 99 | r |
| 154 | 9A |  |
| 155 | 9B |  |
| 156 | 9C |  |
| 157 | 9D |  |
| 158 | 9E |  |
| 159 | 9F |  |
| 160 | A0 |  |
| 161 | A1 | ~ |
| 162 | A2 | s |
| 163 | A3 | t |
| 164 | A4 | u |
| 165 | A5 | v |
| 166 | A6 | w |
| 167 | A7 | x |
| 168 | A8 | y |
| 169 | A9 | z |
| 170 | AA |  |
| 171 | AB |  |
| 172 | AC |  |
| 173 | AD |  |
| 174 | AE |  |
| 175 | AF |  |
| 176 | B0 |  |
| 177 | B1 |  |
| 178 | B2 |  |
| 179 | B3 |  |
| 180 | B4 |  |
| 181 | B5 |  |
| 182 | B6 |  |
| 183 | B7 |  |
| 184 | B8 |  |
| 185 | B9 |  |
| 186 | BA |  |
| 187 | BB |  |
| 188 | BC |  |
| 189 | BD |  |
| 190 | BE |  |
| 191 | BF |  |
| 192 | C0 | { |
| 193 | C1 | A |
| 194 | C2 | B |
| 195 | C3 | C |
| 196 | C4 | D |
| 197 | C5 | E |
| 198 | C6 | F |
| 199 | C7 | G |
| 200 | C8 | H |
| 201 | C9 | I |
| 202 | CA |  |
| 203 | CB |  |
| 204 | CC | non-displayable |
| 205 | CD |  |
| 206 | CE | non-displayable |
| 207 | CF |  |
| 208 | D0 | } |
| 209 | D1 | J |
| 210 | D2 | K |
| 211 | D3 | L |
| 212 | D4 | M |
| 213 | D5 | N |
| 214 | D6 | O |
| 215 | D7 | P |
| 216 | D8 | Q |
| 217 | D9 | R |
| 218 | DA |  |
| 219 | DB |  |
| 220 | DC |  |
| 221 | DD |  |
| 222 | DE |  |
| 223 | DF |  |
| 224 | E0 | \ |
| 225 | E1 |  |
| 226 | E2 | S |
| 227 | E3 | T |
| 228 | E4 | U |
| 229 | E5 | V |
| 230 | E6 | W |
| 231 | E7 | X |
| 232 | E8 | Y |
| 233 | E9 | Z |
| 234 | EA |  |
| 235 | EB |  |
| 236 | EC | non-displayable |
| 237 | ED |  |
| 238 | EE |  |
| 239 | EF |  |
| 240 | F0 | 0 |
| 241 | F1 | 1 |
| 242 | F2 | 2 |
| 243 | F3 | 3 |
| 244 | F4 | 4 |
| 245 | F5 | 5 |
| 246 | F6 | 6 |
| 247 | F7 | 7 |
| 248 | F8 | 8 |
| 249 | F9 | 9 |
| 250 | FA | non-displayable |
| 251 | FB |  |
| 252 | FC |  |
| 253 | FD |  |
| 254 | FE |  |
| 255 | FF | EO |

# 5.1 HSQL:

**HSQLDB là** một database đơn giản, viết bằng Java. **Là** loại database quan hệ giống MySQL, Oracle, SQL Server.

Web hướng dẫn HSQLDB:

<http://o7planning.org/vi/10287/huong-dan-cai-dat-va-cau-hinh-database-hsqldb>

<http://o7planning.org/vi/10203/cau-hinh-hsqldb-datasource-su-dung-data-source-explorer>

## Lệnh tạo table:

**CREATE** MEMORY **TABLE** PUBLIC.DEPARTMENT(DEPT\_ID **INTEGER** **NOT** **NULL** **PRIMARY** **KEY**,DEPT\_NAME **VARCHAR**(255) **NOT** **NULL**,DEPT\_NO **VARCHAR**(20) **NOT** **NULL**,LOCATION **VARCHAR**(255),**UNIQUE**(DEPT\_NO))

## Lệnh insert:

**INSERT** **INTO** DEPARTMENT **VALUES**(10,'ACCOUNTING','D10','NEW YORK')

## Table Employee với Constraint:

**CREATE** MEMORY **TABLE** PUBLIC.EMPLOYEE(EMP\_ID BIGINT **NOT** **NULL** **PRIMARY** **KEY**,EMP\_NAME **VARCHAR**(50) **NOT** **NULL**,EMP\_NO **VARCHAR**(20) **NOT** **NULL**,HIRE\_DATE **DATE** **NOT** **NULL**,IMAGE BLOB(1G),JOB **VARCHAR**(30) **NOT** **NULL**,SALARY **DOUBLE** **NOT** **NULL**,DEPT\_ID **INTEGER** **NOT** **NULL**,MNG\_ID BIGINT,**UNIQUE**(EMP\_NO),**CONSTRAINT** FK75C8D6AE269A3C9 **FOREIGN** **KEY**(DEPT\_ID) **REFERENCES** PUBLIC.DEPARTMENT(DEPT\_ID),**CONSTRAINT** FK75C8D6AE6106A42 **FOREIGN** **KEY**(EMP\_ID) **REFERENCES** PUBLIC.EMPLOYEE(EMP\_ID),**CONSTRAINT** FK75C8D6AE13C12F64 **FOREIGN** **KEY**(MNG\_ID) **REFERENCES** PUBLIC.EMPLOYEE(EMP\_ID))

Không hiểu sao không thêm data được cho table này:

## Sequence:

**CREATE** SEQUENCE AJE1FP\_SEQ START **WITH** 1;

Tạo ra 1 cái rule cho mấy loại:

SMALLINT, INTEGER, BIGINT, DECIMAL and NUMERIC

sequence generator definition

<sequence generator definition> ::= CREATE SEQUENCE [ IF NOT EXISTS ] <sequence generator name> [ <sequence generator options> ]

<sequence generator options> ::= <sequence generator option> ...

<sequence generator option> ::= <sequence generator data type option> | <common sequence generator options>

<common sequence generator options> ::= <common sequence generator option> ...

<common sequence generator option> ::= <sequence generator start with option> | <basic sequence generator option>

<basic sequence generator option> ::= <sequence generator increment by option> | <sequence generator maxvalue option> | <sequence generator minvalue option> | <sequence generator cycle option>

<sequence generator data type option> ::= AS <data type>

<sequence generator start with option> ::= START WITH <sequence generator start value>

<sequence generator start value> ::= <signed numeric literal>

<sequence generator increment by option> ::= INCREMENT BY <sequence generator increment>

<sequence generator increment> ::= <signed numeric literal>

<sequence generator maxvalue option> ::= MAXVALUE <sequence generator max value> | NO MAXVALUE

<sequence generator max value> ::= <signed numeric literal>

<sequence generator minvalue option> ::= MINVALUE <sequence generator min value> | NO MINVALUE

<sequence generator min value> ::= <signed numeric literal>

<sequence generator cycle option> ::= CYCLE | NO CYCLE

## Khác nhau giữa Decimal và BigDecimal



Decimal là int, double, long bình thường. Còn BigDecimal là 1 object, tính toán số lớn và tiền tệ. Tuy nhiên BigDecimal chậm.

# 6 Add code cho etes

Làm nhanh quá k kịp nhìn nhưng tạm thời là vầy:

Chọn regular time

Project nhập mã code vào

Show project

Show Billing, chọn số nào đó (lúc anh Hoàng đưa thì chọn 000)

# 7. Tìm hiểu Project Daito:

# 7.1 Project qre2-core:

Project nằm ở địa chỉ svn: 20.194.10.15/svn/java-csc-qre-core/trunk

## com.csc.qre.core.datatype

### NumberDeclarationSupport.java:

Khai báo của nó như thế này:

**public** **interface** NumberDeclarationSupport<T **extends** LegacyNumber> **extends** DeclarationSupport<T> {…}

T là 1 generic, và T phải extends từ LegacyNumber.

Define các behaviours các data type có liên quan về numeric.

Gồm 6 method:

1. T init(**int** value);

2. T init(**long** value);

3. T init(**double** value);

4. T init(PredefinedConstant value);

5. T setUnsigned(**final** **boolean** unsigned);

6. T setUnsigned();

### LegacyBoolean:



**public** **class** LegacyBooleanImpl **extends** AbstractString **implements** LegacyBoolean {…}

**private** **static** **final** **int** ***FIX\_LENGTH*** = NumberConstants.***I1***; Quy định độ lớn: 1 bit.

**private** **static** **final** String ***TRUE\_STR*** = Character.*toString*(***TRUE***);

**private** **static** **final** String ***FALSE\_STR*** = Character.*toString*(***FALSE***);

Không chắc lắm nhưng hình như quy định TRUE\_STR là 1, FALSE\_STR là 0.

…

### LegacyInteger:



**public** **interface** LegacyInteger **extends** LegacyNumber, NumberDeclarationSupport<LegacyInteger> {…}

LegacyInteger tương tự như Binary, cũng quy định về digit, nhưng ở đây không quy định về byte.

**int** ***LONG\_DIGITS*** = Binary.***LONG\_DIGITS***; 20 digits

**int** ***INT\_DIGITS*** = Binary.***INT\_DIGITS***; 10 digits

**int** ***SHORT\_DIGITS*** = Binary.***SHORT\_DIGITS***; 5 digits

**int** ***TINY\_DIGITS*** = 3; 3 digits, viết code thật vl, đáng lẽ phải là Binary.***TINY\_DIGITS*** chứ, tự nhiên cái viết số 3 ra luôn là sao?

**int**[] ***MAX\_DIGITS*** = { ***TINY\_DIGITS***, ***SHORT\_DIGITS***, ***INT\_DIGITS***, ***LONG\_DIGITS*** };

**int** ***MAX\_DIGITS\_SUPPORTED*** = ***LONG\_DIGITS***;

### PackedDecimal



#### 1. Constructor PackedDecimalImpl

**public** PackedDecimalImpl(**final** **int** precision, **final** **int** scale) {

**super**(precision, scale);

bytes = LegacyNumberUtils.*workoutBytesOfPackedDecimal*(length());

}

Ở đây precision là tổng số chữ số, scale là số sau dấu phẩy (gọi là scale, decimal place, decimal,…) như ta đã biết.

bytes là tổng số byte để lưu số PackedDecimal này. Muốn gọi bytes này dùng hàm **lengthInBytes()** là hàm inherit từ AbstractNumber.

**public** PackedDecimalImpl(**final** **int** precision) {

**this**(precision, 0);

}

Xài lại hàm Constructor, nhưng cho scale bằng 0.

#### 2. Hàm init()

**public** PackedDecimal init(**final** **int** value) {

set(value);

**return** **this**;

}

@Override

**public** PackedDecimal init(**final** **long** value) {

set(value);

**return** **this**;

}

@Override

**public** PackedDecimal init(**final** **double** value) {

set(value);

**return** **this**;

}

@Override

**public** PackedDecimal init(**final** PredefinedConstant value) {

set(value);

**return** **this**;

}

Có 4 hàm init với argument khác nhau. Cùng phân tích hàm init đầu tiên.

Ví dụ với hàm init(int) như sau:

**public** **class** ZZZTest {

**public** **static** **void** main(String[] args) {

PackedDecimalImpl p = **new** PackedDecimalImpl(13, 5);

System.***out***.println(p.lengthInBytes()); // Trả về số byte lưu trữ PackedDecimal, ở đây là 7

PackedDecimal p2 = p.init(150);

System.***out***.println(p2.getInteger());

System.***out***.println("--------End--------");

}

}

Trước khi thực hiện hàm init thì p sẽ như sau:



Hàm .init(150) đây là hàm init(int).

Với PackedDecimalImpl này thì hàm init sẽ trả lại hàm init(long):

@Override

**public** **void** set(**final** **long** value) {

**if** (hasDecimal) {

setUnscaled(value \* decimalMult, **false**);

} **else** {

setUnscaled(value, **false**);

}

}

hasDecimal = true. Nên sẽ chạy vào hàm setUnscaled. decimalMult là 10^5, vì scale = 5.

Với việc mình truyền vào value = 150. Vì thế value \* decimalMult = 15000000 hay 150.10^5

Vì thế sau khi chạy xong init(150) thì unscaledValue sẽ bằng 150.10^5. Để gọi ra cái unscaledValue thì gọi p.unscaledValue vì cái unscaledValue được khai báo là protected, nên có thể sử dụng trong class, package và subclass.

Ở đây sẽ nhảy vào hàm setUnscaled.

@Override

**protected** **final** **void** setUnscaled(**final** **long** unscaled, **final** **boolean** decimalTruncated) {

resetFlagForSetValue();

**final** **long** unscaledToSet;

**if** (unsigned && unscaled < 0) {

unscaledToSet = -unscaled;

} **else** {

unscaledToSet = unscaled;

}

**if** (unscaledToSet >= precisionMult || unscaledToSet <= -precisionMult) {

truncated = **true**;

unscaledValue = unscaledToSet % precisionMult;

} **else** {

unscaledValue = unscaledToSet;

truncated = decimalTruncated;

}

**if** (inStructure) {

updateParent();

}

}

Ở đây inStructure là false nên không updateParent gì cả. Nếu giá trị trong init mà lớn hơn integer(precison – scale), thì sẽ bị truncated.

#### Tổng kết hàm

**public** **class** ZZZTest {

**public** **static** **void** main(String[] args) {

PackedDecimalImpl p = **new** PackedDecimalImpl(13, 5);

System.***out***.println(p.lengthInBytes()); // Trả về số byte lưu trữ PackedDecimal, ở đây là 7

}

}

Kết quả ta được p như sau:



### LegacyData

**public** **interface** LegacyData **extends** Variable, DataStructureSupport, CompareSupport, Serializable {…}

Implements DataStructureSupport (except lengthInBytes) as it depends on the specific data type.

LegacyData extends DataStructureSupport nên nó sẽ có tính chất của interface DataStructureSupport, tùy thuộc vào từng loại data type cụ thể.

Implements compareTo and set methods:

* compareTo LegacyData, LegacyString, LegacyNumber, Number
* compareTo LegacyDate, LegacyTime, LegacyTimestamp
* compareTo LegacyConstant, LegacyPointer
* set LegacyData, Number, Object
* set LegacyDate, LegacyTime, LegacyTimestamp
* set LegacyConstant

Support Pointer Operations

* set to a LegacyPointer
* set to address of a LegacyData

Có tổng cộng 28 method:

1. **void** set(**final** Object value);
2. **void** set(**final** LegacyData value);
3. **void** set(LegacyNumber value);
4. **void** set(LegacyString value);
5. **void** set(LegacyPointer value);
6. **void** set(LegacyDate value);
7. **void** set(LegacyTime value);
8. **void** set(LegacyTimestamp value);
9. **void** set(**final** LegacyConstant value);
10. **void** set(**final** **int** value);
11. **void** set(**final** **long** value);
12. **void** set(**final** **double** value);
13. **void** set(**final** BigDecimal value);
14. **void** set(Number value);
15. **void** set(String value);
16. **void** set(**final** **boolean** value);
17. **void** set(Date value);
18. String toString();
19. **void** clear();
20. **void** setAddress(LegacyPointer pointer);
21. **void** setAddress(LegacyData aData);
22. **void** setInternal(**char**[] value);
23. **char**[] toInternal();
24. **int** length();
25. **int** lengthInBytes();
26. Object value();
27. **boolean** isTruncated();
28. LegacyConstant getSpecialValue();

### ZonedDecimalImpl

Địa chỉ a.Ngôn dùng khi giảng:

<http://www.simotime.com/datazd01.htm>

Địa chỉ hay tìm được: <http://publib.boulder.ibm.com/iseries/v5r2/ic2924/books/c0925083170.htm>

Trong java, double hay long chỉ nên dùng 15 digit để lưu. Vì thế ZonedDecimal và packedDecimal chỉ nên dùng tối đa cho số có 15 digit, nếu lơn hơn phải dùng BigDecimal (BigZonedDecimal, BigPackedDecimal).

**public** **class** ZonedDecimalImpl **extends** AbstractDecimal **implements** ZonedDecimal {…}

#### Phân tích hàm constructor của ZonedDecimalImpl:

**public** ZonedDecimalImpl(**final** **int** precision, **final** **int** scale) {

**super**(precision, scale);

}

precision: độ chính xác

scale: tỉ lệ.

Theo sự nghiên cứu trên mạng thì precision thì là number of digits, còn scale chính là decimal place:

Ví dụ 1234.567

Precision: 7

Scale: 3

Hàm constructor inherit từ hàm constructor của AbstractDecimal:

**public** AbstractDecimal(**final** **int** precision, **final** **int** scale) {

**super**(precision, scale);

**if** (precision > ***LARGE\_NUMBER\_DIGITS***) {

**throw** **new** InvalidValueException(

"Precision is too large. Please use the implementation for large numbers, which should be subclasses of "

+ getClass() + ' ' + precision);

}

}

Hàm constructor này cũng inherit từ hàm của hàm AbstractNumber, tuy nhiên kèm theo quăng ra exception là precision đã vượt LARGE\_NUMBER\_DIGITS.

Constructor của AbstractNumber:

**public** AbstractNumber(**final** **int** precision, **final** **int** scale) {

**super**(precision);

**this**.integer = precision - scale;

**this**.precisionMult = MathUtils.*powerOfTen*(precision);

**this**.integerMult = MathUtils.*powerOfTen*(integer);

**this**.scale = scale;

hasDecimal = scale > 0;

**if** (hasDecimal) {

decimalMult = MathUtils.*powerOfTen*(scale);

} **else** {

decimalMult = NumberConstants.***I1***;

}

bigNumber = precision > ***LARGE\_NUMBER\_DIGITS***;

**if** (scale < 0 || integer < 0) {

**throw** **new** InvalidValueException("precision = " + precision + " & scale=" + scale + " is invalid");

}

// By default, length in bytes is the same as total length

bytes = precision;

}

integer = precision – scale

scale = scale

…

**Ví dụ:**



Tuy nhiên hàm constructor của AbstractNumber còn inherit từ hàm constructor của AbstractLegacyData:

**public** AbstractLegacyData(**final** **int** length) {

**if** (length <= 0) {

**this**.length = 0;

} **else** {

**this**.length = length;

}

}

Hàm này đơn giản là cho cái length bằng tham số (ở đây là precision).

#### Phân tích hàm setPattern của ZonedDecimalImpl:



#### Phân tích hàm isAPartOf của ZonedDecimalImpl

@Override

**public** ZonedDecimal isAPartOf(**final** DataStructureSupport parent) {

asAPartOf(parent, **false**);

**return** **this**;

}

Hàm gì khó như quỷ, khó hiểu vãi nồi.

***QUYẾT TÂM TÌM HIỂU NÀO!!!***

IsAPartOf này có hàm asAPartOf của AbstractLegacyData.java:

**protected** **final** **void** asAPartOf(**final** DataStructureSupport parent, **final** **boolean** retainValue) {

**if** (LegacyDataConfig.*isAutoOffset*()) {

**int** offset = LegacyDataUtils.*getOffset*(parent);

asAPartOf(parent, offset, retainValue);

LegacyDataUtils.*setNextOffset*(parent, offset + lengthInBytes());

} **else** {

asAPartOf(parent, 0, retainValue);

}

}

Ở đây cái khó hiểu nhất là hàm LegacyDataConfig.isAutoOffset(). LegacyDataConfig là class theo kiểu singleton, isAutoOffset là method static, tuy nhiên ở đây chỉ có declaration thôi thì làm sao mà xài?

**int** offset = LegacyDataUtils.*getOffset*(parent); Cái này kệ bà nó, biết là nó trả giá trị int gì đó về từ system.

Từ việc có được thằng offset, nó sẽ gọi về cái hàm overload asAPartOf có nhiều tham số hơn.

**protected** **final** **void** asAPartOf(**final** DataStructureSupport parent, **final** **int** offset, **final** **boolean** retainValue) {

AssertUtils.*notNull*(parent, "parent array must not be null");

// Safety checking, CAN't make a data is a part of itself

**if** (**this** == parent) {

**throw** **new** InvalidValueException("Can't make a data is a part of itself");

}

**final** Object current = retainValue ? value() : **null**;

// Indicates the parent is in a data structure as well.

**if** (!parent.isInStructure()) {

parent.setInStrucutre(**true**, **true**);

}

// Indicates the data is a part of a data structure

**if** (wrappedData == **null**) {

wrappedData = WrappedDataFactory.*createWrappedData*(parent, offset);

} **else** {

// When dynamic datastructure is not supported, then once a field is made a part of structure, then it

// can't be changed as it might break linking between this field and its children

**if** (inStructure && !LegacyDataConfig.*isDynamicStructureSupported*()) {

**throw** **new** InvalidOperationException(

"Can't change parent as the variable has been in a structure and dynamic structure is not supported");

}

wrappedData.setParentData(parent.getStructureData(), offset);

}

// - indicating the parent has at least a child

parent.setHasChildren();

setInStrucutre(**true**, **false**);

hasParent = **true**;

**final** **int** bytes = lengthInBytes();

// Validate offset and bytes

**if** (!ArrayUtils.*validateOffsetAndLength*(wrappedData.getData(), offset, bytes)) {

**throw** **new** InvalidValueException("offset[" + offset + "] & length[" + bytes

+ "] are not valid to parent which has length =" + wrappedData.getData().length);

}

**if** (retainValue) {

set(current);

}

}

AssertUtils.*notNull*(parent, "parent array must not be null");

Đầu tiên dùng assert của spring kiểm tra giá trị đầu vào không được null.

**if** (**this** == parent) {

**throw** **new** InvalidValueException("Can't make a data is a part of itself");

Sau đó kiểm tra cái thằng đang xài isAPartOf này có phải cùng cấp AbstractLegacyData không? Nếu phải thì throw ra exception.

## ZonedDecimalPatternWorker

**public** **interface** ZonedDecimalPatternWorker {…}

**char**[] applyPattern(**final** **char**[] strNumber, **final** **int** scale, **final** String pattern, **final** Sign sign);

String removePattern(**final** **char**[] noDigits, **int** scale, **final** String pattern, **boolean** isUnsigned);

**boolean** isPatternIgnorable(String pattern);

PreparedPattern preparePattern(ZonedDecimal number, String pattern);

**public** **static** **class** PreparedPattern {

**private** **final** String pattern;

**private** **final** **boolean** unsigned;

**public** PreparedPattern(**final** String pattern, **final** **boolean** unsigned) {

**super**();

**this**.pattern = pattern;

**this**.unsigned = unsigned;

}

**public** String getPattern() {

**return** pattern;

}

**public** **boolean** isUnsigned() {

**return** unsigned;

}

}

Ta thấy có 3 method applyPattern, removePattern, isPatternIgnorable

Trong interface này còn có thêm 1 cái class inner, chỉ gồm có String pattern và **boolean** unsigned

## com.csc.qre.core.ulti

### MathUtils

**public** **final** **class** MathUtils {…}

Vì class này khai báo final, nên có thể lôi nó ra sử dụng ở mọi nên, chỉ cần import cái package là được, ví dụ:

**his**.precisionMult = MathUtils.*powerOfTen*(precision);

**Phân tích hàm powerOfTen:**

**public** **static** **long** powerOfTen(**final** **int** exponent) {

**if** (exponent > 0 && exponent < ***POWER\_OF\_TEN\_POSITIVE***.length) {

**return** ***POWER\_OF\_TEN\_POSITIVE***[exponent];

} **else** {

**return** (**long**) Math.*pow*(***BASE\_TEN***, exponent);

}

}

Ở đây sử dụng array static POWER\_OF\_TEN\_POSITIVE:

**private** **static** **final** **long**[] ***POWER\_OF\_TEN\_POSITIVE*** = *initPowerOfTenPositive*(***CACH\_OF\_POWER\_TEN\_POSITIVE***);

Đã vậy POWER\_OF\_TEN\_POSITIVE lại còn sử dụng hàm initPowerOfTenPositive và

***CACH\_OF\_POWER\_TEN\_POSITIVE =*** LegacyNumber.***LARGE\_NUMBER\_DIGITS*** + 1 = 16

**Hàm initPowerOfTenPositive:**

**private** **static** **long**[] initPowerOfTenPositive(**final** **int** maxExponent) {

**long**[] positives = **new** **long**[maxExponent];

**for** (**int** i = 0; i < maxExponent; i++) {

positives[i] = (**long**) Math.*pow*(***BASE\_TEN***, i);

}

**return** positives;

}

ở đây MathUtils quy định ***BASE\_TEN = 10;***

initPowerOfTenPositive sẽ trả về array có length chính bằng maxExponent: 1, 10, 100, …, 10^(maxExponent -1).

ở đây ***CACH\_OF\_POWER\_TEN\_POSITIVE = 16,*** nên hiển nhiên ***POWER\_OF\_TEN\_POSITIVE*** là 1 array 16 phần tử đã nói ở trên.



Giờ hàm powerOfTen có kiểu dũ liệu là long, return ***POWER\_OF\_TEN\_POSITIVE***[exponent];

Nghĩa là dựa vào cái exponent sẽ lấy ra phần tử nào trong ***POWER\_OF\_TEN\_POSITIVE***

## Com.csc.qre.core.datatype.constant

### LegacyCategory

Khai báo: **public** **enum** LegacyCategory {…}

Enum này có các kiểu sau:

***NUMERIC***("^[-\\+]?\\d+$", LegacyNumber.**class**)

***ALPHANUMERIC***("^[a-zA-Z0-9\u0020]+$", LegacyString.**class**) {

***ALPHABETIC***("^[a-zA-Z\u0020]+$", LegacyString.**class**),

***ALPHABETIC\_LOWER***("^[a-z,\u0020]+$", LegacyString.**class**),

***ALPHABETIC\_UPPER***("^[A-Z\u0020]+$", LegacyString.**class**);

## Pom.xml

Vì không hiểu được cái version cho mỗi dependency nó nằm ở đâu, nên sửa lại file pom như sau:

<project xmlns=*"http://maven.apache.org/POM/4.0.0"* xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xsi:schemaLocation=*"http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"*>

<modelVersion>4.0.0</modelVersion>

<artifactId>qre2-core</artifactId>

<name>${project.artifactId}</name>

<groupId>com.csc.qre</groupId>

<version>1.3.0-SNAPSHOT</version>

<dependencies>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>5.1.38</version>

</dependency>

<!-- tạo ra 1 cái pool để kết nối database cho database dạng DBMS -->

<dependency>

<groupId>commons-dbcp</groupId>

<artifactId>commons-dbcp</artifactId>

<version>1.4</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-orm</artifactId>

<version>4.3.4.RELEASE</version>

</dependency>

<dependency>

<groupId>javax.inject</groupId>

<artifactId>javax.inject</artifactId>

<version>1</version>

</dependency>

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-core</artifactId>

<version>5.2.5.Final</version>

</dependency>

<dependency>

<groupId>org.aspectj</groupId>

<artifactId>aspectjrt</artifactId>

<version>1.8.9</version>

</dependency>

<dependency>

<groupId>org.aspectj</groupId>

<artifactId>aspectjweaver</artifactId>

<version>1.8.9</version>

</dependency>

<dependency>

<groupId>org.hibernate.java-persistence</groupId>

<artifactId>jpa-api</artifactId>

<version>2.0-cr-1</version>

</dependency>

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-entitymanager</artifactId>

<version>5.2.2.Final</version>

</dependency>

<dependency>

<groupId>org.springframework.security</groupId>

<artifactId>spring-security-core</artifactId>

<version>4.2.0.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-expression</artifactId>

<version>4.3.4.RELEASE</version>

</dependency>

<!-- easymock dùng để làm unit test -->

<dependency>

<groupId>org.easymock</groupId>

<artifactId>easymock</artifactId>

<version>3.4</version>

</dependency>

<!-- cglib là một phần của Spring AOP

Kinh nghiệm lập trình chỉ ra rằng, để

hoàn thành một chương trình, bạn không chỉ phải giải quyết các

business logic thỏa mãn được yêu cầu khách hàng mà con cần giải quyết

những logic bắt buộc khác. Những logic này tuy không thể hiện các

business rules, business processes của khách hàng nhưng lại luôn xuất

hiện trong mọi chương trình. Vài ví dụ như: login/logout, validation,

authentication và authorization, logging, trail log... Những thành

phần vừa nêu giữ cho chương trình của bạn bền vững. Rắc rối ở chỗ nó

xuất hiện ở nhiều nơi trong chương trình nên việc lập trình nó sẽ dẫn

đến duplicate code nhiều nơi trong chương trình. Đây là một thiết kế

xấu. Những code của các thành phần như vậy gọi là concern hoặc cross

cutting code. Lập trình viên chúng ta luôn muốn viết code một lần mà

chạy được nhiều nơi, độc lập để dễ tái sử dụng. Spring AOP sẽ giải

quyết cho chúng ta các cross cutting code này. -->

<dependency>

<groupId>cglib</groupId>

<artifactId>cglib</artifactId>

<version>3.2.4</version>

</dependency>

<dependency>

<groupId>cglib</groupId>

<artifactId>cglib-nodep</artifactId>

<version>3.2.4</version>

</dependency>

<!-- <dependency> <groupId>com.csc.qre</groupId> <artifactId>qre2-test</artifactId>

</dependency> -->

</dependencies>

<profiles>

<profile>

<id>release</id>

<build>

<plugins>

<plugin>

<groupId>com.github.wvengen</groupId>

<artifactId>proguard-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</profile>

</profiles>

</project>

Khi import các giá trị sau:

**import** **static** org.junit.Assert.\*;

**import** org.junit.Test;

Vì chưa có package junit nên sẽ bị error. Vì thế thêm dependency sau vào file pom.xml:

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-test</artifactId>

<version>4.3.4.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<version>1.4.2.RELEASE</version>

</dependency>7.2 Project qre-testharness-cobol

New server:



## QAStyle.css

*.turquoise* { /\*màu này là màu chữ của 6 cái menu, turqoise nghĩa là ngọc lam\*/

color: *#2216F6*;

}

Các file sử dụng cái turqoise này:

1. C:\Workspace\QRE-Demo\qre2-testharness-cobol\src\main\java\com\csc\personnel\online\Ordrent.java:

2. C:\Workspace\QRE-Demo\qre2-testharness-cobol\src\main\webapp\Menset1Menmap1Form.jsp:

3. C:\Workspace\QRE-Demo\qre2-testharness-cobol\src\main\webapp\Ordset1Map1Form.jsp

4. C:\Workspace\QRE-Demo\qre2-testharness-cobol\src\main\webapp\Ordset1Ordmap1Form.jsp:

5. C:\Workspace\QRE-Demo\qre2-testharness-cobol\src\main\webapp\Test.jsp:

Cái 2,3,4 sử dụng khá nhiều, tuy nhiên ở đây dùng nhiều chữ in hoa, hình như TURQUOISE là màu ngọc lam có sẵn của HMTL.



Cũ:



# 8. QRE chính thức:

Sử dụng file daito-all-iterations-project-set.psf để import vào C:\Workspace\QRE

Right click qre2=parent, clean build kết quả được cái này:



Lấy đủ project: [\\20.203.146.68\Software\Documents\PSF](file:///\\20.203.146.68\Software\Documents\PSF)

daito-as01-iterations.psf

honda-poc.psf

# 9. Builder Pattern

<http://www.javaworld.com/article/2074938/core-java/too-many-parameters-in-java-methods-part-3-builder-pattern.html>

package: dustin.examples

## Cách 1:

**package** dustin.examples;

**public** **class** PersonBuilder {

**private** String newLastName;

**private** String newFirstName;

**private** String newMiddleName;

**private** String newSalutation;

**private** String newSuffix;

**private** String newStreetAddress;

**private** String newCity;

**private** String newState;

**private** **boolean** newIsFemale;

**private** **boolean** newIsEmployed;

**private** **boolean** newIsHomeOwner;

**public** PersonBuilder() {

}

// thay vì sử dụng hàm setter như bình thường, thì ở đây void thay bằng PersonBuilder và return this.

**public** PersonBuilder setNewLastName(String newLastName) {

**this**.newLastName = newLastName;

**return** **this**;

}

**public** PersonBuilder setNewFirstName(String newFirstName) {

**this**.newFirstName = newFirstName;

**return** **this**;

}

**public** PersonBuilder setNewMiddleName(String newMiddleName) {

**this**.newMiddleName = newMiddleName;

**return** **this**;

}

**public** PersonBuilder setNewSalutation(String newSalutation) {

**this**.newSalutation = newSalutation;

**return** **this**;

}

**public** PersonBuilder setNewSuffix(String newSuffix) {

**this**.newSuffix = newSuffix;

**return** **this**;

}

**public** PersonBuilder setNewStreetAddress(String newStreetAddress) {

**this**.newStreetAddress = newStreetAddress;

**return** **this**;

}

**public** PersonBuilder setNewCity(String newCity) {

**this**.newCity = newCity;

**return** **this**;

}

**public** PersonBuilder setNewState(String newState) {

**this**.newState = newState;

**return** **this**;

}

**public** PersonBuilder setNewIsFemale(**boolean** newIsFemale) {

**this**.newIsFemale = newIsFemale;

**return** **this**;

}

**public** PersonBuilder setNewIsEmployed(**boolean** newIsEmployed) {

**this**.newIsEmployed = newIsEmployed;

**return** **this**;

}

**public** PersonBuilder setNewIsHomeOwner(**boolean** newIsHomeOwner) {

**this**.newIsHomeOwner = newIsHomeOwner;

**return** **this**;

}

@Override

**public** String toString() {

**return** "LastName: "+newLastName+"FirstName: "+newFirstName;

}

}

Test:

**package** dustin.examples;

**public** **class** PersonBuilderTest {

**public** **static** **void** main(String[] args) {

PersonBuilder p = **new** PersonBuilder().setNewLastName("Nguyen").setNewFirstName("Thuat");

System.***out***.println(p);

}

}

Result:



## Cách 2: sử dụng Nested Class

<http://ktmt.github.io/blog/2013/06/14/design-pattern-ap-dung-builder-pattern-trong-test-java/>

Book.java: với Bulder là nested class nằm trong Book

**package** dustin.examples;

**import** java.util.Calendar;

**import** java.util.GregorianCalendar;

**public** **class** Book {

**public** **enum** Genre {

***FICTION***, ***NONFICTION***, ***TECHNOLOGY***, ***SELFHELP***, ***BUSINESS***, ***SPORT***

};

**private** String title;

**private** String author;

**private** Genre genre;

**private** GregorianCalendar publishDate;

**private** String ISBN;

**public** **static** **class** Builder {

// required params

**private** String title;

**private** String author;

// optional params

**private** Genre genre = Genre.***FICTION***;

**private** GregorianCalendar publishDate = **new** GregorianCalendar(1900, 1, 1);

**private** String ISBN = "000000000";

**public** Builder(String title, String author) {

**this**.title = title;

**this**.author = author;

}

**public** Builder setGenre(Genre val) {

**this**.genre = val;

**return** **this**;

}

**public** Builder setPublishDate(GregorianCalendar val) {

**this**.publishDate = val;

**return** **this**;

}

**public** Builder setISBN(String val) {

**this**.ISBN = val;

**return** **this**;

}

**public** Book build() {

**return** **new** Book(**this**);

}

}

**public** Book(Builder builder) {

title = builder.title;

author = builder.author;

genre = builder.genre;

publishDate = builder.publishDate;

ISBN = builder.ISBN;

}

@Override

**public** String toString() {

**return** "Title: " + title + ", author: " + author + ", genre: " + genre.toString() + ", publish year: "

+ publishDate.get(Calendar.***YEAR***) + ", ISBN: " + ISBN;

}

}

Booktest.java:

**package** dustin.examples;

**import** java.util.GregorianCalendar;

**public** **class** BookTest {

**public** **static** **void** main(String[] args) {

Book book = **new** Book.Builder("Effective Java", "Joshua Bloch")

.setPublishDate(**new** GregorianCalendar(2008, 05, 28)).build();

System.***out***.println(book);

}

}

Result:



## Cách 2: Nested Class kiểu khác – cách này là chuẩn nhất

Person.java:

**package** dustin.examples;

**public** **class** Person

{

**private** **final** String lastName;

**private** **final** String firstName;

**private** **final** String middleName;

**private** **final** String salutation;

**private** **final** String suffix;

**private** **final** String streetAddress;

**private** **final** String city;

**private** **final** String state;

**private** **final** **boolean** isFemale;

**private** **final** **boolean** isEmployed;

**private** **final** **boolean** isHomewOwner;

**private** Person( // nếu để public ở đây thì có thể new Person(...), vì thể t nên để lại là private

**final** String newLastName,

**final** String newFirstName,

**final** String newMiddleName,

**final** String newSalutation,

**final** String newSuffix,

**final** String newStreetAddress,

**final** String newCity,

**final** String newState,

**final** **boolean** newIsFemale,

**final** **boolean** newIsEmployed,

**final** **boolean** newIsHomeOwner)

{

**this**.lastName = newLastName;

**this**.firstName = newFirstName;

**this**.middleName = newMiddleName;

**this**.salutation = newSalutation;

**this**.suffix = newSuffix;

**this**.streetAddress = newStreetAddress;

**this**.city = newCity;

**this**.state = newState;

**this**.isFemale = newIsFemale;

**this**.isEmployed = newIsEmployed;

**this**.isHomewOwner = newIsHomeOwner;

}

**public** **static** **class** PersonBuilder

{

**private** String nestedLastName;

**private** String nestedFirstName;

**private** String nestedMiddleName;

**private** String nestedSalutation;

**private** String nestedSuffix;

**private** String nestedStreetAddress;

**private** String nestedCity;

**private** String nestedState;

**private** **boolean** nestedIsFemale;

**private** **boolean** nestedIsEmployed;

**private** **boolean** nestedIsHomeOwner;

// required param

**public** PersonBuilder(

**final** String newFirstName,

**final** String newCity,

**final** String newState)

{

**this**.nestedFirstName = newFirstName;

**this**.nestedCity = newCity;

**this**.nestedState = newState;

}

**public** PersonBuilder lastName(String newLastName)

{

**this**.nestedLastName = newLastName;

**return** **this**;

}

**public** PersonBuilder firstName(String newFirstName)

{

**this**.nestedFirstName = newFirstName;

**return** **this**;

}

**public** PersonBuilder middleName(String newMiddleName)

{

**this**.nestedMiddleName = newMiddleName;

**return** **this**;

}

**public** PersonBuilder salutation(String newSalutation)

{

**this**.nestedSalutation = newSalutation;

**return** **this**;

}

**public** PersonBuilder suffix(String newSuffix)

{

**this**.nestedSuffix = newSuffix;

**return** **this**;

}

**public** PersonBuilder streetAddress(String newStreetAddress)

{

**this**.nestedStreetAddress = newStreetAddress;

**return** **this**;

}

**public** PersonBuilder city(String newCity)

{

**this**.nestedCity = newCity;

**return** **this**;

}

**public** PersonBuilder state(String newState)

{

**this**.nestedState = newState;

**return** **this**;

}

**public** PersonBuilder isFemale(**boolean** newIsFemale)

{

**this**.nestedIsFemale = newIsFemale;

**return** **this**;

}

**public** PersonBuilder isEmployed(**boolean** newIsEmployed)

{

**this**.nestedIsEmployed = newIsEmployed;

**return** **this**;

}

**public** PersonBuilder isHomeOwner(**boolean** newIsHomeOwner)

{

**this**.nestedIsHomeOwner = newIsHomeOwner;

**return** **this**;

}

**public** Person createPerson()

{

**return** **new** Person(

nestedLastName, nestedFirstName, nestedMiddleName,

nestedSalutation, nestedSuffix,

nestedStreetAddress, nestedCity, nestedState,

nestedIsFemale, nestedIsEmployed, nestedIsHomeOwner);

}

}

@Override

**public** String toString() {

**return** "lastName: "+lastName + "\n" +

"firstName: "+firstName + "\n" +

"middleName: "+middleName + "\n" +

"salutation: "+salutation + "\n" +

"suffix: "+suffix + "\n" +

"streetAddress: "+streetAddress + "\n" +

"city: "+city + "\n" +

"state: "+state + "\n" +

"isFemale: "+isFemale + "\n" +

"isEmployed: "+isEmployed + "\n" +

"isHomewOwner: "+isHomewOwner+ "\n"

;

}

}

PersonTest.java:

**package** dustin.examples;

**public** **class** PersonTest {

**public** **static** **void** main(String[] args) {

// khởi tạo cho 3 param required thôi. Các param còn lại sẽ là null.

Person p = **new** Person.PersonBuilder("Thuat", "Ben Tre", "Ben Tre State").createPerson();

System.***out***.println(p);

System.***out***.println("---------------------");

// khởi tạo thêm có các param optional

Person p2 = **new** Person.PersonBuilder("Hien", "Chau Duc", "Dong Nai State").lastName("Tran").createPerson();

System.***out***.println(p2);

System.***out***.println("---------------------");

}

}

Result:



# 10. Oracle:

Import file dump bằng giao diện:

<https://oracle-base.com/articles/misc/sql-developer-31-data-pump-wizards>

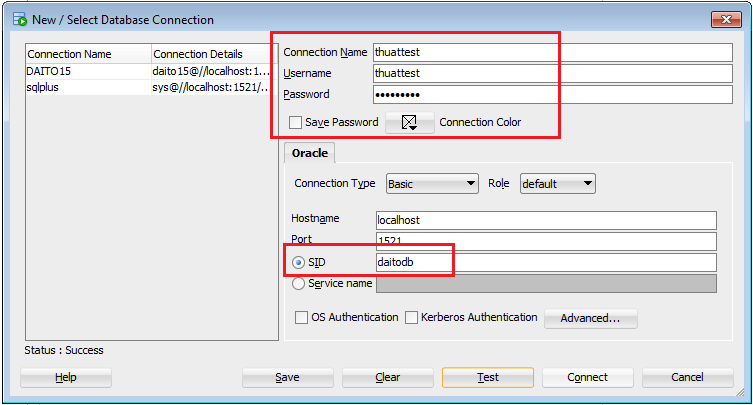
Tạo user mới:

CREATE USER thuattest identified by thuattest default tablespace I15\_DATA temporary tablespace I15\_TEMP;

GRANT UNLIMITED TABLESPACE TO "THUATTEST" WITH ADMIN OPTION;

GRANT "CONNECT" TO "THUATTEST" WITH ADMIN OPTION;

GRANT "DBA" TO "THUATTEST" WITH ADMIN OPTION;



## Cài bằng desktop

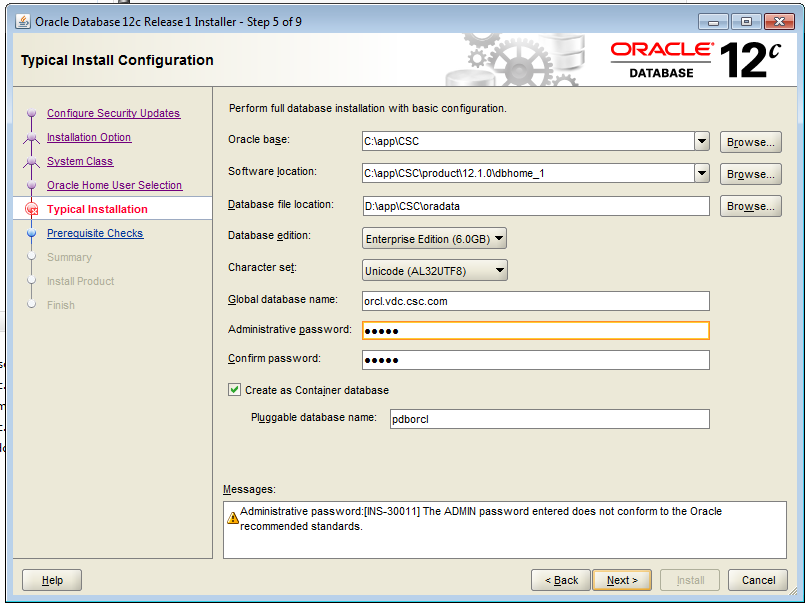
Đậu móa nhớ là extract cả 2 file, dung lượng là

2.71 GB (2,912,047,104 bytes)

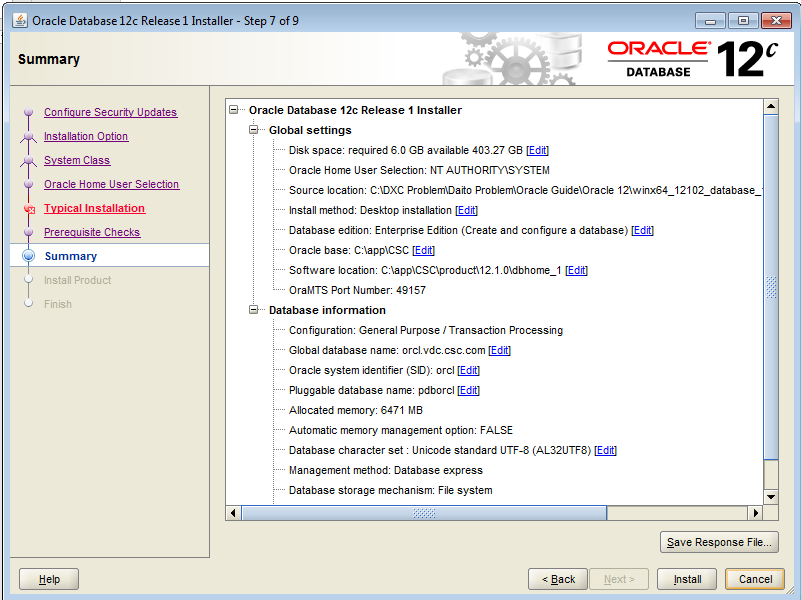
Tran, Man Minh 10:55 AM:

2.70 GB (2,908,728,482 bytes)

Need Skype for Business? [Install](https://aka.ms/getsfbv2) it on all your devices.



Password: daito



# 11. Thuế



Mã số thuế: [8413533195](javascript:submitform('8413533195'))

Trang tra cứu:

<http://tracuunnt.gdt.gov.vn/tcnnt/mstcn.jsp>

# 12. Enum trong java:

<https://www.mkyong.com/java/java-enum-example/>

# 13. Lambda expressions

**Lambda expressions là** một tính năng mới quan trọng trong Java 8. **Lambda expressions** giống class vô danh biểu diễn dưới dạng biểu thức. Chỉ bằng một biểu thức nó có thể biểu diễn thực thi cho method của functional interfaces. Functional interfaces **là** interface chỉ có 1 method.

Ví dụ:

<http://www.oracle.com/webfolder/technetwork/tutorials/obe/java/Lambda-QuickStart/index.html>

Toàn bộ ví dụ này nằm trong package com.lambda

## Runnable Lambda

**package** com.lambda;

**public** **class** RunnableTest {

**public** **static** **void** main(String[] args) {

System.***out***.println("=== RunnableTest ===");

// Anonymous Runnable

Runnable r1 = **new** Runnable() {

@Override

**public** **void** run() {

System.***out***.println("Hello world one!");

}

};

// Lambda Runnable

Runnable r2 = () -> System.***out***.println("Hello world two!");

// Run em!

r1.run();

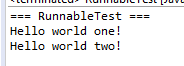
r2.run();

}

}

Cách dùng cho anonymous và lambda ở đây là hoàn toàn giống nhau. Chỉ có điều viết kiểu lambda nó cool hơn, elegant hơn :v

Result:



## Comparator Lambda

Gender.java

**package** com.lambda;

/\*\*

\* **@author** MikeW

\*/

**public** **enum** Gender { ***MALE***, ***FEMALE*** }

Person.java

**package** com.lambda;

**import** java.util.ArrayList;

**import** java.util.List;

**public** **class** Person {

**private** String givenName;

**private** String surName;

**private** **int** age;

**private** Gender gender;

**private** String eMail;

**private** String phone;

**private** String address;

**public** **static** **class** Builder{

**private** String givenName="";

**private** String surName="";

**private** **int** age = 0;

**private** Gender gender = Gender.***FEMALE***;

**private** String eMail = "";

**private** String phone = "";

**private** String address = "";

**public** Person.Builder givenName(String givenName){

**this**.givenName = givenName;

**return** **this**;

}

**public** Person.Builder surName(String surName){

**this**.surName = surName;

**return** **this**;

}

**public** Person.Builder age (**int** val){

age = val;

**return** **this**;

}

**public** Person.Builder gender(Gender val){

gender = val;

**return** **this**;

}

**public** Person.Builder email(String val){

eMail = val;

**return** **this**;

}

**public** Person.Builder phoneNumber(String val){

phone = val;

**return** **this**;

}

**public** Person.Builder address(String val){

address = val;

**return** **this**;

}

**public** Person build(){

**return** **new** Person(**this**);

}

}

**private** Person(){

**super**();

}

**private** Person(Person.Builder builder){

givenName = builder.givenName;

surName = builder.surName;

age = builder.age;

gender = builder.gender;

eMail = builder.eMail;

phone = builder.phone;

address = builder.address;

}

**public** String getGivenName(){

**return** givenName;

}

**public** String getSurName(){

**return** surName;

}

**public** **int** getAge(){

**return** age;

}

**public** **void** print(){

System.***out***.println(

"\nName: " + givenName + " " + surName + "\n" +

"Age: " + age + "\n" +

"Gender: " + gender + "\n" +

"eMail: " + eMail + "\n" +

"Phone: " + phone + "\n" +

"Address: " + address + "\n"

);

}

**public** **void** printName(){

System.***out***.println(

"Name: " + givenName + " " + surName);

}

@Override

**public** String toString(){

**return** "Name: " + givenName + " " + surName + "\n" + "Age: " + age + " Gender: " + gender + "\n" + "eMail: " + eMail + "\n" + "Address: " + address + "\n";

}

**public** **static** List<Person> createShortList(){

List<Person> people = **new** ArrayList<>();

people.add(

**new** Person.Builder()

.givenName("Bob")

.surName("Baker")

.age(21)

.gender(Gender.***MALE***)

.email("bob.baker@example.com")

.phoneNumber("201-121-4678")

.address("44 4th St, Smallville, KS 12333")

.build()

);

people.add(

**new** Person.Builder()

.givenName("Jane")

.surName("Doe")

.age(25)

.gender(Gender.***FEMALE***)

.email("jane.doe@example.com")

.phoneNumber("202-123-4678")

.address("33 3rd St, Smallville, KS 12333")

.build()

);

people.add(

**new** Person.Builder()

.givenName("John")

.surName("Doe")

.age(25)

.gender(Gender.***MALE***)

.email("john.doe@example.com")

.phoneNumber("202-123-4678")

.address("33 3rd St, Smallville, KS 12333")

.build()

);

people.add(

**new** Person.Builder()

.givenName("James")

.surName("Johnson")

.age(45)

.gender(Gender.***MALE***)

.email("james.johnson@example.com")

.phoneNumber("333-456-1233")

.address("201 2nd St, New York, NY 12111")

.build()

);

people.add(

**new** Person.Builder()

.givenName("Joe")

.surName("Bailey")

.age(67)

.gender(Gender.***MALE***)

.email("joebob.bailey@example.com")

.phoneNumber("112-111-1111")

.address("111 1st St, Town, CA 11111")

.build()

);

people.add(

**new** Person.Builder()

.givenName("Phil")

.surName("Smith")

.age(55)

.gender(Gender.***MALE***)

.email("phil.smith@examp;e.com")

.phoneNumber("222-33-1234")

.address("22 2nd St, New Park, CO 222333")

.build()

);

people.add(

**new** Person.Builder()

.givenName("Betty")

.surName("Jones")

.age(85)

.gender(Gender.***FEMALE***)

.email("betty.jones@example.com")

.phoneNumber("211-33-1234")

.address("22 4th St, New Park, CO 222333")

.build()

);

**return** people;

}

}

ComparatorTest.java

**package** com.lambda;

**import** java.util.Collections;

**import** java.util.Comparator;

**import** java.util.List;

**public** **class** ComparatorTest {

**public** **static** **void** main(String[] args) {

List<Person> personList = Person.*createShortList*();

// Get SurName using lambda

personList.forEach(p -> {

System.***out***.println(p.getSurName());

});

// Sort with Inner Class

Collections.*sort*(personList, **new** Comparator<Person>() {

**public** **int** compare(Person p1, Person p2) {

**return** p1.getSurName().compareTo(p2.getSurName());

}

});

System.***out***.println("=== Sorted Asc SurName ===");

**for** (Person p : personList) {

p.printName();

}

// Use Lambda instead

// Print Asc

System.***out***.println("=== Sorted Asc SurName ===");

Collections.*sort*(personList, (Person p1, Person p2) -> p1.getSurName().compareTo(p2.getSurName()));

personList.forEach(p -> {

p.printName();

;

});

// Print Desc

System.***out***.println("=== Sorted Desc SurName ===");

Collections.*sort*(personList, (p1, p2) -> p2.getSurName().compareTo(p1.getSurName()));

personList.forEach(p -> {

p.printName();

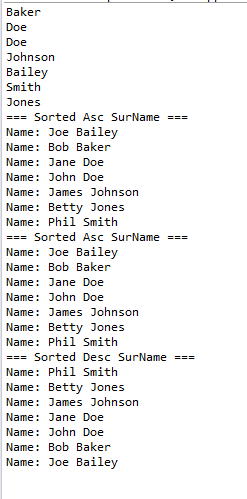
;

});

}

}

Result:



## ListenerTest

**package** com.lambda;

**import** java.awt.BorderLayout;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** javax.swing.JButton;

**import** javax.swing.JFrame;

**public** **class** ListenerTest {

**public** **static** **void** main(String[] args) {

JButton testButton = **new** JButton("Test Button");

testButton.addActionListener(**new** ActionListener(){

@Override **public** **void** actionPerformed(ActionEvent ae){

System.***out***.println("Click Detected by Anon Class");

}

});

testButton.addActionListener(e -> System.***out***.println("Click Detected by Lambda Listner"));

// Swing stuff

JFrame frame = **new** JFrame("Listener Test");

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.add(testButton, BorderLayout.***CENTER***);

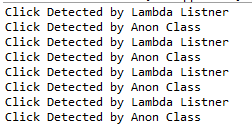
frame.pack();

frame.setVisible(**true**);

}

}

Result:



## Improving code with lambda expression

Cái này có cấu trúc khá hay, ráng học hỏi và làm theo.

Person.java

**package** com.lambda;

**import** java.util.ArrayList;

**import** java.util.List;

**public** **class** Person {

**private** String givenName;

**private** String surName;

**private** **int** age;

**private** Gender gender;

**private** String eMail;

**private** String phone;

**private** String address;

**public** **static** **class** Builder{

**private** String givenName="";

**private** String surName="";

**private** **int** age = 0;

**private** Gender gender = Gender.***FEMALE***;

**private** String eMail = "";

**private** String phone = "";

**private** String address = "";

**public** Person.Builder givenName(String givenName){

**this**.givenName = givenName;

**return** **this**;

}

**public** Person.Builder surName(String surName){

**this**.surName = surName;

**return** **this**;

}

**public** Person.Builder age (**int** val){

age = val;

**return** **this**;

}

**public** Person.Builder gender(Gender val){

gender = val;

**return** **this**;

}

**public** Person.Builder email(String val){

eMail = val;

**return** **this**;

}

**public** Person.Builder phoneNumber(String val){

phone = val;

**return** **this**;

}

**public** Person.Builder address(String val){

address = val;

**return** **this**;

}

**public** Person build(){

**return** **new** Person(**this**);

}

}

**private** Person(){

**super**();

}

**private** Person(Person.Builder builder){

givenName = builder.givenName;

surName = builder.surName;

age = builder.age;

gender = builder.gender;

eMail = builder.eMail;

phone = builder.phone;

address = builder.address;

}

**public** String getGivenName() {

**return** givenName;

}

**public** String getSurName() {

**return** surName;

}

**public** **int** getAge() {

**return** age;

}

**public** Gender getGender() {

**return** gender;

}

**public** String getEmail() {

**return** eMail;

}

**public** String getPhone() {

**return** phone;

}

**public** String getAddress() {

**return** address;

}

**public** **void** print(){

System.***out***.println(

"\nName: " + givenName + " " + surName + "\n" +

"Age: " + age + "\n" +

"Gender: " + gender + "\n" +

"eMail: " + eMail + "\n" +

"Phone: " + phone + "\n" +

"Address: " + address + "\n"

);

}

**public** **void** printName(){

System.***out***.println(

"Name: " + givenName + " " + surName);

}

@Override

**public** String toString(){

**return** "Name: " + givenName + " " + surName + "\n" + "Age: " + age + " Gender: " + gender + "\n" + "eMail: " + eMail + "\n" + "Address: " + address + "\n";

}

**public** **static** List<Person> createShortList(){

List<Person> people = **new** ArrayList<>();

people.add(

**new** Person.Builder()

.givenName("Bob")

.surName("Baker")

.age(21)

.gender(Gender.***MALE***)

.email("bob.baker@example.com")

.phoneNumber("201-121-4678")

.address("44 4th St, Smallville, KS 12333")

.build()

);

people.add(

**new** Person.Builder()

.givenName("Jane")

.surName("Doe")

.age(25)

.gender(Gender.***FEMALE***)

.email("jane.doe@example.com")

.phoneNumber("202-123-4678")

.address("33 3rd St, Smallville, KS 12333")

.build()

);

people.add(

**new** Person.Builder()

.givenName("John")

.surName("Doe")

.age(25)

.gender(Gender.***MALE***)

.email("john.doe@example.com")

.phoneNumber("202-123-4678")

.address("33 3rd St, Smallville, KS 12333")

.build()

);

people.add(

**new** Person.Builder()

.givenName("James")

.surName("Johnson")

.age(45)

.gender(Gender.***MALE***)

.email("james.johnson@example.com")

.phoneNumber("333-456-1233")

.address("201 2nd St, New York, NY 12111")

.build()

);

people.add(

**new** Person.Builder()

.givenName("Joe")

.surName("Bailey")

.age(67)

.gender(Gender.***MALE***)

.email("joebob.bailey@example.com")

.phoneNumber("112-111-1111")

.address("111 1st St, Town, CA 11111")

.build()

);

people.add(

**new** Person.Builder()

.givenName("Phil")

.surName("Smith")

.age(55)

.gender(Gender.***MALE***)

.email("phil.smith@examp;e.com")

.phoneNumber("222-33-1234")

.address("22 2nd St, New Park, CO 222333")

.build()

);

people.add(

**new** Person.Builder()

.givenName("Betty")

.surName("Jones")

.age(85)

.gender(Gender.***FEMALE***)

.email("betty.jones@example.com")

.phoneNumber("211-33-1234")

.address("22 4th St, New Park, CO 222333")

.build()

);

**return** people;

}

}

Predicate.java

**package** com.lambda;

**public** **interface** Predicate<T> {

**public** **boolean** test(T t);

}

RoboContactLambda.java

**package** com.lambda;

**import** java.util.List;

**public** **class** RoboContactLambda {

**public** **void** phoneContacts(List<Person> pl, Predicate<Person> pred) {

pl.forEach(p -> {

**if** (pred.test(p)) {

roboEmail(p);

}

});

}

**public** **void** emailContacts(List<Person> pl, Predicate<Person> pred) {

pl.forEach(p -> {

**if** (pred.test(p)) {

roboCall(p);

}

});

}

**public** **void** mailContacts(List<Person> pl, Predicate<Person> pred) {

pl.forEach(p -> {

**if** (pred.test(p)) {

roboMail(p);

}

});

}

**public** **void** roboCall(Person p) {

System.***out***.println("Calling " + p.getGivenName() + " " + p.getSurName() + " age " + p.getAge() + " at " + p.getPhone());

}

**public** **void** roboEmail(Person p) {

System.***out***.println("EMailing " + p.getGivenName() + " " + p.getSurName() + " age " + p.getAge() + " at " + p.getEmail());

}

**public** **void** roboMail(Person p) {

System.***out***.println("Mailing " + p.getGivenName() + " " + p.getSurName() + " age " + p.getAge() + " at " + p.getAddress());

}

}

RoboCallTest04.java

**package** com.lambda;

**import** java.util.List;

/\*\*

\*

\* **@author** MikeW

\*/

**public** **class** RoboCallTest04 {

**public** **static** **void** main(String[] args){

List<Person> pl = Person.*createShortList*();

RoboContactLambda robo = **new** RoboContactLambda();

// Predicates

Predicate<Person> allDrivers = p -> p.getAge() >= 16;

Predicate<Person> allDraftees = p -> p.getAge() >= 18 && p.getAge() <= 25 && p.getGender() == Gender.***MALE***;

Predicate<Person> allPilots = p -> p.getAge() >= 23 && p.getAge() <= 65;

System.***out***.println("\n==== Test 04 ====");

System.***out***.println("\n=== Calling all Drivers ===");

robo.phoneContacts(pl, allDrivers);

System.***out***.println("\n=== Emailing all Draftees ===");

robo.emailContacts(pl, allDraftees);

System.***out***.println("\n=== Mail all Pilots ===");

robo.mailContacts(pl, allPilots);

// Mix and match becomes easy

System.***out***.println("\n=== Mail all Draftees ===");

robo.mailContacts(pl, allDraftees);

System.***out***.println("\n=== Call all Pilots ===");

robo.phoneContacts(pl, allPilots);

}

}

Result:



## The java.ulti.function Package

Person2.java

**package** com.lambda;

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.function.Function;

/\*\*

\* **@author** MikeW

\*/

**public** **class** Person2 {

**private** String givenName;

**private** String surName;

**private** **int** age;

**private** Gender gender;

**private** String eMail;

**private** String phone;

**private** String address;

**public** **static** **class** Builder{

**private** String givenName="";

**private** String surName="";

**private** **int** age = 0;

**private** Gender gender = Gender.***FEMALE***;

**private** String eMail = "";

**private** String phone = "";

**private** String address = "";

**public** Person2.Builder givenName(String givenName){

**this**.givenName = givenName;

**return** **this**;

}

**public** Person2.Builder surName(String surName){

**this**.surName = surName;

**return** **this**;

}

**public** Person2.Builder age (**int** val){

age = val;

**return** **this**;

}

**public** Person2.Builder gender(Gender val){

gender = val;

**return** **this**;

}

**public** Person2.Builder email(String val){

eMail = val;

**return** **this**;

}

**public** Person2.Builder phoneNumber(String val){

phone = val;

**return** **this**;

}

**public** Person2.Builder address(String val){

address = val;

**return** **this**;

}

**public** Person2 build(){

**return** **new** Person2(**this**);

}

}

**private** Person2(){

**super**();

}

**private** Person2(Person2.Builder builder){

givenName = builder.givenName;

surName = builder.surName;

age = builder.age;

gender = builder.gender;

eMail = builder.eMail;

phone = builder.phone;

address = builder.address;

}

**public** String getGivenName(){

**return** givenName;

}

**public** String getSurName(){

**return** surName;

}

**public** **int** getAge(){

**return** age;

}

**public** Gender getGender(){

**return** gender;

}

**public** String getEmail(){

**return** eMail;

}

**public** String getPhone(){

**return** phone;

}

**public** String getAddress(){

**return** address;

}

**public** **void** print(){

System.***out***.println(

"\nName: " + givenName + " " + surName + "\n" +

"Age: " + age + "\n" +

"Gender: " + gender + "\n" +

"eMail: " + eMail + "\n" +

"Phone: " + phone + "\n" +

"Address: " + address + "\n"

);

}

**public** String printCustom(Function <Person2, String> f){

**return** f.apply(**this**);

}

// Ở đây k cần define vì cái này sẽ được define luôn trong hàm main

/\*public void printWesternName(){

System.out.println("\nName: " + this.getGivenName() + " " + this.getSurName() + "\n" +

"Age: " + this.getAge() + " " + "Gender: " + this.getGender() + "\n" +

"EMail: " + this.getEmail() + "\n" +

"Phone: " + this.getPhone() + "\n" +

"Address: " + this.getAddress());

}\*/

// Ở đây k cần define vì cái này sẽ được define luôn trong hàm main

/\*public void printEasternName(){

System.out.println("\nName: " + this.getSurName() + " " + this.getGivenName() + "\n" +

"Age: " + this.getAge() + " " + "Gender: " + this.getGender() + "\n" +

"EMail: " + this.getEmail() + "\n" +

"Phone: " + this.getPhone() + "\n" +

"Address: " + this.getAddress());

}\*/

@Override

**public** String toString(){

**return** "Name: " + givenName + " " + surName + "\n" + "Age: " + age + " Gender: " + gender + "\n" + "eMail: " + eMail + "\n";

}

**public** **static** List<Person2> createShortList(){

List<Person2> people = **new** ArrayList<>();

people.add(

**new** Person2.Builder()

.givenName("Bob")

.surName("Baker")

.age(21)

.gender(Gender.***MALE***)

.email("bob.baker@example.com")

.phoneNumber("201-121-4678")

.address("44 4th St, Smallville, KS 12333")

.build()

);

people.add(

**new** Person2.Builder()

.givenName("Jane")

.surName("Doe")

.age(25)

.gender(Gender.***FEMALE***)

.email("jane.doe@example.com")

.phoneNumber("202-123-4678")

.address("33 3rd St, Smallville, KS 12333")

.build()

);

people.add(

**new** Person2.Builder()

.givenName("John")

.surName("Doe")

.age(25)

.gender(Gender.***MALE***)

.email("john.doe@example.com")

.phoneNumber("202-123-4678")

.address("33 3rd St, Smallville, KS 12333")

.build()

);

people.add(

**new** Person2.Builder()

.givenName("James")

.surName("Johnson")

.age(45)

.gender(Gender.***MALE***)

.email("james.johnson@example.com")

.phoneNumber("333-456-1233")

.address("201 2nd St, New York, NY 12111")

.build()

);

people.add(

**new** Person2.Builder()

.givenName("Joe")

.surName("Bailey")

.age(67)

.gender(Gender.***MALE***)

.email("joebob.bailey@example.com")

.phoneNumber("112-111-1111")

.address("111 1st St, Town, CA 11111")

.build()

);

people.add(

**new** Person2.Builder()

.givenName("Phil")

.surName("Smith")

.age(55)

.gender(Gender.***MALE***)

.email("phil.smith@examp;e.com")

.phoneNumber("222-33-1234")

.address("22 2nd St, New Park, CO 222333")

.build()

);

people.add(

**new** Person2.Builder()

.givenName("Betty")

.surName("Jones")

.age(85)

.gender(Gender.***FEMALE***)

.email("betty.jones@example.com")

.phoneNumber("211-33-1234")

.address("22 4th St, New Park, CO 222333")

.build()

);

**return** people;

}

}

NameTestNew.java

**package** com.lambda;

**import** java.util.List;

**import** java.util.function.Function;

/\*\*

\* **@author** MikeW

\*/

**public** **class** NameTestNew {

**public** **static** **void** main(String[] args) {

System.***out***.println("\n==== NameTestNew ===");

List<Person2> list1 = Person2.*createShortList*();

// Print Custom First Name and e-mail

System.***out***.println("===Custom List===");

list1.forEach(person2 -> {

System.***out***.println(person2.printCustom(p -> "Name: " + p.getGivenName() + " EMail: " + p.getEmail()));

});

// Define Western and Eastern Lambdas

Function<Person2, String> westernStyle = p -> {

**return** "\nName: " + p.getGivenName() + " " + p.getSurName() + "\n" + "Age: " + p.getAge() + " "

+ "Gender: " + p.getGender() + "\n" + "EMail: " + p.getEmail() + "\n" + "Phone: " + p.getPhone()

+ "\n" + "Address: " + p.getAddress();

};

Function<Person2, String> easternStyle = p -> "\nName: " + p.getSurName() + " " + p.getGivenName() + "\n"

+ "Age: " + p.getAge() + " " + "Gender: " + p.getGender() + "\n" + "EMail: " + p.getEmail() + "\n"

+ "Phone: " + p.getPhone() + "\n" + "Address: " + p.getAddress();

// Print Western List

System.***out***.println("\n===Western List===");

list1.forEach(person2 -> {

System.***out***.println(person2.printCustom(westernStyle));

});

// Print Eastern List

System.***out***.println("\n===Eastern List===");

**for** (Person2 person : list1) {

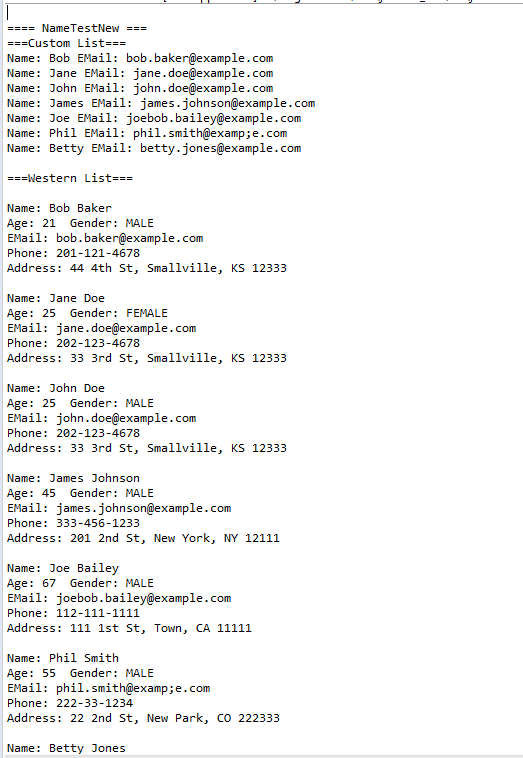
System.***out***.println(person.printCustom(easternStyle));

}

}

}

Result:



….

## Default Method

**package** com.lambda;

**public** **class** DemoDefaultMethod {

**public** **interface** Test {

**public** **void** setup();

**public** **default** **void** run() {

System.***out***.println("Hello Tester");

}

}

**public** **static** **void** main(String[] args) {

Test test = () -> {

System.***out***.println("Setup environment in here");

};

test.run();

}

}

Nghĩa là cái lambda expression nó chỉ apply cho cái nào k phải là default void.

Result:



## Lambda Expressions and Collections

### Filter dùng class searchCriteria theo singleton pattern khá hay.

SearchCriteria.java

**package** com.lambda;

**import** java.util.HashMap;

**import** java.util.Map;

**import** java.util.function.Predicate;

**public** **class** SearchCriteria {

**private** **final** Map<String, Predicate<Person2>> searchMap = **new** HashMap<>();

**private** SearchCriteria() {

**super**();

initSearchMap();

}

**private** **void** initSearchMap() {

Predicate<Person2> allDrivers = p -> p.getAge() >= 16;

Predicate<Person2> allDraftees = p -> p.getAge() >= 18 && p.getAge() <= 25 && p.getGender() == Gender.***MALE***;

Predicate<Person2> allPilots = p -> p.getAge() >= 23 && p.getAge() <= 65;

searchMap.put("allDrivers", allDrivers);

searchMap.put("allDraftees", allDraftees);

searchMap.put("allPilots", allPilots);

}

**public** Predicate<Person2> getCriteria(String PredicateName) {

Predicate<Person2> target;

target = searchMap.get(PredicateName);

**if** (target == **null**) {

System.***out***.println("Search Criteria not found... ");

System.*exit*(1);

}

**return** target;

}

**public** **static** SearchCriteria getInstance() {

**return** **new** SearchCriteria();

}

}

Test02Filter.java

**package** com.lambda;

**import** java.util.List;

/\*\*

\* **@author** MikeW

\*/

**public** **class** Test02Filter {

**public** **static** **void** main(String[] args) {

List<Person2> pl = Person2.*createShortList*();

SearchCriteria search = SearchCriteria.*getInstance*();

System.***out***.println("\n=== Western Pilot Phone List ===");

pl.stream().filter(search.getCriteria("allPilots")).forEach(Person2::printWesternName);

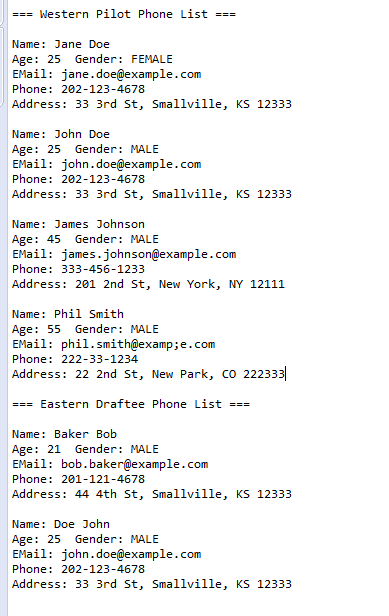
System.***out***.println("\n=== Eastern Draftee Phone List ===");

pl.stream().filter(search.getCriteria("allDraftees")).forEach(Person2::printEasternName);

}

}

Result:



### Tolist: cái này dùng để tạo ra list mới từ 1 cái list có sẵn:

Test03toList.java

**package** com.lambda;

**import** java.util.List;

**import** java.util.stream.Collectors;

**public** **class** Test03toList {

**public** **static** **void** main(String[] args) {

List<Person2> pl = Person2.*createShortList*();

SearchCriteria search = SearchCriteria.*getInstance*();

// Make a new list after filtering.

List<Person2> pilotList = pl.stream().filter(search.getCriteria("allPilots")).collect(Collectors.*toList*());

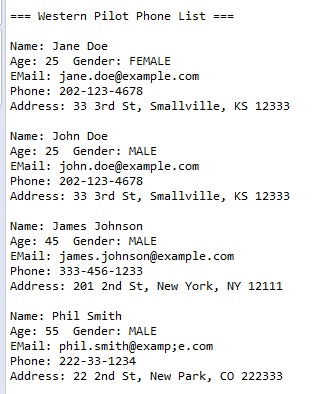
System.***out***.println("\n=== Western Pilot Phone List ===");

pilotList.forEach(Person2::printWesternName);

}

}

Result:



### Tính sum và average theo phong cách mới

Test04Map.java

**package** com.lambda;

**import** java.util.List;

**import** java.util.OptionalDouble;

/\*\*

\* **@author** MikeW

\*/

**public** **class** Test04Map {

**public** **static** **void** main(String[] args) {

List<Person2> pl = Person2.*createShortList*();

SearchCriteria search = SearchCriteria.*getInstance*();

// Calc average age of pilots old style

System.***out***.println("== Calc Old Style ==");

**int** sum = 0;

**int** count = 0;

**for** (Person2 p : pl) {

**if** (p.getAge() >= 23 && p.getAge() <= 65) {

sum = sum + p.getAge();

count++;

}

}

**long** average = sum / count;

System.***out***.println("Total Ages: " + sum);

System.***out***.println("Average Age: " + average);

// Get sum of ages

System.***out***.println("\n== Calc New Style ==");

**long** totalAge = pl.stream().filter(search.getCriteria("allPilots")).mapToInt(p -> p.getAge()).sum();

// Get average of ages

OptionalDouble averageAge = pl.parallelStream().filter(search.getCriteria("allPilots"))

.mapToDouble(p -> p.getAge()).average();

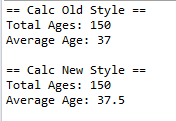
System.***out***.println("Total Ages: " + totalAge);

System.***out***.println("Average Age: " + averageAge.getAsDouble());

}

}

Result:



# 14. QTE-5335 (closed)

## Thông tin task

Link lấy task: <https://vdc-jira.asia.csc.com/jira/browse/QTE>

Title:

Implement reference library in ADDLIBLE POSITION parameter

Description:

Need to handle the reference library name in ADDLIBLE:

cmdExecutor.addLibraryListEntry(this, AddLibraryListEntryParameter.build().setLibrary("DXXOBJ").setPosition(ClpConstants.BEFORE, "HXXOBJ"));

## Tìm hiểu

ADDLIBLE: Add Library List Entry

Link ibm: <http://www.ibm.com/support/knowledgecenter/ssw_i5_54/cl/addlible.htm>

The Add Library List Entry (ADDLIBLE) command adds a library name to the user portion of the library list for the current thread. The user portion is the last portion of the library list. It follows the system portion and, if they exist, any product libraries and the current library entry. You can specify where the library should be placed in the user portion of the library list.

portion: một phần.

entry: lối vào

ADDLIBLE thêm tên thư viện vào cái user portion trong cái list library hiện tại. User portion là phần cuối cùng của list library. Nó tuân theo system portion. Bạn có thể chỉ định nơi mà library được đặt trong user portion của list library.

Hình như user portion chính là user library (nếu k phải thì user library là 1 phần của user portion).

**Library (LIB)**

Specifies the library to be added to the user portion of the library list for the current thread. A maximum of 250 libraries may exist in the user portion of the library list. Only one library name is added at a time with this command.

This is a required parameter.

***name***

Specify the name of the library to be added to the user portion of the library list for the current thread.

**Library list position (POSITION)**

Specifies the position in the user portion of the library list for the current thread where the library is inserted. This parameter can be specified as a single value or as a list of two elements.

**Single values**

**\*FIRST**

The library is inserted in front of the libraries existing in the user portion of the library list for the current thread.

Thư viện này đặt trước thư viện đã tồn tại trong user portion.

**\*LAST**

The library is added to the end of the user portion of the library list for the current thread.

Đặt cuối trong user portion.

**Element 1: List position**

**\*AFTER**

The library specified for the **Library (LIB)** parameter is added to the user portion of the library list for the current thread after the reference library (element 2).

Library sẽ được thêm zô sau cái library thứ 2

**\*BEFORE**

The library specified for the LIB parameter is added to the user portion of the library list for the current thread before the reference library (element 2).

Library được thêm zô sẽ trước cái library thứ 2. Ví dụ:

cmdExecutor.addLibraryListEntry(**this**, AddLibraryListEntryParameter.*build*().setLibrary("DXXOBJ").setPosition(ClpConstants.***BEFORE***, "HXXOBJ"));

Vậy thì cái DXXOBJ sẽ đứng trước cái DXXOBJ

**\*REPLACE**

The library specified for the LIB parameter is inserted into the library list for the current thread in the position currently held by the reference library (element 2) and the reference library is then removed from the list.

**Element 2: Reference library**

***name***

Specify the name of the library to be the reference library when \*AFTER, \*BEFORE, or \*REPLACE is specified for the list position (element 1). This library must exist in the user portion of the library list for the current thread.

Các cpf được define:

**CPF2103**

Library &1 already exists in library list.

**CPF2106**

Library list not available.

**CPF2110**

Library &1 not found.

**CPF2113**

Cannot allocate library &1.

**CPF2118**

Library &1 not added.

**CPF2149**

Library &1 was not found in the user library list.

**CPF2176**

Library &1 damaged.

**CPF2182**

Not authorized to library &1.

Tuy nhiên trong project lại dùng cpf0000, nghĩa là có thể catch tất cả các cpf, trong đó bao gồm cả các cpf ở trên.

## Do it

Các class có dòng command này:

Kvfv04c.java

Kvfv05c.java

Kvfv06c.java

Kvfv07c.java

Kvfw0lc.java

Kvfw0mc.java

Kvfw0pc.java

Kvfw0qc.java

Ldf010c.java

Ldl020c.java

Lee120c.java

Lfl181c.java

Lfl300c.java

Lrf030c.java

Lrl020c.java

Lrl030c.java

Lub050c.java

Lub090c.java

Lub201c.java

Lub202c.java

Lub203c.java

Lub204c.java

Lub210c.java

Lvf240c.java

Lvf260c.java

Lvl820c.java

Tổng cộng 26 file java, và mỗi file chỉ chứa duy nhất 1 dòng command đó thôi.

Và thường kèm theo là 1 dãy các câu này:

@Override

**public** **void** process() {

/\* 処理実行\*/

/\* ----------\*/

/\* ライブラリー追加\*/

**try** {

cmdExecutor.addLibraryListEntry(**this**, AddLibraryListEntryParameter.*build*().setLibrary("DXXOBJ").setPosition(ClpConstants.***BEFORE***, "HXXOBJ"));

}

**catch** (Cpf0000 e){

}

**try** {

cmdExecutor.addLibraryListEntry(**this**, AddLibraryListEntryParameter.*build*().setLibrary("DMMDBF").setPosition(ClpConstants.***BEFORE***, "HMMDBF"));

}

**catch** (Cpf0000 e){

}

**try** {

cmdExecutor.addLibraryListEntry(**this**, AddLibraryListEntryParameter.*build*().setLibrary("DXXWRK").setPosition(ClpConstants.***BEFORE***, "HXXWRK"));

}

**catch** (Cpf0000 e){

}

Có 1 task khá giống task của mình: 5433:

cmdExecutor.reorganizePhysicalFileMbr(this, ReorganizePhysicalFileMbrParameter.build().setFile("LAG1RP").setMember(ClpConstants.FIRST));

Tuy nhiên task 5433 này duplicate của 5187

Một số library cũng sử dụng command setLibrary:

DKTOBJ – không có gì

HKTOBJ – không có gì

DXXOBJ – cái này mình làm mà @@

DKTDBF

HKTDBF

DMMDBF – cái này nằm 1 trong 3 cái chung với DXXOBJ

HIJDBF

Theo mentor nói thì các hàm này đã implement hết rồi. Giờ mình chỉ viết unit test thôi.

Các unit test case có trong AddLibraryListEntryTest.java

testDoExecute: thành công trong 8s

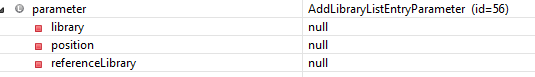
testJavaAndLegacyTypes: thành công trong 8s

testLibraryForNull: thành công trong 8s

testLibraryForException: thành công trong 8s

**final** AddLibraryListEntryParameter parameter = **new** AddLibraryListEntryParameter();

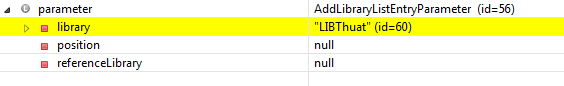
kết quả câu lệnh trên:



Tuy nhiên chạy file này là mình đã có thêm 2 variable sau:



parameter.setLibrary("LIBThuat");



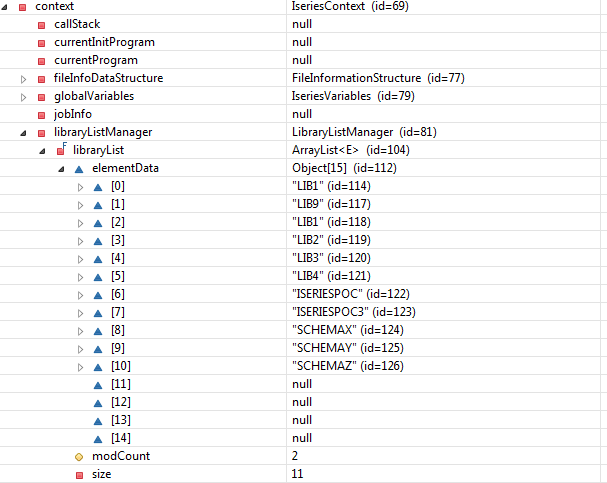
addLibraryList khi được mới tạo ra:



Lệnh lấy chiều dài của cái Library:

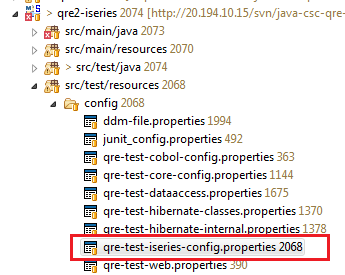
System.***out***.println(context.getLibraryListManager().getLibraryListArray().length);

Variable Context trong file AddLibraryListEntryTest



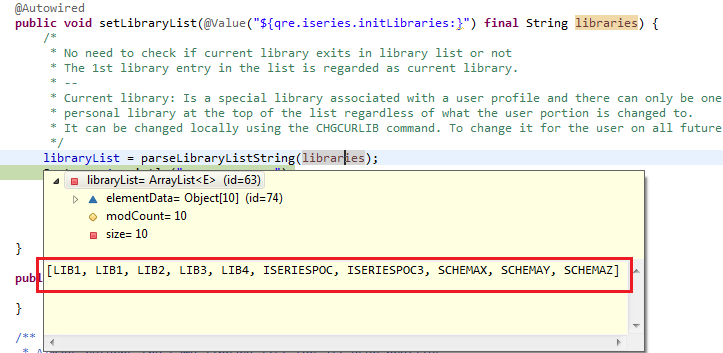
Size nghĩa là length, ở đây tính từ 0 đến 10, tuy nhiên FIRST thì tính từ 1, k tính từ 0, vì tính khi print ra vẫn print từ 1.

qre-test-iseries-config.properties: File này sẽ define các lib sử dụng cho test case.



LibraryListLoaderImpl: file này dùng để init lib. Nào cùng tìm hiểu.

Hàm setLibrary sẽ gán các giá trị lib:



## Cần phải thêm test case:

Error Cpf2186: Object &1 cannot be created into library &2.

## Tìm hiểu thêm

File khá hay định nghĩa các method và các exception: LibraryListManager

AddLibraryListEntryParameter chỉ định nghĩa 1 library.

ChangeLibraryListParameter địnhg nghĩa 1 list library.

QTE-5335: [DAITO][ITER2.2][QRE][CL] Implement reference library in ADDLIBLE POSITION parameter

Sử dụng file kết nối database: qre-test-dataaccess.properties

## Nơi các file log sẽ được in ra:

C:\Workspace\QRE2\qre2-core\target\log

C:\Workspace\QRE2\qre2-iseries\target\log

# 16. QTE-5274 (closed)

Không hiểu sao file này có so sánh rồi:

TestAbstractLegacyDateTime

# 16.1 Câu lệnh display:

org.hsqldb.util.DatabaseManagerSwing.main(**new** String[] {

  "--url",  "jdbc:hsqldb:mem:loadDB;sql.syntax\_ora=true", "--noexit"

});

# 17. Pi Calculation

Lời giải:

<http://stackoverflow.com/questions/41094056/how-to-write-pi-calculation-program-in-java-using-multi-thread>

# 18. QTE-5188: EVAL(R) (closed)

dmax = maximum decimal places of all operands and result involved

Chữ số thập phân tối đa của tất cả các số hạng và kết quả tham gia.

có 1 đoạn code cũng có boolean cũng tham khảo được:

/\*\*

\* Internally used method to handle add operation, with option rounded or not.

\*

\* **@param** round - true indicates the ADD is executed with rounding.

\* **@param** operand1 - operand 1

\* **@param** operand2 - operand 2

\* **@return** PackedDecimal result of ADD

\*/

**private** **static** LegacyNumber addWithRound(**final** **boolean** round, **final** LegacyNumber operand1, **final** Object operand2) {

**final** LegacyNumber fni1 = operand1;

**final** LegacyNumber fni2 = LegacyNumberUtils.*parse*(operand2);

**final** PackedDecimal ir = *getIntermediateResult*(ArithmeticOperation.***PLUS***, fni1, fni2, round);

**final** BigDecimal result = fni1.toBigDecimal().add(fni2.toBigDecimal(), MathContext.***UNLIMITED***);

**if** (round) {

ir.setRounded(result);

} **else** {

ir.set(result);

}

**return** ir;

}

Các ticket liên quan:

5096, 5384

## Ibm

<https://publib.boulder.ibm.com/iseries/v5r1/ic2924/books/c0925083654.htm>

EVAL (Evaluate expression)

Chỉ cung cấp 1 dạng biểu thức:

|  |  |
| --- | --- |
| **Free-Form Syntax** | {EVAL{(HMR)}} *result* = *expression* |

QTE transform:

* EVAL(H)

C EVAL(H) BP612Q = WK@KN\_SU2 \* WK@KN\_SU2

transform:

compute(BP612Q, 10).setRounded(multiply(wkAtKnSu2, wkAtKnSu2));

* EVAL(R)

C EVAL(R) BP612Q = BP612Q + WK@KN\_SU2

transform:

compute(BP612Q, 10, true).setRetained(add(BP612Q, wkAtKnSu2));

* EVAL(RH)

C EVAL(RH) BP612Q = WK@KN\_SU2 \* WK@KN\_SU2

transform:

compute(BP612Q, 10, true).setRetainedRounded(multiply(wkAtKnSu2, wkAtKnSu2));

The EVAL operation code evaluates an assignment statement of the form result=expression. The expression is evaluated and the result placed in **result**. Therefore, **result** cannot be a literal or constant but must be a field name, array name, array element, data structure, data structure subfield, or a string using the %SUBST built-in function.

Một biểu thức EVAL có dạng result = expression. Phần expression được tính toán và kết quả cuối cùng đặt trong result. Vì vậy, kết quả không thể là chữ hoặc hằng số, phải là tên trường, tên array, nhân tố array, cấu trúc data, trường con cấu trúc data, hoặc là một chuỗi sử dụng %SUBST built-in fuction.

The expression may yield any of the RPG data types. The type of the expression must be the same as the type of the result. A character, graphic, or UCS-2 result will be left justified and padded with blanks on the right or truncated as required. |If result is a variable-length field, its length will be |set to the length of the result of the expression.

Expression có thể mang bất kỳ kiểu data type nào của RPG. Type của expression phải giống như type của result. Kết quả của character, graphic hoặc UCS-2 sẽ bị left justified và padded (độn) với blank vào bên phải hoặc bị truncate theo yêu cầu. Nếu result có chiều dài có thể thay đổi, chiều dài của nó sẽ được set theo chiều dài của kết quả của expression.

If the result represents an unindexed array or an array specified as array(\*), the value of the expression is assigned to each element of the result, according to the rules described in Specifying an Array in Calculations. Otherwise, the expression is evaluated once and the value is placed into each element of the array or sub-array. For numeric expressions, the half-adjust operation code extender is allowed. The rules for half adjusting are equivalent to those for the arithmetic operations.

Nếu kết quả đại diện cho một unindexed array (mảng không thứ tự) hoặc một array được biệt như array(\*), giá trị của expression được assigned cho mỗi phần từ của result, dựa vào các quy tắc trong Specifying an Array in Calculations. Nếu không, expression được tính toán một lần và giá trị được đặt vào bên trong mỗi phần tử array hoặc sub-array. Đối với numeric expression, half-adjust operation (hoạt động bán điều chỉnh) code extender is allowed. Những quy tắc cho half adjusting thì tương đương cho phép tính số học (arithmetic operation)

On a free-form calculation specification, the operation code name may be omitted if no extenders are needed.

Trên một đặc điểm kỹ thuật tính toán tự do, tên operation code có thể được bỏ qua nếu không có extender.

See [Expressions](https://publib.boulder.ibm.com/iseries/v5r1/ic2924/books/c0925083527.htm#HDREXPR9) for general information on expressions. See [Precision Rules for Numeric Operations](https://publib.boulder.ibm.com/iseries/v5r1/ic2924/books/c0925083535.htm#HDRPRCSNR) for information on precision rules for numeric expressions. This is especially important if the expression contains any divide operations, or if the EVAL uses any of the operation extenders.

Xem expression cho thông tin chung cho expression. Xem Precision Rules for Numeric Operations để biết thông tin cho precision rules cho numeric expressions. Điều này thì đặc biệt quan trọng nếu expression contain any divide operations, or if the EVAl use any of the opreration extender.

|  |  |  |
| --- | --- | --- |
| **Extender** | **Description** | **Comments** |
| (H) | Half adjust (round the numeric result) |  |
| (M) | Default precision rules |  |
| (R) | ″Result Decimal Position″ precision rules | apply only to packed decimal intermediate results. |

Half adjust (round the numeric result)

Theo tài liệu tìm hiểu trên google thì thuật ngữ half adjusting được như sau:

<http://www.jaymoseley.com/hercules/rpgtutor/rpg007.htm>

**Half Adjusting**

“Half adjusting” is a term that refers to a method of rounding the results of calculations. Suppose, for example, that a program is to divide one three-digit value into a four-digit value and the result is to be rounded to four significant digits.  The answer, known as the quotient, might have any number of digits.  If the divisor were 035 and the dividend were 4671, the quotient would be 0133.462....  If we want to round this to four significant digits, we would write 0133.  If the digit in the tenths position had been five or greater, we would have written 0134.  When calculations are being performed by a computer, a way must be developed that will cause the computer to record the rounded value, 0134 in this case, rather than the 0133 which would result from simply dropping the low-order digits.  This can be done by adding the digit 5 one position to the right of the position containing the least significant digit of the answer. In our example, this would look like this:

0133.46 + .5 = 0133.96

“Half adjusting” là một thuật ngữ dùng để chỉ một phương pháp làm tròn kết quả tính toán. Giả sử, 1 chương trình chia 1 số có 3 chữ số cho 1 số có 4 chữ số thì kết quả được làm tròn thành số có 4 chữ số. Ví dụ lấy 4671 chia cho 035 thì kết quả là 0133.462… Nếu chúng ta muốn làm tròn thành 4 chữ số có nghĩa, chúng ta sẽ ghi 0133. Nếu chữ số ở phần 10 bằng 5 hoặc lớn hơn, thì ta ghi là 0134. Khi tính toán được thực hiện bởi máy tính, một cách phải được phát triển để làm cho máy tính làm tròn được giá trị, 0134 trong trường hợp này chứ không phải 0133 mà sẽ cho kết quả đơn giản là loại bỏ chữ số thứ tự thấp. Điều này có thể thực hiện bằng việc thêm 0.5

Now, dropping the two digits to the right of the decimal point yields the value 0133 which is the correct rounded value given a tenths position digit of less than 5.  If the result of a similar calculation were 1198.67, a similar rounding technique would result in the following:

1198.67 + .5 = 1199.17

Dropping the two unwanted digits yields, again, the correct rounded value of 1199.  Half adjusting is a technique that adds the digit 5 to the position immediately to the right of the least significant digit desired in a result data field.  In RPG programming, half adjusting of result fields is caused by entering the letter H in column 53 of the specification containing the arithmetic calculation operation to be half adjusted.

## setRounded

hàm setRounded nằm trong AbstractNumber.java

Tìm được 1 class có sử dụng + 0.5 là NumberUtils.java

## Tìm hiểu \*H\* ***compute*(workA, 3).setRounded(workD);**

compute(workA,3) thì 3 ở đây là scale – số chữ số sau dấu thập phân, tương đương dmax.

hàm compute sẽ so sánh scale của workA với 3, lấy max và bỏ vào ThreadLocal *calcDMAX*

ở đây có sử dụng ThreadLocal:

<http://drunkkid2000.blogspot.com.au/2013/07/thread-local_2564.html>

Thread là một object nhưng là một object được sử dụng để tạo ra môi trường thực thi cho các object khác.

Khi tìm hiểu truncated thì đượ cái này nằm trong arithmeticFunction.java:

**private** **static** PackedDecimal getIntermediateResult(**final** ArithmeticOperation operation, **final** LegacyNumber operand1,

**final** LegacyNumber operand2, **final** **boolean** round) {

// Get precision of the first operand1

**final** LegacyNumber fni1 = operand1;

**final** **int** d1 = fni1.getScale();

**final** **int** i1 = fni1.length() - d1;

**final** LegacyNumber fni2 = operand2;

**final** **int** d2 = fni2.getScale();

**final** **int** i2 = fni2.length() - d2;

/\*

\* Re-calculate dmax with specified operands dmax = maximum decimal places of all operands and result involved

\*/

**int** dmax = *getCalcDMAX*();//Math.max(getCalcDMAX(), getMaxScale(fni1, fni2));

**int** i;

**int** d;

**switch** (operation) {

**case** ***DIVIDE***:

i = i1 + d2;

d = Math.*max*(dmax, d1 - d2);

**break**;

**case** ***MULTIPLY***:

i = i1 + i2;

d = d1 + d2;

**break**;

**case** ***PLUS***:

**case** ***MINUS***:

i = Math.*max*(i1, i2) + 1;

d = Math.*max*(d1, d2);

**break**;

**case** ***POWER***:

**if** (i2 == 0) {

**if** (d2 > 0) {

i = Math.*max*(Math.*min*(i1, ***I18***), ***I1***);

} **else** {

i = Math.*max*(Math.*min*(i1 \* i1, ***I18***), ***I1***);

}

} **else** {

**if** (d2 > 0) {

i = Math.*max*(Math.*min*(i1 \* (***I9*** \* i2), ***I18***), ***I1***);

} **else** {

i = Math.*max*(Math.*min*(i1 \* i1 \* (***I9*** \* i2), ***I18***), ***I1***);

}

//

}

// Decimal places

**if** (d2 == 0) {

d = Double.*valueOf*(d1 \* fni2.toDouble()).intValue();

} **else** {

d = dmax;

}

**break**;

**default**:

**throw** **new** UnsupportedOperationException("Unsupported operation - " + operation);

}

**if** (round) {

**if** (d > 0) {

d++;

} **else** {

i++;

}

}

**if** (i + d > ***MAXIMUM\_LENGTH\_OF\_NUMBER***) {

**if** (d <= dmax) {

i = ***MAXIMUM\_LENGTH\_OF\_NUMBER*** - d;

} **else** {

**if** (i + dmax <= ***MAXIMUM\_LENGTH\_OF\_NUMBER***) {

d = ***MAXIMUM\_LENGTH\_OF\_NUMBER*** - i;

} **else** {

d = dmax;

i = ***MAXIMUM\_LENGTH\_OF\_NUMBER*** - dmax;

}

}

}

**return** DeclarationFunctions.*createPackedDecimal*(i + d, d);

}

**Operation Extender**

**Entry Explanation**

**Blank** No operation extension supplied.

**H** Half adjust (round) result of numeric operation

**M** Default precision rules

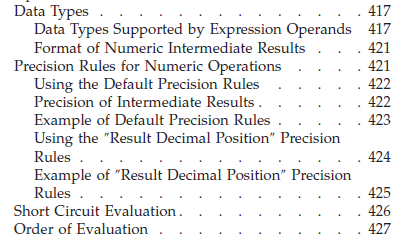
**R** ″Result Decimal Position″ precision rules

**E** Error handling

**H: Half adjust (round) result of numeric operation**

**R: Result Decimal Position precision rules**

**M: Default precision rules.**

****

**///// Đoạn này ngoài lề rồi**

**Format of Numeric Intermediate Results**

For binary operations involving numeric fields, the format of the intermediate

result depends on the format of the operands.

**For the operators +, -, and \*:**

v If at least one operand has a float format, the result is float format.

v Otherwise, if at least one operand has packed-decimal, zoned-decimal, or binary

format, the result has packed-decimal format.

v Otherwise, if at least one operand has integer format, the result has integer

format.

v Otherwise, the result has unsigned format.

v For numeric literals that are not in float format:

– If the literal is within the range of an unsigned integer, the literal is assumed

to be an unsigned integer.

Nếu literal(giá trị 1 chữ) nằm trong vùng số nguyên không dấu, thì literal đó được xem như là số nguyên không dấu.

– Otherwise, if the literal is within the range of an integer, the literal is assumed

to be an integer.

Nếu không, nếu literal đó nằm trong vùng số nguyên có dấu (integer), thì literal đó được xem như số nguyên có dấu

– Otherwise, the literal is assumed to be packed decimal.

Nếu không, literal đó được xem như packed decimal

**For the / operator:**

If one operand is float or the FLTDIV keyword is specified on the control

specification, then the result of the / operator is float. Otherwise the result is

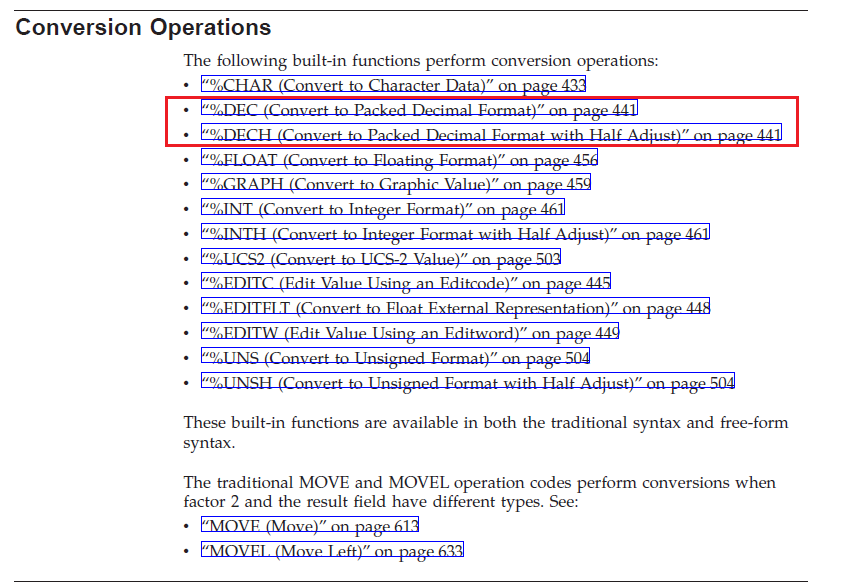
packed-decimal.

**For the \*\* operator:**

The result is represented in float format.

/////

Trang 387 có đoạn:



## Các file đã sửa:

ArithmeticFunctions

ArithmeticFunctionsTest

## Các ticket liên quan:

5384

5354 – QTE của anh Khương

4601 – cobol của anh Huyền

5096 – anh Khương

## RPGIV Manual:

**EXPROPTS(\*MAXDIGITS | \*RESDECPOS)**

The EXPROPTS (expression options) keyword specifies the type of precision rules

to be used for an entire program. If not specified or specified with \*MAXDIGITS,

the default precision rules apply. If EXPROPTS is specified, with \*RESDECPOS, the

″Result Decimal Position″ precision rules apply and force intermediate results in

expressions to have no fewer decimal positions than the result.

**Note:** Operation code extenders R and M are the same as

EXPROPTS(\*RESDECPOS) and EXPROPTS(\*MAXDIGITS) respectively, but

for single free-form expressions.

**Using the** ″**Result Decimal Position**″ **Precision Rules**

The ″Result Decimal Position″ precision rule means that the precision of a decimal

intermediate will be computed such that the number of decimal places will never

be reduced smaller than the number of decimal positions of the result of the

assignment.

Số chữ số sau dấu phẩy của intermediate (số trung gian) sẽ không bao giờ được giảm nhỏ hơn số sau dấu phẩy của kết quả của assignment.

This is specified by:

1. **EXPROPTS(\*RESDECPOS)** on the Control Specification. Use this to specify

this behaviour for an entire module.

2. Operation code extender **R** specified for a free-form operation.

Result Decimal Position rules apply in the following circumstances:

# File có thể test private:

SpoolFileManagerJdbcTest

# 19. Các câu lệnh hay

## 19.1 Copy kiểu T

T result = LegacyDataUtils.copy(arg);

Trong đó arg được khai bái là kiểu T. Method đầy đủ được thực hiện trong EVAL(R):

**public** **static** <T **extends** LegacyData> T compute(**final** T arg, **final** **int** dmax, **final** **boolean** resultDecimalPositions) {

**final** T result = LegacyDataUtils.*copy*(arg);

**if** (resultDecimalPositions) {

**return** *compute*(arg, dmax);

} **else** {

**return** *compute*(arg, dmax);

}

}

# 20. QTE-5278 (postponed)

## Đọc tài liệu:

Ibm: <https://www.ibm.com/support/knowledgecenter/ssw_i5_54/apis/QDCXLATE.htm>

Send Message API: <http://www.ibm.com/support/knowledgecenter/ssw_i5_54/apis/QEZSNDMG.htm>

Retrieve Message API: <http://www.ibm.com/support/knowledgecenter/ssw_i5_54/apis/QMHRTVM.htm>

Retrieve User Space API: <https://www.ibm.com/support/knowledgecenter/ssw_i5_54/apis/qusrtvus.htm>

Excute Command API: <https://www.ibm.com/support/knowledgecenter/ssw_i5_54/apis/qcmdexc.htm>

Đọc tài liệu cái Convert Data API:

The Convert Data (QDCXLATE) API converts data through the use of a table object. (If you need to convert the case of your data, it is recommended that you use the [Convert Case](https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/apis/QLGCNVCS.htm?view=kc) (QLGCNVCS, QlgConvertCase API.) You also can use the equivalent API QTBXLATE to achieve the same function. The call interface to QTBXLATE is identical to Convert Data (QDCXLATE).

Convert Data (QDCXLATE) API chuyển đổi dữ liệu thông qua việc sử dụng một đối tượng table. Nếu bạn muốn chuyển đổi các trường hợp dữ liệu của bạn, nó được đề nghị sử dụng Convert Case (QLGCNVCS, QlgConvertCase API). Bạn cũng có thể sử dụng QTBXLATE tương đương để thực hiện chức năng giống như vậy. Việc gọi đến interface QTBXLATE giống hệt QDCXLATE.

This API is available for compatibility purposes or user-defined mappings only. Do not use this API in new development; instead, use the [iconv()--Code Conversion API](https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/apis/iconv.htm?view=kc) or the [Convert a Graphic Character String (CDRCVRT) API](https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/apis/CDRCVRT.htm?view=kc).End of change



API này chỉ sử dụng cho mục đích tương thích hoặc tự định nghĩa ánh xạ (user-defined mappings). Không sử dụng API này để phát triển mới. (Nghĩa là chuyển đổi qua lại giữa ASCII và EBCDIC chứ không phát triển mã code mới).

You can create the conversion table that QDCXLATE uses for the conversion, or you can use an IBM-supplied table. The IBM-supplied tables can be found in the QUSRSYS library. For a list of both the conversion tables and the casing tables, see the [Globalization](https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/nls/rbagsglobalmain.htm?view=kc) topic in the iSeries Information Center. These tables are not the same as those used by the intersystem communications function (ICF) for conversion support. For information on this, see [Sockets Programming](https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/rzab6/rzab6soxoverview.htm?view=kc).

Bạn có thể tạo ra bảng chuyển đổi cho QDCXLATE sử dụng cho việc chuyển đổi, có thể sử dụng bảng do IBM cung cấp (hình như cái này trong ticket có cung cấp 1 đường link về cái bảng chuyển đổi này thì phải). Bảng IBM có thể tìm thấy trong QUSRSYS library.

You can create your own conversion tables using the Create Table (CRTTBL) command.

Bạn có thể tự tạo cho riêng mình bảng chuyển đổi bằng command CRTTBL

When the QDCXLATE API is called with parameters 1, 2, 3, and 4, it converts single-byte data. When all parameters are specified, DBCS conversion is taking place.

Khi QDCXLATE API được gọi với param 1,2,3,4, nó sẽ được chuyển đổi sang single-byte. Khi tất cả các parameter được định rõ, được xác định. DBCS conversion bị loại bỏ.

The QDCXLATE API can distinguish double-byte from single-byte characters when converting from EBCDIC to ASCII and from ASCII to EBCDIC if the proper parameters have been supplied. The QDCXLATE API converts data byte for byte and returns the converted data to your program.

QDCXLATE API có thể phân biệt được double-byte và single-byte khi chuyển đổi từ EBCDIC sang ASCII và ngược lại nếu được cung cấp parameter thích hợp. QDCXLATE API chuyển đổi data byte sang byte và trả về dữ liệu đã được chuyển đổi cho chương trình của bạn.

When only single-byte data is converted, the input (unconverted) data is replaced with the converted data. When double-byte data is converted, the converted data is placed in the output data parameter.

Chỉ khi single-byte data được chuyển đổi, data đầu vào (chưa chuyển đổi) bị thay đổi bởi dữ liệu đã được chuyển đổi. Khi double-byte data được chuyển đổi, data đã được chuyển đổi được đặt vào trong tham số data đầu ra.

The QDCXLATE API is thread safe only when converting single-byte data or T.61 data.

Thread safe chỉ được sử dụng khi chuyển đổi single-byte hoặc T.61.

### Required Parameter Group

**Length of data being converted**

INPUT; PACKED(5,0)

The length of the data being converted. This value cannot exceed 32 767.

**Conversion data**

I/O; CHAR(\*)

The data to be converted. This buffer also contains the output data after conversion when the API is called with only the required parameter group.

Đây là data chính, trong file parameter mình làm thì có tên là **data.** Buffer này cũng chứa dữ liệu đầu ra sau khi chuyển đổi khi API được gọi chỉ với các parameter bắt buộc.

**SBCS conversion table name**

INPUT; CHAR(10)

The name of the single-byte character set (SBCS) conversion table to be used. The table may be a system-supplied or user-supplied conversion table. The table name must be left-justified.

Tên của bộ ký tự bảng chuyển đổi single-byte (SBCS) sẽ được sử dụng. Bảng chuyển đổi do hệ thống cung cấp hoặc user tự define. Tên của table phải được canh lề trái (left-justified)

**Note:** This parameter is ignored when the DBCS language parameter is set to \*BG5, \*KSC, \*SCGS \*J90X5026, \*J90X5035, or \*SCGBK.

### Optional Parameter Group

**SBCS conversion table library name**

INPUT; CHAR(10)

The name of the library that contains the SBCS conversion table. The library name must be left-justified. If this parameter is not specified, the library list is used to locate the conversion table. This parameter is ignored when the DBCS language parameter is set to \*BG5, \*KSC, \*SCGS, \*J90X5026, \*J90X5035 or \*SCGBK.

**Output data**

OUTPUT; CHAR(\*)

The output buffer that contains the double-byte character set (DBCS) data that was converted.

Because of the insertion of shift-out and shift-in characters, it is possible that the converted data is longer than the source data. If this is the case, it is not possible to do the conversion in place, as is done when you use only a required-parameter-group call. The converted data is then placed in the area pointed to by this parameter.

Buffer đầu ra chứa dữ liệu double-byte, trong parameter mình làm là outputData.

Bởi vì sự thêm vào của ký tự shift-out và shift-in, nó có thể chuyển đổi data dài hơn data gốc. Trong trường hợp này, điều đó là không thể thực hiện sự chuyển đổi tại chỗ như khi chỉ sử dụng các param bắt buộc. Data đã chuyển đổi được đặt đển nơi trỏ đến của param này.

**Length of output buffer**

INPUT; PACKED(5,0)

The size of the output data buffer. The maximum length should match the actual size of the output data parameter. If the converted output is longer than the length of output buffer parameter, an exception is signaled.

**Length of converted data**

OUTPUT; PACKED(5,0)

The actual length of the converted output in the output data parameter.

**DBCS language**

INPUT; CHAR(10)

The DBCS language that is being converted. All values must be padded on the right with blanks. The possible values follow:

|  |  |
| --- | --- |
| \*JPN | IBM Japanese graphic character set |
| \*KOR | IBM Korean graphic character set |
| \*CHS | IBM Simplified Chinese graphic character set |
| \*CHT | IBM Traditional Chinese graphic character set |
| \*BG5 | Taiwan industry standard graphic character set (BIG-5) |
| \*KSC | Korean industry standard graphic character set (KS) |
| \*SCGS | The People's Republic of China National standard graphic character set (GB) |
| \*J90X5026 | The Japanese JIS X 0208 1990 standard mapped using CCSID 5026. |
| \*J90X5035 | The Japanese JIS X 0208 1990 standard mapped using CCSID 5035. |
| \*SCGBK | The People's Republic of China National standard graphic character set extended (GBK) |

**Shift-out and shift-in characters**

INPUT; CHAR(1)

Whether shift-out and shift-in characters should be inserted during the conversion. This parameter is ignored when the DBCS language parameter is set to \*BG5, \*KSC, \*SCGS, \*J90X5026, \*J90X5035 or \*SCGBK. The possible values follow:

|  |  |
| --- | --- |
| Y | Insert shift-out and shift-in characters |
| N | Do not insert shift-out and shift-in characters |

**Type of conversion**

INPUT; CHAR(10)

The type of DBCS conversion being done. The possible values follow:

|  |  |
| --- | --- |
| \*AE | Convert ASCII to EBCDIC |
| \*EA | Convert EBCDIC to ASCII |

**Note:** You are responsible for specifying the correct SBCS table name for the type of conversion being done by this DBCS request except when the DBCS language parameter is set to \*BG5, \*KSC, \*SCGS, \*J90X5026, \*J90X5035 or \*SCGBK.

## Các file có chữ **TODO** QTE-5278:

ConvertData

ConvertDataParameter

IseriesApiExecutor

## Các file đã sửa:

IseriesApiExecutor: trong đó có 1 hàm excute chưa biết làm gì với nó

Tuy nhiên có 1 cái sendmessage khá là giống phần mình làm, có hẳn file sendmessagetest.java

ConvertData

ConvertDataImpl

ConverDataParameter

IseriesApiExecutor

Cpf269c

IseriesErrors

qre-iseries-api.xml: đã thêm zô nhưng k biết đúng k, có cần thiết k.

## Các ticket liên quan:

QTE-3802 : ticket này y chang cái mình cần làm luôn.

QTE-428: ticket bao gồm nhiều message

## Các file có thể tham khảo:

File parameter này khá khá giống parameter của convertdata mình làm: SendDataQueueParameter

## Làm sao để làm hàm doExcute đây???

### RetrieveDataQueueDescriptionImpl

@Autowired

**private** DataQueueManager dataQueueManager;

@Override

**protected** **void** doExecute(**final** Object from, **final** RetrieveDataQueueDescriptionParameter parameter) {

DataQueue queue = dataQueueManager.getQueue(parameter.getQueueName().trim(), parameter.getLibrary().trim());

// Queue does not exists

**if** (queue == **null**) {

**throw** **new** Cpf9801(parameter.getQueueName().trim(), parameter.getLibrary()

.trim());

} **else** {

// lock before used

**boolean** locked = queue.lock(IseriesObjectLockType.***EXCLRD***, ***LOCK\_TIMEOUT***);

**if** (!locked) {

**throw** **new** Cpf2114(parameter.getQueueName(), parameter.getLibrary(), IseriesObjectType.***DTAQ***);

}

**try** {

parameter.setReceiver(queue.retrieveInfo(parameter.getFormat(), parameter.getReceiverLen()));

} **finally** {

queue.unlock(IseriesObjectLockType.***EXCLRD***);

}

}

}

### SendMessage

/\*\*

\* nonProgramMessenger use in API

\*/

**private** IseriesNonProgramMessenger nonProgramMessenger;

@Override

**protected** **void** doExecute(**final** Object from, **final** SendMessageParameter parameter) {

**int** numberOfUsers = parameter.getNoOfDestinations();

**for**(**int** i = 0;i < numberOfUsers; i ++) {

String userProfile = parameter.getDestinations()[i];

String messageType = parameter.getMessageType();

String messageText = parameter.getMessage();

**if** (StringUtils.*isEmpty*(messageText)) {

messageText = parameter.getMessageData();

}

String replyQueue = parameter.getQualifiedReplyQueue();

LegacyString errorCode = parameter.getErrorCode();

nonProgramMessenger.sendSingleUserMessage(from, userProfile, messageType, messageText, replyQueue, errorCode);

}

// currently we don't check names valid or not, so message sent indicator always store 1

LegacyNumber binary =parameter.getMessageSentIndicator();

**if**(binary != **null**) binary.set(1);

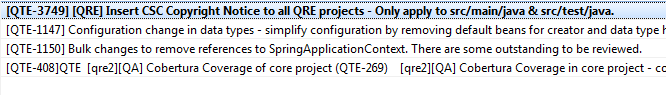
}

**public** **void** setNonProgramMessenger(**final** IseriesNonProgramMessenger nonProgramMessenger) {

**this**.nonProgramMessenger = nonProgramMessenger;

}

# 21. Commit code:



Hi Max,

Hi have resolved this ticket, please help me to review.

Revision: 1727

Author: tnguyen443

Date: Monday, January 16, 2017 11:25:27 AM

Message:

QTE-5188: [DAITO][ITER2.2][QRE] Implement EVAL with Extenders (BPF205R)

----

Modified : /trunk/src/main/java/com/csc/qre/lang/ArithmeticFunctions.java

Modified : /trunk/src/test/java/com/csc/qre/lang/ArithmeticFunctionsTest.java

Thanks and Regards,  
Thuat T Nguyen

# 22. QTE-428: cái này tham khảo chứ mình k làm

To implement messaging related APIs as well as supporting classes.   
Three class diagrams have been attached corresponding to three API to be implemented.   
For supporting classes, refer to 3.3.4 and 3.3.5 in [QRE2 Utilities Design document[https://vdc-jira.asia.csc.com/jira/images/icons/linkext7.gif](https://c3.csc.com/docs/DOC-195525)](https://c3.csc.com/docs/DOC-195525), which has been revised recently.   
Besides, refer to [System API design document[https://vdc-jira.asia.csc.com/jira/images/icons/linkext7.gif](https://c3.csc.com/docs/DOC-406657)](https://c3.csc.com/docs/DOC-406657) 9.3 for API implemetation itself.

# 23. QTE-3014: QWCRNETA

Cái này cũng không sử dụng cái manager, có thể tham khảo.

# 24. QTE-5590: 4 method init (closed)

iSeries specific data type, which behaves like boolean type data. what different to boolean is Indicator may have a no associated with it. In such case, the indicator will be associated to a value works like global variable.

(Kiểu dữ liệu iSeries cụ thể, tựa như boolean. Điểm khác biệt là Indicator không có liên kết với nó. Trong trường hợp này, Indicator sẽ liên kết với 1 giá trị như biến toàn cầu).

## Init thứ 1

Indicator[] switches = *initArray*(**new** Indicator[11]);

Kết quả của swithces có 11 giá trị: [null, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

## Init thứ 3

**final** FixedLengthString newValue2 = **new** DataStructure<FixedLengthString>(999);

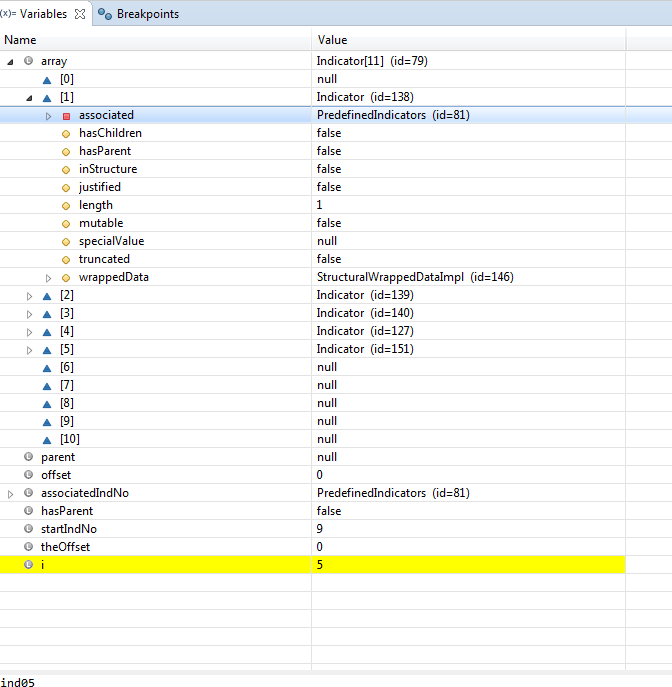
Indicator[] switches = *initArray*(**new** Indicator[11],newValue2,16);

[null, , , , , , , , , , ]

## Init thứ 2

Indicator[] switches = *initArray*(**new** Indicator[11],***IND05***);

[null, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]



Ta thấy giá trị 1 sẽ có associated là ind05.

## Init thứ 4

## Tìm hiểu

ArrayDeclarationFunctions có cấu trúc khá giống cái mình làm

# 25. QTE-5559: DisplaySystemValue (closed)

## 1. Các file đã sửa

qre-iseries-cl.xml (edit)

DisplaySystemValue (add)

DisplaySystemValueParameter (add)

DisplaySystemValueImpl (add)

ClCommandExecutor

DisplaySystemValueTest

## 2. Các ticket liên quan:

2616

2541

## 3. Tìm hiểu

**QDYNPTYADJ**

Dynamic priority adjustment. The QDYNPTYADJ system value controls whether the priority of interactive jobs is dynamically adjusted to maintain high performance of batch job processing. This adjustment capability is only effective on systems that are rated for both interactive and non-interactive throughput and have Dynamic Priority Scheduling enabled. A change to this value takes effect at the next IPL.

* 0 means the dynamic priority adjustment support is turned off.
* 1 means the dynamic priority adjustment support is turned on.

LanguageFunctions có Goto khá giống isMode

## 4. Chỉnh sửa theo Max:

Changed

* SystemValueRepository
* DisplaySystemValueImpl
* DisplaySystemValueTest.

pls review and sycn changes if everything is understood & no issues.

DisplaySystemValue: không đổi

DisplaySystemValueParameter: đổi systemvalue thành nhiều giá trị.

DisplaySystemValueImpl: batch mode ghi ra file log, online mode thì throw ra exception

ClCommandExecutor: không đổi

SystemValueRepository: refactor lại code

qre-iseries-cl.xml: không đổi

DisplaySystemValueTest: test batch mode và onlinde mode.

# 26. Cách tạo 1 test class:

\_ phải extend (ví dụ **extends** AbstractIseriesTestSupport)

\_ Phải thêm vào file bean: ví dụ qre-iseries-cl.xml

# 27. Về file log

Tạo file log trong Impl (lấy ví dụ QTE-5559):

/\*\*

\* Logger

\*/

**private** **static** **final** Logger ***logger*** = LoggerFactory.*getLogger*(DisplaySystemValueImpl.**class**);

***logger***.info("Displaying system value {}", parameter.getSystemValue());

## Nơi các file log sẽ được in ra:

C:\Workspace\QRE2\qre2-core\target\log

C:\Workspace\QRE2\qre2-iseries\target\log

# 28. QTE-5597: Copy Source File (CPYSRCF)

## 1. Các file đã sửa:

CopySourceFile (add)

CopySourceFileParameter (add)

CopySourceFileImpl (add)

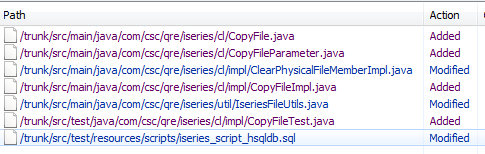
qre-iseries-cl.xml (edit)

CopySourceFileTest (add)

Iseries\_script\_hsqldb (edit)

## 2. Các ticket liên quan:

922



2541

4487

## 3. Tìm hiểu:

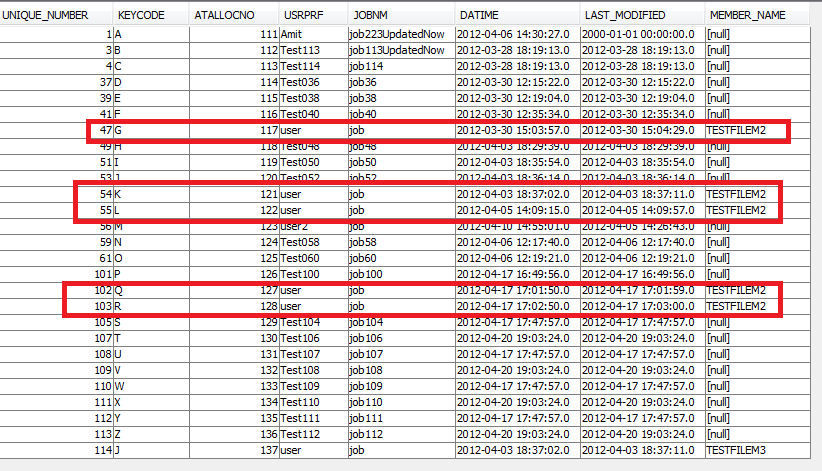
<http://www.ibm.com/support/knowledgecenter/ssw_ibm_i_72/cl/cpysrcf.htm>

### CopyFile

trước khi copy:

FromFile: ATNMPF

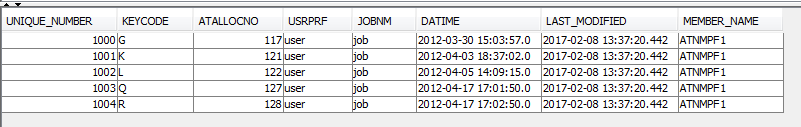
FromMember: TESTFILEM2



Sau khi copy:

ToFile: atnmpf1

ToMember: atnmpf1



Sau khi chạy bùi nhùi thì được câu sql:

INSERT INTO ISERIESPOC.ATNMPF1 (UNIQUE\_NUMBER,LAST\_MODIFIED,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME,MEMBER\_NAME) SELECT ISERIESPOC.ATNMPF1\_SEQ.NEXTVAL,CURRENT\_TIMESTAMP,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME,'ATNMPF1' FROM (SELECT row\_.\*, ROWNUM rownum\_ FROM (SELECT UNIQUE\_NUMBER,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME FROM ISERIESPOC.ATNMPF WHERE 1=1 AND MEMBER\_NAME = ? AND UNIQUE\_NUMBER >= ? AND UNIQUE\_NUMBER <= ? ORDER BY UNIQUE\_NUMBER ASC ) row\_ WHERE ROWNUM <= 9223372036854775807 )

sqlInsertColumns:

UNIQUE\_NUMBER,LAST\_MODIFIED,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME,MEMBER\_NAME

sqlSelectColumns:

ISERIESPOC.ATNMPF1\_SEQ.NEXTVAL,CURRENT\_TIMESTAMP,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME,'ATNMPF1'

## 4. Do it

Sau khi làm sơ sơ ta được câu sql cùi chuối sau đây:

INSERT INTO ISERIESPOC.ATNMPF1 (UNIQUE\_NUMBER,LAST\_MODIFIED,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME,MEMBER\_NAME) SELECT ISERIESPOC.ATNMPF1\_SEQ.NEXTVAL,CURRENT\_TIMESTAMP,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME,'ATNMPF1' FROM (SELECT UNIQUE\_NUMBER,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME FROM ISERIESPOC.ATNMPF WHERE 1=1 ORDER BY UNIQUE\_NUMBER ASC )

Có gì đó sai sai

sqlInsertColumns: đúng

UNIQUE\_NUMBER,LAST\_MODIFIED,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME,MEMBER\_NAME

sqlSelectColumns: đúng

ISERIESPOC.ATNMPF1\_SEQ.NEXTVAL,CURRENT\_TIMESTAMP,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME,'ATNMPF1'

## 5. So sánh với max’s patch

CopySourceFileTest: không đổi

CopySourceFile: không đổi

qre-iseries-cl.xml: đổi tên đầy đủ thành mã code

CopySourceFileParameter: không đổi

CopySourceFileImpl: oh man đổi hết k liên quan gì cái cũ cả.

## 6. ConstrucDto

**public** **static** IseriesDto constructDto(**final** String schema, **final** String name) { schema=ISERIESPOC, name=ATNMPF

**final** String qualifiedName = JdbcUtils.*getQualifiedName*(schema, name, JdbcUtils.*isTemporarySchema*(schema)); qualifiedName=ISERIESPOC.ATNMPF

**final** String dynamicTableName = IseriesDaoHelper.*getDynamicTable*(qualifiedName); dynamicTableName=null

**final** String tableName4Manipulate = dynamicTableName == **null** ? name : dynamicTableName; table4Manipulate=ATNMPF

**final** Class<?> actualDtoClass = InMemoryCacheManager.*instance*().loadDtoClass(tableName4Manipulate);

Class<?> dtoClass = InMemoryCacheManager.*instance*().loadDtoClass(tableName4Manipulate);

**if** (dtoClass.isAnnotationPresent(FlattenTable.**class**)) {

// flatten DTO

**final** **int** flattenTableSize = *getFlatRecordSize*(schema, name, tableName4Manipulate);

**if** (flattenTableSize <= 0) {

**throw** **new** IseriesGenericException(String.*format*("Flat record size for file %s.%s is zero", schema, name));

}

**final** FixedLengthString inRec = *createFixedLengthString*(flattenTableSize);

**final** Constructor<?> ctor = BeanUtils.*getConstructor*(dtoClass, FixedLengthString.**class**);

**return** (IseriesDto) BeanUtils.*instantiateClass*(ctor, inRec);

} **else** {

// specific DTO?

**return** (IseriesDto) BeanUtils.*instantiate*(actualDtoClass);

}

}

/\*\*

\* Get flat record size for specified schema and table

\*

\* **@param** schema schema

\* **@param** name name

\* **@param** colName column name

\* **@return** flat record size

\*/

**public** **static** **int** getFlatRecordSize(**final** String schema, **final** String name, **final** String colName) {

**int** flattenTableDataSize = JdbcUtils.*getFlatRecordSize*(IseriesFileUtils.*getJdbcTemplate*(schema, **false**), schema, name,

colName);

**if** (flattenTableDataSize <= 0) {

**final** IseriesDataAccessContext idac = IseriesDataAccessContextHolder.*getContext*();

flattenTableDataSize = idac.getFlatRecSizeMapping(schema + '.' + name);

}

**return** flattenTableDataSize;

}

## 7. isSameStructureTable

----------------------+++++++++++++++------------------------

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC3

fromTable: FBUTUICOL1

toSchema: ISERIESPOC3

toTable: FLATFILE4

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF3

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF3

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC3

fromTable: COPYTEST3

toSchema: ISERIESPOC

toTable: FBUTUIS

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC3

fromTable: FLATFILE3

toSchema: ISERIESPOC3

toTable: INCCHAR

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC3

fromTable: FLATFILE3

toSchema: ISERIESPOC3

toTable: INCCHAR

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC3

fromTable: FLATFILE3

toSchema: ISERIESPOC3

toTable: INCCHAR

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC3

fromTable: FLATFILE3

toSchema: ISERIESPOC3

toTable: INCCHAR

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC3

fromTable: FBUTUICOL1

toSchema: ISERIESPOC3

toTable: FBUTUICOL

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC

fromTable: ATNMPF7

toSchema: ISERIESPOC

toTable: ATNMPF8

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC

fromTable: ATNMPF7

toSchema: ISERIESPOC

toTable: ATNMPF8

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF3

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF3

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC3

fromTable: COPYTEST3

toSchema: ISERIESPOC3

toTable: FBUTUICOL

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC3

fromTable: FBUTUICOL1

toSchema: ISERIESPOC3

toTable: COPYTEST1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC3

fromTable: FBUTUICOL1

toSchema: ISERIESPOC3

toTable: FBUTUICOL

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC3

fromTable: FLATFILE3

toSchema: ISERIESPOC3

toTable: FLATFILE4

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC3

fromTable: FLATFILE3

toSchema: ISERIESPOC3

toTable: FLATFILE4

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC3

fromTable: FBUTUICOL1

toSchema: ISERIESPOC3

toTable: FBUTUICOL

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC

fromTable: ATNMPF7

toSchema: ISERIESPOC

toTable: ATNMPF8

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC3

fromTable: FBUTUICOL1

toSchema: ISERIESPOC3

toTable: FLATFILE4

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC3

fromTable: FBUTUICOL1

toSchema: ISERIESPOC3

toTable: COPYTEST1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC3

fromTable: FBUTUICOL1

toSchema: ISERIESPOC3

toTable: COPYTEST1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF6

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF6

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF6

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF6

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF6

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF6

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF6

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF6

toSchema: ISERIESPOC

toTable: ATNMPF11

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF

toSchema: ISERIESPOC

toTable: ATNMPF12

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: false

fromSchema: ISERIESPOC3

fromTable: FBUTUICOL1

toSchema: ISERIESPOC3

toTable: FLATFILE4

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC3

fromTable: COPYTEST3

toSchema: ISERIESPOC3

toTable: COPYTEST9

++++++++++++++++++++++++

MemberHandlerSeparateTable

isSameStructureTables: true

fromSchema: ISERIESPOC

fromTable: ATNMPF6

toSchema: ISERIESPOC

toTable: ATNMPF1

++++++++++++++++++++++++

# 7. Table trong hsqldb:

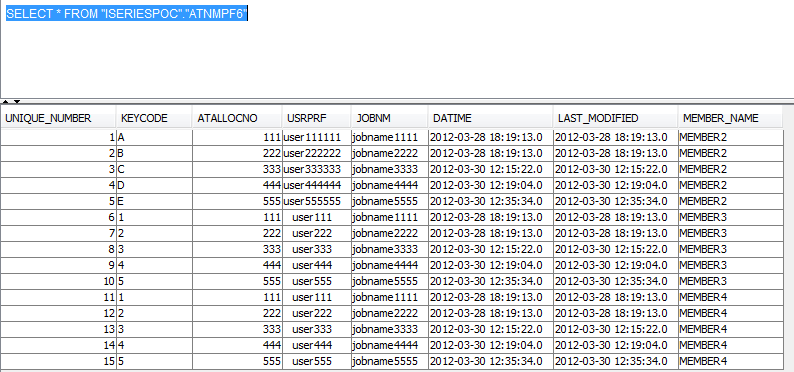
ATNMPF: … - multiple table

ATNMPF1: DB File – multiple table

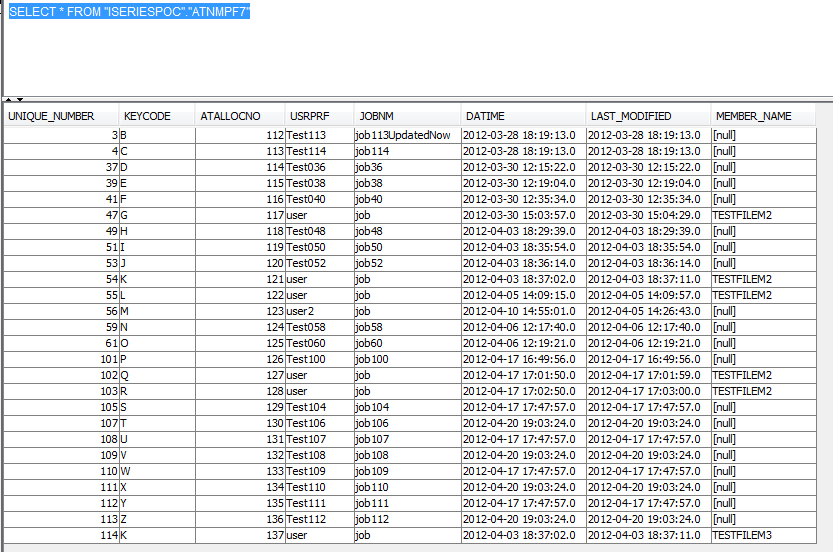
ATNMPF3: DB File – multiple table

ATNMPF6: … - single table

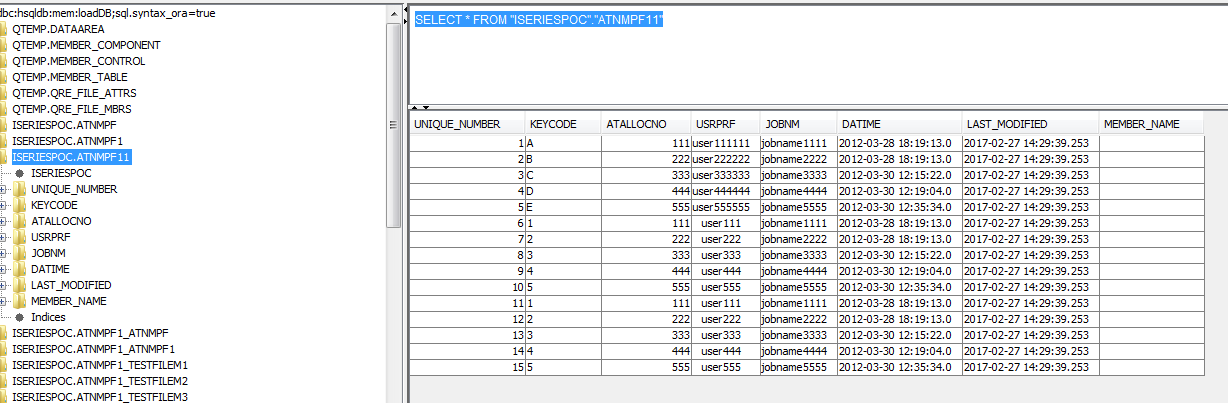
FBUTUICOL1: flat file -



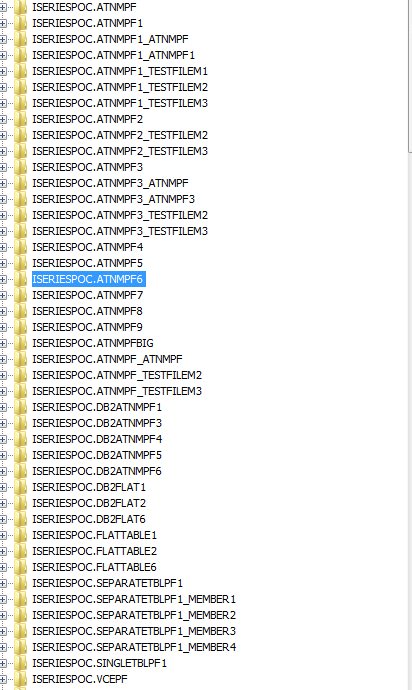
ATNMPF7:



ATNMPF11



# \*\*\*\*Tổng quan các table có:



## 8. Không hiểu tại sao sai

Param copy file:

FROMFILE(\*LIBL/ATNMPF) TOFILE(\*LIBL/ATNMPF1) FROMMBR(\*FIRST) TOMBR(ATNMPF1) MBROPT(\*ADD) FROMKEY(UNIQUE\_NUMBER 47) INCREL) FMTOPT(\*MAP \*DROP) ERRLVL(0)

INSERT INTO ISERIESPOC.ATNMPF1\_ATNMPF1 (UNIQUE\_NUMBER,LAST\_MODIFIED,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME,MEMBER\_NAME) SELECT ISERIESPOC.ATNMPF1\_ATNMPF1\_SEQ.NEXTVAL,CURRENT\_TIMESTAMP,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME,MEMBER\_NAME FROM (SELECT row\_.\*, ROWNUM rownum\_ FROM (SELECT UNIQUE\_NUMBER,KEYCODE,ATALLOCNO,USRPRF,JOBNM,DATIME FROM ISERIESPOC.ATNMPF\_TESTFILEM2 WHERE 1=1 ORDER BY UNIQUE\_NUMBER ASC ) row\_ WHERE ROWNUM <= 9223372036854775807 )

Đúng với antmpf6 và atnmpf1:

INSERT INTO ISERIESPOC3.COPYTEST9 (UNIQUE\_NUMBER,LAST\_MODIFIED,NUM\_COL,DEC\_COL,CHAR\_COL) SELECT ISERIESPOC3.COPYTEST9\_SEQ.NEXTVAL,CURRENT\_TIMESTAMP,NUM\_COL,DEC\_COL,CHAR\_COL FROM (SELECT row\_.\*, ROWNUM rownum\_ FROM (SELECT UNIQUE\_NUMBER,NUM\_COL,DEC\_COL,CHAR\_COL FROM ISERIESPOC3.COPYTEST3 WHERE 1=1 ORDER BY UNIQUE\_NUMBER ASC ) row\_ WHERE ROWNUM <= 9223372036854775807 )

Bị lỗi trong buildSqlForInputTable

## 9. Những cái ok:

testCopyWithMemberOption

testCopyFlatFileWithoutSameTableStructure

testCopyNotFoundFromSourceFile

testCopyFileToNotFoundTargetTable

10. Chat voi Max:

[‎3/‎1/‎2017 2:25 PM]

Hi Max, I have edited code to meet your requirement. Please help me to check again.

[‎3/‎1/‎2017 4:22 PM] Wang, Jinhai:

Hi Thaut,

[‎3/‎1/‎2017 4:22 PM] Wang, Jinhai:

thanks

[‎3/‎1/‎2017 4:22 PM] Wang, Jinhai:

will check it soon

We saved this conversation. You'll see it soon in the Conversations tab in Skype for Business and in the Conversation History folder in Outlook.

[‎3/‎1/‎2017 4:35 PM] Wang, Jinhai:

sorry Thuat,

I think you misunderstood my comments

firstly

[‎3/‎1/‎2017 4:37 PM] Wang, Jinhai:

why you still testing files with different structure?

as I said, source file should always have the same structure

[‎3/‎1/‎2017 4:38 PM]

I use ATNMPF, ATNMPF3, they are same structure table

[‎3/‎1/‎2017 4:38 PM] Wang, Jinhai:

2ndly, I think copyfile still have a defect - hum...

I thought it has been fixed.

[‎3/‎1/‎2017 4:40 PM]

Can you clarify which part I miss in comment?

[‎3/‎1/‎2017 4:41 PM] Wang, Jinhai:

two comments

1) source file should always have the same structure

2) copy file - when target member doesn't exist, it should create member automatically. you don't have to created yourself.

it's a defect in copyfile, which I thought it had been fixed.

but looks like validation part is not removed.

[‎3/‎1/‎2017 4:42 PM] Wang, Jinhai:

I will apply fix and pass it to Nguyen to review

[‎3/‎1/‎2017 4:43 PM]

1) ATNMPF, ATNMPF1, ATNMPF3. They are same structure, right?

[‎3/‎1/‎2017 4:52 PM] Wang, Jinhai:

cpysrcf.patch cpyf.patch

## dòng code đem ra rồi đem zô:

**final** IseriesObjectQualifier toQualifier = copyParameter.getToFile();

**final** String toTable = toQualifier.getName();

**final** String toSchema = IseriesFileUtils.*resolveSchema*(toQualifier.getLibrary(), toTable);

**final** String toMember = parameter.getToMember();

**final** **boolean** isToTableExists = !StringUtils.*isEmpty*(toSchema) && JdbcUtils.*isTableExists*(toSchema, toTable);

**if** (isToTableExists) {

**final** **boolean** isToMemberExists = memberManager.isMemberExisting(toSchema, toTable, toMember);

**if** (!isToMemberExists) {

copyParameter.setCreateFile(**true**);

}

}

## 10. File sửa thêm:

Iseries\_script\_hsqldb.sql

qre-test-iseries-member-attributes\_hsqldb

qre-test-iseries-member-control\_hsqldb (Cái này k cần)

copyfileImpl

# 29. Database

Các hiện hsql:

Mở tab display và thêm lệnh:

org.hsqldb.util.DatabaseManagerSwing.main(**new** String[] {

"--url", "jdbc:hsqldb:mem:loadDB;sql.syntax\_ora=true", "--noexit"

});

# 30. Trang download sách:

<http://www.allitebooks.com>

# 31. QTE-5733 Merge Error Message (closed)

## 1. Các file đã sửa:

IseriesErrors Y

qre-iseries-cpfmsgf-db2.sql Y

qre-iseries-cpfmsgf-hsqldb.sql Y

qre-iseries-cpfmsgf-mssql.sql Y

qre-iseries-cpfmsgf-mysql.sql Y

qre-iseries-cpfmsgf-oracle.sql Y

Cpf3220.java – sửa 1 dấu chấm Y

Cpf6ac7 – sửa lại comment Y

Cpfa089 – sửa lại comment Y

qre-iseries-cpfmsgf-data.sql (add) Y

**thêm data.xml**

qre-test-dataqueue-init-db.xml (edit) Y

qre-test-infrastructure-init-db-ovrdb.xml (edit) Y

qre-test-infrastructure-init-db.xml (edit) Y

qre-test-messaging-init-db.xml (edit) Y

qre-test-userspace-init-db.xml (edit) Y - 15

qre-test-infrastructure-init-db.xml (edit – nằm trong qre2-iseries-cobol) Y

qre-test-infrastructure-init-db.xml (edit – nằm trong qre2-iseries-rpg) Y

qre-web-infrastructure-init-db.xml (edit – nằm trong honda-poc) Y - 18

**làm cho oracle:**

qre-iseries-cpfmsgf-data.sql (add – thêm vào as01) Y

qre-iseries-cpfmsgf-oracle.sql ( edit – nằm trong as01) Y

qre-iseries-cpfmsgf-data.sql (add – thêm vào daito-pilot) Y

qre-iseries-cpfmsgf-oracle.sql ( edit – nằm trong daito-pilot) Y

app-infrastructure-init-db.xml (edit – nằm trong as01) Y

app-infrastructure-init-db.xml (edit – nằm trong as01-iteration 2) Y

qre-scripts.txt (edit – nằm trong as01) Y

qre-scripts.txt (edit – nằm trong daito-common) Y

qre-scripts.txt (edit – nằm trong honda-common) Y

qre-scripts.txt (edit – nằm trong honda-poc) Y

qre-scripts.txt (edit – nằm trong daito-pilot) Y - 29

**làm cho db2:**

qre-iseries-cpfmsgf-data.sql (add -nằm trong as01) Y

qre-iseries-cpfmsgf-DB2.sql (edit -nằm trong as01) Y

qre-iseries-cpfmsgf-data.sql (add -nằm trong daito-common) Y

qre-iseries-cpfmsgf-DB2.sql (edit -nằm trong daito-common) Y

qre-iseries-cpfmsgf-data.sql (add -nằm trong honda-common) Y

qre-iseries-cpfmsgf-DB2.sql (edit -nằm trong honda -common) Y – 35

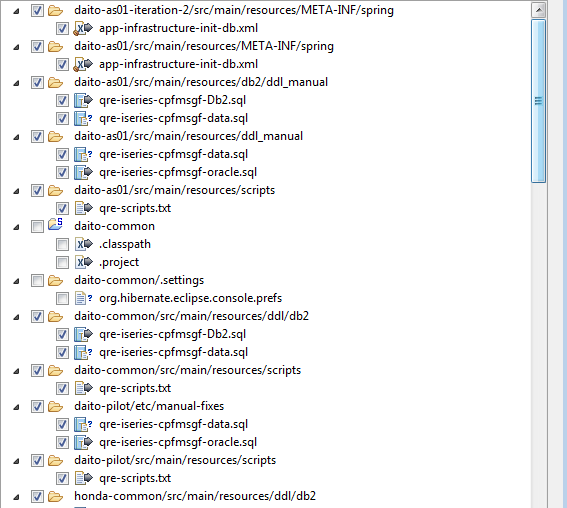
Sửa các file test error:

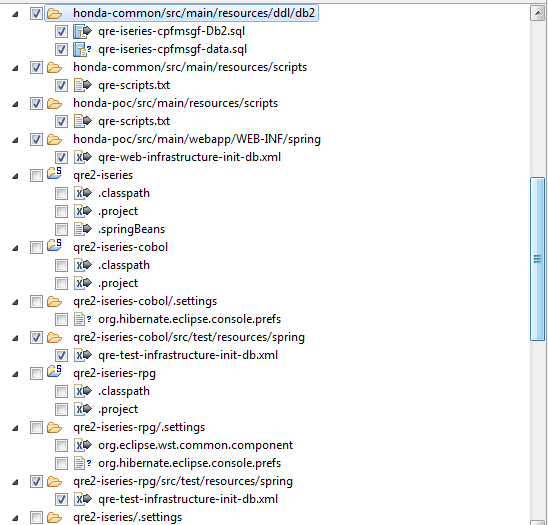
RetrieveMessageImplTest – sửa 245 thành 237

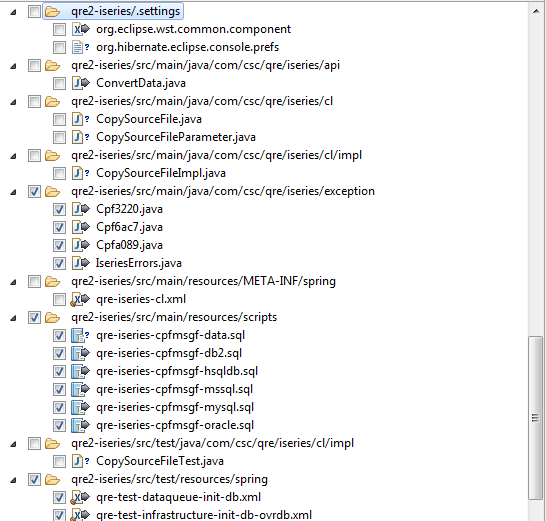
SendNonProgramMessageImplTest – sửa 32 thành 31

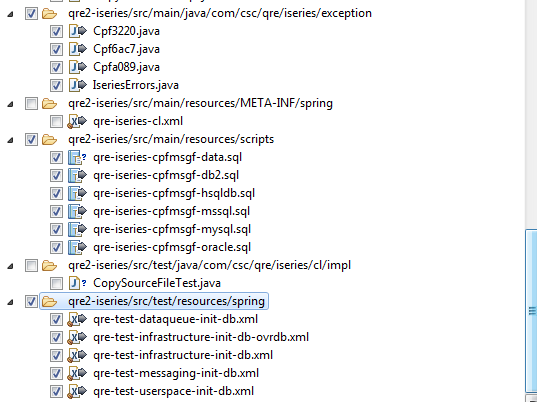
DBMessageRepositoryTest – bỏ khoảng trắng

**Tổng: 35 file**









## 2. Các ticket liên quan:

## 3. Tìm hiểu

[DAITO][ITER2][FT] Merge Error Message SQL and ensure it's update to date with Error Exceptions

1. merge DDL for loading error messages  
   under qre2-iseries/src/main/resources/scripts, there are a few versions of qre-iseries-cpfmsgf-\*.sql.   
   each for one DB. Personally, I don't feel it's necessary have data load included in those individual scripts (Cá nhân, tao cảm thấy nó không cần thiết để load data mà bao gồm những script riêng biệt)  
   Compare all the scripts and split and merge the scripts (use oracle and hsqldb as the major input)
   * each db has a script for creating tables  
     named as qre-iseriesqcpfmsgf-ddl-<db>.sql
   * one single ddl for load data, named as qre-iseries-qcpfmsg-data.sql
   * revise configuration of all projects to include the new script for test resources  
     Need to cover at least daito projects n Honda PoC project.   
     search where includes qre-iseries-qcpfmsg-hsqldb.sql, which should have this new script added.
2. review and add to qre-iseries-qcpfmsg-data.sql for message definitions for classes under com.csc.qre.iseries.exception  
   only for Fmt9001, C\*, Ect0110, Rpg0202, excluding XXX0000 or XXXnn00

## 4. Do it

Các file thỏa tiêu chí under **qre2-iseries/src/main/resources/scripts** , có dạng **qre-iseries-cpfmsgf-\*.sql**

1. qre-iseries-cpfmsgf-db2.sql

2. qre-iseries-cpfmsgf-hsqldb.sql

3. qre-iseries-cpfmsgf-mssql.sql

4. qre-iseries-cpfmsgf-mysql.sql

5. qre-iseries-cpfmsgf-oracle.sql

Phần 1 chưa hiểu yêu cầu. Thôi làm phần 2

### Phần 2:

Các exception chưa thêm vào file xml:

1. Cpd0071

Parameter &2 required.

2. Cpd2967

Position error occurred opening member &3.

3. Cpd8063

Distribution to user &1 &2 not done; user ID is not valid.

4. Cpf0610 – cái này thiếu 1 số chỗ

File &1 in &2 not available.

5. Cpf0864 – dư 1 cái trong hsqldb

6. Cpf1023 – chỉnh lại cho đúng vị trí

7. Cpf1023 – bị duplicate phần parameter

8. Cpf1028

&1 not valid for parameter SYSVAL.

9. Cpf1336 – thiếu hết 1 file xml (mysql)

10. Cpf180c – thiếu trong 2 file xml (mysql và mssql)

11. Cpf1860

Value &1 in list not valid.

12. Cpf1861

Length of the receiver variable not valid.

13. Cpf1862

Number of values to retrieve not valid.

14. Cpf2109

NEWOBJ must be \*SAME when OBJ parameter is \*ALL or generic name.

15. Cpf2130 – thiếu trong 2 file xml

16. Cpf2155

\*LIBL cannot be specified for FROMLIB

17. Cpf2185

TOLIB, TOASPDEV, or NEWOBJ parameter not correct.

18. Cpf2186

Object &1 cannot be created into library &2.

19. Cpf2410 – thiếu 2 file xml

20. Cpf2415 – thiếu 2 file xml

21. Cpf241a – thiếu phần parameter

22. Cpf24a7 – thiếu 2 file xml

23. Cpf24ab – thiếu 2 file xml

24. Cpf2816

File &1 in &2 not copied because of error.

25. Cpf2869 – thiếu 2 file xml

26. Cpf2985

Source sequence numbers exceeded maximum value allowed. (G C)

27. Cpf3012 – thiếu 2 file xml

File &1 in library &2 not found.

28. Cpf3019 – thiếu 2 file xml

29. Cpf3049 – thiếu 2 file xml

30. Cpf3130 – thiếu 2 file xml, sai &2 mà insert &1

Member &2 already in use.

31. Cpf3133

File &1 in library &3 contains no members.

32. Cpf3135

Access path for member &2 already in use.

33. Cpf3141 – thiếu 2 file xml, sai &2 thành &1

Member &2 not found.

34. Cpf3142

File &1 in library &3 not found

35. Cpf3156

File &1 in library &3 in use.

36. Cpf3178

Member &3 not renamed to &4.

37. Cpf3204

Cannot find object needed for file &1 in &2.

38. Cpf3213

Members for file &1 more than maximum allowed.

39. Cpf3220

Cannot do operation on file &1 in &2.

40. Cpf3275

Member &3 file &1 in &2 not found.

41. Cpf327f

Operation not successful for file &1 in library &2.

42. Cpf3288

Member &3 file &1 in &2 not changed.

43. Cpf3311 – thiếu 2 file xml, sai &

Copy request failed for file &6 in &7.

44. Cpf3335 – thiếu 2 file xml và sai &

45. Cpf3336 – thiếu 2 file xml và sai &

Job &5/&4/&3 no longer in the system.

46. Cpf3340 – thiếu 2 file xml và sai &

More than one file with specified name found in job &5/&4/&3.

47. Cpf3344 – thiếu 2 file xml và sai &

File &1 number &8 no longer in the system.

48. Cpf34a4 – dư 1 cái trong 1 file xml

49. Cpf3c19

Error occurred with receiver variable specified.

50. Cpf3c21

51. Cpf3c42

52. Cpf3c43

53. Cpf3c44

54. Cpf3c58 – thiếu 2 file xml

55. Cpf3cf1

56. Cpf4125 – thiếu sai lung tung

57. Cpf4128

58. Cpf4137

59. Cpf5812

60. Cpf6ac7

61. Cpf7301

62. Cpf7306

63. Cpf7310

64. Cpf7360

65. Cpf8067

66. Cpf8351

67. Cpf9006

68. Cpf9803

69. Cpf9806

70. Cpf9812

71. Cpf9815

72. Cpf9818 – sai tè le

73. Cpf9841

74. Cpfa089 – thiếu 2 file xml

Pattern not allowed in path name ?

75. Cpfa0a0

76. Cpfa0a7

77. Cpd0071

79. Cpd2967

80. Cpd8063

### Phần 1:

Các bước add data vào table ví dụ:

**1. file bean**

<jdbc:initialize-database data-source="applicationDataSource" ignore-failures="DROPS">

<jdbc:script location="classpath:scripts/qre-schema-hsqldb.sql" />

<jdbc:script location="classpath:scripts/qre-iseries-cpfmsgf-hsqldb.sql" />

<jdbc:script location="classpath:scripts/qre-resourcelock-ddl-hsqldb.sql" />

2<jdbc:script location="classpath:scripts/qre-iseries-dataqueue-ddl-hsqldb.sql" />

3<jdbc:script location="classpath:scripts/qre-iseries-member-control-ddl-hsqldb.sql" />

4<jdbc:script location="classpath:scripts/qre-iseries-member-attributes-ddl-hsqldb.sql" />

5<jdbc:script location="classpath:scripts/qre-test-iseries-member-control\_hsqldb.sql" />

6<jdbc:script location="classpath:scripts/qre-test-iseries-member-attributes\_hsqldb.sql" />

</jdbc:initialize-database>

</beans>

Phần test:

**2. qre-iseries-dataqueue-ddl-hsqldb.sql: cái này để tạo table**

**DROP** **TABLE** DATAQUEUE IF **EXISTS**;

**CREATE** **TABLE** DATAQUEUE (

ID **INTEGER** GENERATED **BY** **DEFAULT** **AS** IDENTITY (START **WITH** 1 INCREMENT **BY** 1) **NOT** **NULL**,

NAME **CHAR**(10) **NOT** **NULL**,

LIBRARY **CHAR**(10) **NOT** **NULL**,

MAXIMUM\_LENGTH **INTEGER** **NOT** **NULL**,

SEQUENCE **CHAR**(10),

KEY\_LENGTH **INTEGER**,

DESCRIPTION **VARCHAR**(50) **NOT** **NULL**,

**PRIMARY** **KEY**(ID));

GRANT **all** **on** DATAQUEUE to sa;

**3. qre-iseries-member-control-ddl-hsqldb.sql: cái này để tạo table luôn**

**DROP** **TABLE** member\_control IF **EXISTS**;

**CREATE** **TABLE** member\_control (

CREATOR **varchar**(10) **NOT** **NULL**,

TABLE\_NAME **varchar**(20) **NOT** **NULL**,

MEMBER\_NAME **varchar**(10) **NOT** **NULL**,

CONTENTS\_TYPE **varchar**(10) **NOT** **NULL**,

DESCRIPTION **varchar**(50) **NOT** **NULL**,

EXPIRY\_DATE **date** **DEFAULT** **NULL**,

CREATION\_DATE **date** **DEFAULT** **NULL**,

MEMBER\_ID **varchar**(40) **DEFAULT** **NULL**,

MEMBER\_TYPE **char**(1) **DEFAULT** **NULL**,

**PRIMARY** **KEY** (CREATOR,TABLE\_NAME,MEMBER\_NAME)

) ;

GRANT **all** **on** member\_control to sa;

**DROP** **TABLE** MEMBER\_COMPONENT IF **EXISTS**;

**CREATE** **TABLE** MEMBER\_COMPONENT (

MEMBER\_ID **VARCHAR**(40) **NOT** **NULL**,

PT\_CREATOR **VARCHAR**(10) **NOT** **NULL** ,

PT\_TABLE\_NAME **VARCHAR**(20) **NOT** **NULL** ,

PT\_MEMBER **VARCHAR**(10) **NOT** **NULL**,

**PRIMARY** **KEY** (MEMBER\_ID)

);

GRANT **all** **on** MEMBER\_COMPONENT to sa;

--

-- Definition of table member\_table

--

**DROP** **TABLE** member\_table IF **EXISTS**;

**CREATE** **TABLE** member\_table (

CREATOR **varchar**(64) **NOT** **NULL**,

TABLE\_NAME **varchar**(64) **NOT** **NULL**,

MAX\_MEMBERS **INTEGER** **NOT** **NULL**,

**PRIMARY** **KEY** (CREATOR,TABLE\_NAME)

) ;

GRANT **all** **on** member\_table to sa;

**4. qre-iseries-member-attributes-ddl-hsqldb.sql: cái này vẫn là tạo table**

--

-- Definition of table QRE\_FILE\_ATTRS

--

**DROP** **TABLE** QRE\_FILE\_ATTRS IF **EXISTS**;

**CREATE** **TABLE** QRE\_FILE\_ATTRS (

LIBRARY **VARCHAR**(20) **NOT** **NULL**,

NAME **VARCHAR**(20) **NOT** **NULL**,

FILE\_TYPE **INTEGER** **DEFAULT** 0,

MAX\_MEMBERS **INTEGER** **DEFAULT** 1,

**PRIMARY** **KEY** (LIBRARY, NAME)

) ;

GRANT **all** **on** QRE\_FILE\_ATTRS to sa;

--

-- Definition of table QRE\_FILE\_MBRS

--

**DROP** **TABLE** QRE\_FILE\_MBRS IF **EXISTS**;

**CREATE** **TABLE** QRE\_FILE\_MBRS (

LIBRARY **VARCHAR**(20) **NOT** **NULL**,

NAME **VARCHAR**(20) **NOT** **NULL**,

MEMBER\_NAME **VARCHAR**(20) **NOT** **NULL**,

DESCRIPTION **VARCHAR**(50),

CREATION\_DATE **date** **DEFAULT** CURRENT\_TIMESTAMP,

**PRIMARY** **KEY** (LIBRARY,NAME,MEMBER\_NAME)

) ;

GRANT **all** **on** QRE\_FILE\_MBRS to sa;

**ALTER** **TABLE** QRE\_FILE\_MBRS

**ADD** **CONSTRAINT** FK\_QRE\_FILE\_MBRS

**FOREIGN** **KEY** (LIBRARY, NAME)

**REFERENCES** QRE\_FILE\_ATTRS (LIBRARY, NAME)

**5. qre-test-iseries-member-control\_hsqldb.sql: cái này insert data**

**SET** **SCHEMA** QRE;

**INSERT** **INTO** member\_control (CREATOR,TABLE\_NAME,MEMBER\_NAME,CONTENTS\_TYPE,DESCRIPTION,EXPIRY\_DATE,CREATION\_DATE,MEMBER\_ID,MEMBER\_TYPE) **VALUES** ('ISERIESPOC','ATNMPF','TESTFILEM3',' ','ForTesting','2016-04-06 14:30:27','2016-04-06 14:30:27','iseriespocAtnmpfTestFileM2','P');

**INSERT** **INTO** member\_control (CREATOR,TABLE\_NAME,MEMBER\_NAME,CONTENTS\_TYPE,DESCRIPTION,EXPIRY\_DATE,CREATION\_DATE,MEMBER\_ID,MEMBER\_TYPE) **VALUES** ('ISERIESPOC','ATNMPF','TESTFILEM2',' ','ForTesting','2016-04-06 14:30:27','2016-04-06 14:30:27','iseriespocAtnmpfTestFileM2','P');

**INSERT** **INTO** member\_control (CREATOR,TABLE\_NAME,MEMBER\_NAME,CONTENTS\_TYPE,DESCRIPTION,EXPIRY\_DATE,CREATION\_DATE,MEMBER\_ID,MEMBER\_TYPE) **VALUES** ('ISERIESPOC','ATNMPF1','TESTFILEM2',' ','ForTesting','2016-04-06 14:30:27','2016-04-06 14:30:27','iseriespocAtnmpfTestFileM2','P');

**INSERT** **INTO** member\_control (CREATOR,TABLE\_NAME,MEMBER\_NAME,CONTENTS\_TYPE,DESCRIPTION,EXPIRY\_DATE,CREATION\_DATE,MEMBER\_ID,MEMBER\_TYPE) **VALUES** ('ISERIESPOC','ATNMPF2','TESTFILEM2',' ','ForTesting','2016-04-06 14:30:27','2016-04-06 14:30:27','iseriespocAtnmpfTestFileM2','P');

**INSERT** **INTO** member\_control (CREATOR,TABLE\_NAME,MEMBER\_NAME,CONTENTS\_TYPE,DESCRIPTION,EXPIRY\_DATE,CREATION\_DATE,MEMBER\_ID,MEMBER\_TYPE) **VALUES** ('ISERIESPOC','ATNMPF3','TESTFILEM2',' ','ForTesting','2016-04-06 14:30:27','2016-04-06 14:30:27','iseriespocAtnmpfTestFileM2','P');

**INSERT** **INTO** member\_control (CREATOR,TABLE\_NAME,MEMBER\_NAME,CONTENTS\_TYPE,DESCRIPTION,EXPIRY\_DATE,CREATION\_DATE,MEMBER\_ID,MEMBER\_TYPE) **VALUES** ('ISERIESPOC','ATNMPF3','TESTFILEM3',' ','ForTesting','2016-04-06 14:30:27','2016-04-06 14:30:27','iseriespocAtnmpfTestFileM2','P');

**INSERT** **INTO** member\_control (CREATOR,TABLE\_NAME,MEMBER\_NAME,CONTENTS\_TYPE,DESCRIPTION,EXPIRY\_DATE,CREATION\_DATE,MEMBER\_ID,MEMBER\_TYPE) **VALUES** ('ISERIESPOC','ATNM','TESTFILEM2',' ','ForTesting','2016-04-06 14:30:27','2016-04-06 14:30:27','iseriespocAtnmTestFileM2','P');

**INSERT** **INTO** member\_control (CREATOR,TABLE\_NAME,MEMBER\_NAME,CONTENTS\_TYPE,DESCRIPTION,EXPIRY\_DATE,CREATION\_DATE,MEMBER\_ID,MEMBER\_TYPE) **VALUES** ('iseriespoc','ATNM','MEMBERA',' ','ForTesting','2016-04-06 14:30:27','2016-04-06 14:30:27','iseriespocAtnmTestFileM2','P');

**INSERT** **INTO** member\_control (CREATOR,TABLE\_NAME,MEMBER\_NAME,CONTENTS\_TYPE,DESCRIPTION,EXPIRY\_DATE,CREATION\_DATE,MEMBER\_ID,MEMBER\_TYPE) **VALUES** ('iseriespoc','ATNM','MEMBERN',' ','ForTesting','2016-04-06 14:30:27','2016-04-06 14:30:27','iseriespocAtnmTestFileM2','P');

**INSERT** **INTO** member\_table (CREATOR,TABLE\_NAME,MAX\_MEMBERS) **VALUES** ('ISERIESPOC','ATNMPF',32767);

**INSERT** **INTO** member\_table (CREATOR,TABLE\_NAME,MAX\_MEMBERS) **VALUES** ('ISERIESPOC','ATNMPF1',32767);

**INSERT** **INTO** member\_table (CREATOR,TABLE\_NAME,MAX\_MEMBERS) **VALUES** ('ISERIESPOC','ATNM',32767);

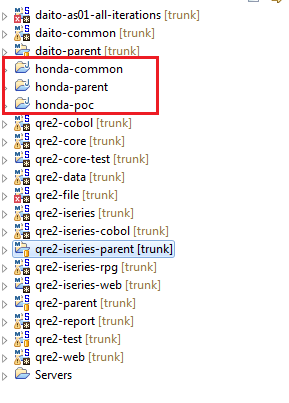
**INSERT** **INTO** member\_table (CREATOR,TABLE\_NAME,MAX\_MEMBERS) **VALUES** ('iseriespoc','ATNM',32767);

**INSERT** **INTO** member\_table (CREATOR,TABLE\_NAME,MAX\_MEMBERS) **VALUES** ('ISERIESPOC','ATNMPF2',0);

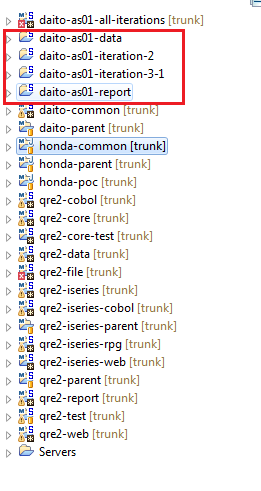
**INSERT** **INTO** member\_table (CREATOR,TABLE\_NAME,MAX\_MEMBERS) **VALUES** ('ISERIESPOC','ATNMPF3',32767);

**INSERT** **INTO** member\_table (CREATOR,TABLE\_NAME,MAX\_MEMBERS) **VALUES** ('ISERIESPOC2','ATNMPFNEW',0);

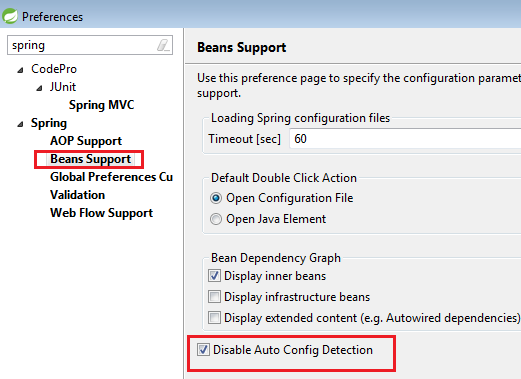
### Project được thêm bởi honda-poc.psf

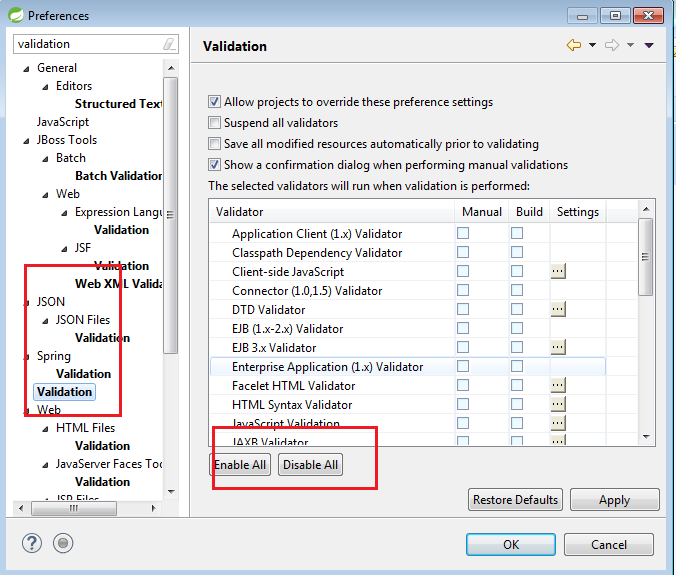


### Project được thêm bởi daito-as01-iterations.psf



Project nặng quá phải disable cái spring validation (có 2 cái)





# 32. Làm về exception

Muốn thêm 1 exception thì

### Step 1: thêm vào câu message vào 5 file sql sau:

1. qre-iseries-cpfmsgf-db2.sql

2. qre-iseries-cpfmsgf-hsqldb.sql

3. qre-iseries-cpfmsgf-mssql.sql

4. qre-iseries-cpfmsgf-mysql.sql

5. qre-iseries-cpfmsgf-oracle.sql

### Step 2:

Thêm vào file **IseriesErrors.java**

# 33. QTE-5757 support setText and setLevelCheck (closed)

## 1. Các file đã sửa:

ChangePhysicalFileParameter (edit)

ChangePhysicalFileTest (edit)

## 2. Các ticket liên quan:

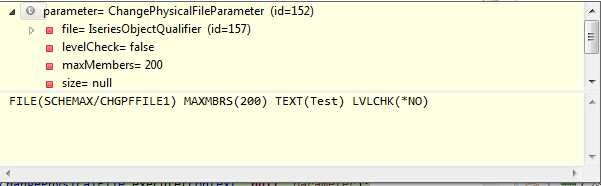
## 3. Tìm hiểu:

**Sql nằm trong MemberControlDaoImpl:**

SELECT MAX\_MEMBERS FROM QRE.QRE\_FILE\_ATTRS WHERE LIBRARY = ? AND NAME = ?

isSingleMemberTable nghĩa là table không có data

parameter:



**Sql nằm trong ChangeLogicalFileMember:**

UPDATE QRE.QRE\_FILE\_MBRS SET DESCRIPTION = DESCRIPTION TEST1 WHERE LIBRARY = SCHEMAX AND NAME = FILEXV1 AND MEMBER\_NAME = FILEXV1

# 34. Maven

## Cách build maven bằng console command

<https://thinhnv2.wordpress.com/2013/04/14/maven-can-ban/>

Hinh nhu maven chi can download khong can setp, them path vao environment, hien tai cua minh la 3.5.0

# 35. QTE-5685 Manual rewrite FICONV program

## 1. Các file đã sửa

Fincov.java (edit)

NumberConstants (edit)

com.daito.common.dto (add) (package)

Jefibml1Dto (add)

Jefibml2Dto (add)

JefibmpDto (add)

com.daito.common.entity (add) (package)

AbstractJefibmpDo (add)

Jefibml1Do (add)

Jefibml2Do (add)

JefibmpDo (add)

dto\_mapping.properties (add)

app-annotation-processors.properties (add)

## 2. Các ticket liên quan:

5210: cũng manual rewrite 3 programs mà sao tìm submit code k thấy

## 3. Tìm hiểu

## 4. Do it

C \*ENTRY PLIST

C PARM PARM01 6 変換方法

C PARM PARM02 256 変換前データ

C PARM PARM03 256 変換後データ

C PARM PARM04 256 エラーデータ

C PARM PARM05 3 0 データ長

C PARM PARM06 2 エラー文字(X'4040')

C PARM PARM07 1 制御コード

C PARM PARM08 3 0 リターンコード

**Transform to:**

**private** FixedLengthString parm01 = *createFixedLengthString*(6);

**private** FixedLengthString parm02 = *createFixedLengthString*(256);

**private** FixedLengthString parm03 = *createFixedLengthString*(256);

**private** FixedLengthString parm04 = *createFixedLengthString*(256);

**private** PackedDecimal parm05 = *createPackedDecimal*(3, 0);

**private** FixedLengthString parm06 = *createFixedLengthString*(2);

**private** FixedLengthString parm07 = *createFixedLengthString*(1);

**private** PackedDecimal parm08 = *createPackedDecimal*(3, 0);

-----------------------------

C JEFCOD OREQ X'0A41'

**Transform to:**

**private** **final** HexConstant HEXCONSTANTFORHEX2 = HexConstant("0x0A41"); có gì đó sai sai khi khi báo constant như thế này

mình chuyển sang:

**private** **final** **int** HEXCONSTANTFORHEX2 = HexConstant.***HEX\_A41***;

----------------------------------------------

### DCPRTN3 có:

D IGCARA DS INZ

D MODE 1 6

D BEFR 7 262

D AFTR 263 518

D ERRR 519 774

D LENGTH 775 777P 0

D IGCCOD 778 779

D CTLCOD 780 780

D RTNCOD 781 783P 0

**Transform to:**

@SuppressWarnings("serial")

**public** **final** FixedLengthString igcara = **new** DataStructure<FixedLengthString>(783) {

@Override

**public** **void** clear() {

**super**.clear();

*clearArray*(befr);

*clearArray*(aftr);

*clearArray*(errr);

}

};

**private** FixedLengthString mode = *createFixedLengthString*(6).isAPartOf(igcara, 0);

**private** FixedLengthString[] befr = *initArray*(**new** FixedLengthString[129], 2, igcara, 6);

**private** FixedLengthString[] aftr = *initArray*(**new** FixedLengthString[129], 2, igcara, 262);

**private** FixedLengthString[] errr = *initArray*(**new** FixedLengthString[129], 2, igcara, 518);

**private** PackedDecimal length = *createPackedDecimal*(5, 0).isAPartOf(igcara, 774);

**private** FixedLengthString igccod = *createFixedLengthString*(2).isAPartOf(igcara, 777);

**private** FixedLengthString ctlcod = *createFixedLengthString*(1).isAPartOf(igcara, 779);

**private** PackedDecimal rtncod = *createPackedDecimal*(5, 0).isAPartOf(igcara, 780);

### DCPRTN5 có khá nhiều hex:

47 - C N55JEFCOD IFEQ X'4040'

48 - C JEFCOD OREQ X'0A41'

49 - C JEFCOD OREQ X'0A42'

Tuy nhiên X’4040 là được moveleft ở dòng 50:

50 - C MOVEL X'4040' AFTR(K)

73 - C BEFR(K) IFGE X'80A1'

74 - C BEFR(K) ANDLE X'A0FE'

75 - C HEXCD IFNE X'A0'

76 - C HEXCD ANDNE X'FF'

### Lỗi indArea.setOn

indArea.setOn(***INDLR***);

Tham khảo Ebg100r extend từ AbstractRpgCodeModel và fix được

### Lỗi GotoLabels label

GotoLabels label= GotoLabels.***GOTO\_START***;

Tham khảo ChashTgb3300j và fix được

### Lỗi set.(ZERO)

Thêm RpgConstant (tham khảo Aja850r) là fix được.

### Lỗi *isEQ*

Thêm ConditionFunction, tham khảo ChashTgb3300j và fix được.

### Lỗi GotoLabels.***pass***

Chỉnh sửa lại enum GotoLabels (tham khảo) Lxs02cal và fix được.

### Lỗi compute, divide

Thêm **import** **static** com.csc.qre.cobol.lang.CobolArithmeticFunctions.\*;

### Lỗi GotoException

Thêm cái support.GotoException (tham khảo ChashTgb3300j) và fix được

### Lỗi OpenMode.***INPUT***

Thêm OpenMode từ qre.file… (tham khảo Tgb3330) và fix được

## Lỗi setValues

Thêm cái support. ProgramUtils.\* (tham khảo ChashTgb3300j) và fix được

### Không biết có cần tạo cái dto\_mapping.properties không?

Mình tạo luôn rồi

# 36. QTE-5815

## 1. các file đã sửa

### Env-local.properties:

### env-oracle.properties:

### app-data-config.properties

app-user-oracle.sql (tạo thêm 2 schema qtemp và HXXTFR)

Thực tế:

1. CreateDDMFileParameter

## 2. Các ticket liên quan:

Mẫn được phân công làm mấy cái này khá nhiều

## 3. Tìm hiểu

Config để chạy được Oracle:

env-local.properties

env-local.properties

đổi daito15 thành daitoDB

## 4. Do it

Các câu lệnh sql under src/resources/ddl:

**CREATE** SEQUENCE AAUGZP\_SEQ START **WITH** 1;

COMMENT **ON** **COLUMN** AAUGZP.AAUJ2D **IS** '対象年月';

**CREATE** **TABLE** AAUGZP (

UNIQUE\_NUMBER NUMBER(19,0) **NOT** **NULL**,

AAUJ1B NVARCHAR2(1) **DEFAULT** ' ',

**CREATE** **UNIQUE** **INDEX** AAUGZP\_UNIQUE\_IDX **ON** AAUGZP (AAUJ1B, AAUJ2D, AAUJ3C);

**ALTER** **TABLE** AAUGZP PARALLEL 8;

2. Tìm hiểu đợt 2:

Gặp lỗi ngay chỗ này

**if** (clCommand == **null**) {

**throw** **new** QreException("Command implementation is not configured for " + parameter.toString());

}

Có thể là chưa tạo file bean.

Câu lệnh sql:

SELECT UNIQUE\_NUMBER,ESF01C,ESF02M,ESF03M,ESF04C,ESF05C,ESF06C,ESF07C,ESF08A,ESF09A,ESF10S,ESF11S,ESF12S,ESF13D,ESF14D,ESF15H,ESF16D,ESF17D,ESF18H,ESF19S,ESF20C,ESF21C FROM \*LIBL.ESFCTP ORDER BY ESF01C ASC ,UNIQUE\_NUMBER ASC

### 4.1 Tạo cái OracleDialect

SELECT COLUMN\_NAME, COLUMN\_DEFAULT, DATA\_TYPE, IS\_NULLABLE, CASE WHEN COLUMN\_DEFAULT IS NOT NULL THEN COLUMN\_DEFAULT WHEN IS\_NULLABLE = 'YES' AND COLUMN\_DEFAULT IS NULL THEN NULL WHEN DATA\_TYPE = 'BIGINT' OR DATA\_TYPE = 'DECIMAL' OR DATA\_TYPE = 'DOUBLE' OR DATA\_TYPE = 'FLOAT' THEN '0' WHEN DATA\_TYPE = 'INTEGER' OR DATA\_TYPE = 'NUMERIC' OR DATA\_TYPE = 'SMALLINT' OR DATA\_TYPE = 'TINYINT' THEN '0' WHEN DATA\_TYPE = 'DATE' THEN ''||CURRENT\_DATE||'' WHEN DATA\_TYPE = 'TIME' THEN ''||CURRENT\_TIME||'' WHEN DATA\_TYPE = 'TIMESTAMP' THEN ''||CURRENT\_TIMESTAMP||'' ELSE '' END AS DEFAULT\_VALUE FROM INFORMATION\_SCHEMA.COLUMNS WHERE TABLE\_NAME = 'ATNMPF1\_ATNMPF1' AND TABLE\_SCHEMA = 'ISERIESPOC'

## Đệt mợ cái này, làm mãi k xong là sao:

com.daito.as01.hxx.pf.a2o.tables.dto.Xmw4kpDto

A2obepDto

Class com.daito.as01.hxx.pf.xmw.tables.dto.Xmw4kpDto

Class com.daito.as01.hxx.pf.xmw.tables.dto.Xmw4kpDto

Class com.daito.as01.hxx.pf.xmw.tables.entity.Xmw4kpDo

class com.daito.as01.hxx.pf.a2o.tables.dto.A2obepDto

XMW4KP=com.daito.as01.hxx.pf.xmw.tables.dto.Xmw4kpDto

# 37. Java SE 8 Fundamentals: Introduction to Java

# 38. Drop table khi gặp lỗi 14452

### ORA-14452: attempt to create, alter or drop an index on temporary table already in use

**Cause:** Tried to drop a GTT Global temporary table but I got an error 'ORA-14452: attempt to create, alter or drop an index on temporary table already in use'.  
  
Here problem with not completed the transaction in any session and tried to drop the table then we will get this error. For session specifict GTT, one created with ON COMMIT PRESERVE ROWS, the session needs to truncate the GTT.  
  
**Action:** Followed below steps to DROP GTT  
  
1) Tried to drop a GTT.  
  
SQL> drop table REGISTR.ENT\_GTT;  
  
drop table REGISTR.ENT\_GTT  
  
ORA-14452: attempt to create, alter or drop an index on temporary table already in use  
  
2) Using below query, first any active session which is locking this object and will find SID of that locked object.  
  
select  \*   from  v$lock   where id1 = (select  object\_id from  dba\_objects   
where owner = 'REGISTR' and object\_name = 'ENT\_GTT');  
  
SID is 447.  
  
3) Using that SID I have killed that session.  
  
select inst\_id,'alter system kill session ''' || SID || ',' || SERIAL# || ''' IMMEDIATE;'   
from gv$session gg where gg.SID =447;  
  
SQL> alter system kill session '447,4045' IMMEDIATE;  
  
System altered  
  
4) Now I dropped mentioned global temporary table.  
  
SQL> drop table REGISTR.ENT\_GTT;  
  
Table dropped  
  
SQL>  
  
Thats It..Hope this will help you...:)  
  
Best Regards,

# 39. QTE-5858

## 1. Các file đã sửa

## 2. Các ticket liên quan

5539

Ticket của Tiến: 5689

Ticket cần so sánh sumary: 5477

CCH010R. Program này khá giống mình cần làm – 5210, 5696

## 3. Tìm hiểu

## 4. Do it

# 40. Tạo user(schema) oracle

Dựa vào app-user-oracle.sql

# 41. Oracle 12:

## Uninstall:

1. Stop hết các service bằng cách mở task manager hoặc :

1. Open the Windows Services utility: From the Start menu, choose Programs, then Administrative Tools, then Services.
2. If any Oracle services (names begin with Oracle or Ora) exist and have the status Started, then select each of the services, and click Stop.
3. Stop the Microsoft Distributed Transaction Coordinator service.
4. Exit Services.
5. Restart the computer.

Phải chạy cả 2 để đảm bảo đã tắt hết các service.

2. Vào nơi cài đặt chạy quyền admin deinstall, nơi cài đặt thường có chữ app đầu tiên:

Máy 77: D:\app\cscv\product\11.2.0\dbhome\_1\deinstall

Máy mình: D:\app\CSC\product\12.1.0\dbhome\_1\deinstall

Hiện lên cmd, boostraping gì đó, dòng đầu tiên kiu nhập enter luôn, dòng thứ 2 kiu nhập có thteer là ORCL (hoặc DAITODB) chưa biết

Sau đó hỏi modify the details: n

Sau đó hỏi Continue: y

3.

Run regedit.exe and delete the HKEY\_LOCAL\_MACHINE/SOFTWARE/Oracle key. This contains registry entires for all Oracle products.

4.

If you are running 64-bit Windows, you should also delete the HKEY\_LOCAL\_MACHINE/SOFTWARE/Wow6432Node/Oracle key if it exists.

5.

 If you are running 64-bit Windows, you should also delete the HKEY\_LOCAL\_MACHINE/SOFTWARE/Wow6432Node/Oracle key if it exists.

 Delete any references to Oracle services left behind in the following part of the registry (HKEY\_LOCAL\_MACHINE/SYSTEM/CurrentControlSet/Services/Ora\*). It should be pretty obvious which ones relate to Oracle.

 Reboot your machine.

 Delete the "C:\Oracle" directory, or whatever directory is your ORACLE\_BASE.

 Delete the "C:\Program Files\Oracle" directory.

 If you are running 64-bit Wiindows, you should also delete the "C:\Program Files (x86)\Oracle" directory.

 Remove any Oracle-related subdirectories from the "C:\ProgramData\Microsoft\Windows\Start Menu\Programs\" directory.

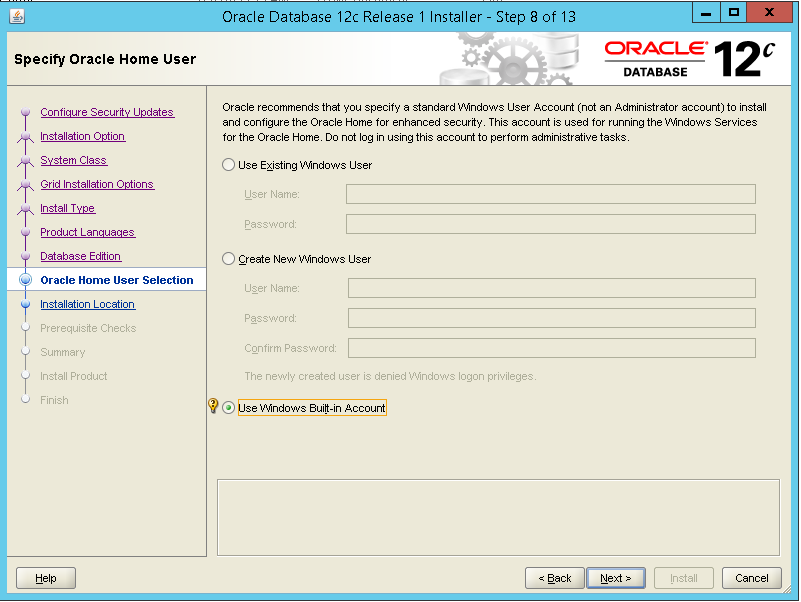
 Empty the contents of your "C:\temp" directory.

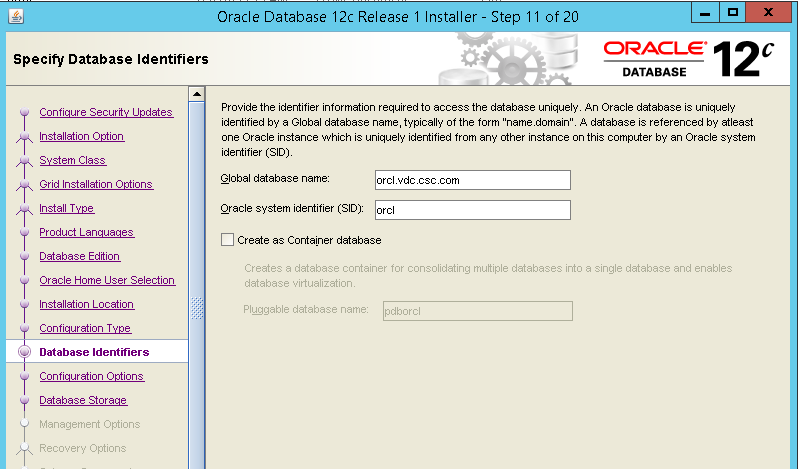
 Empty your recycle bin.

## Install

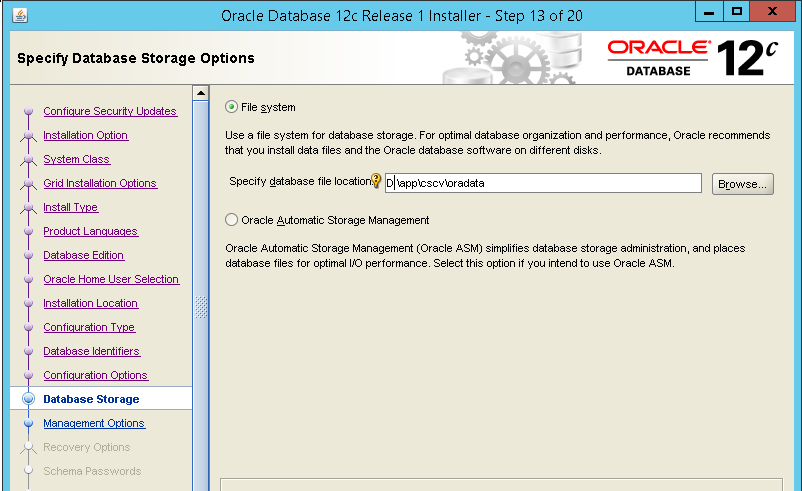
Chạy theo step: C:\Users\tnguyen443\Desktop\Daito\anh Huyen\oracle11.2\Install\_Steps

Step home user:

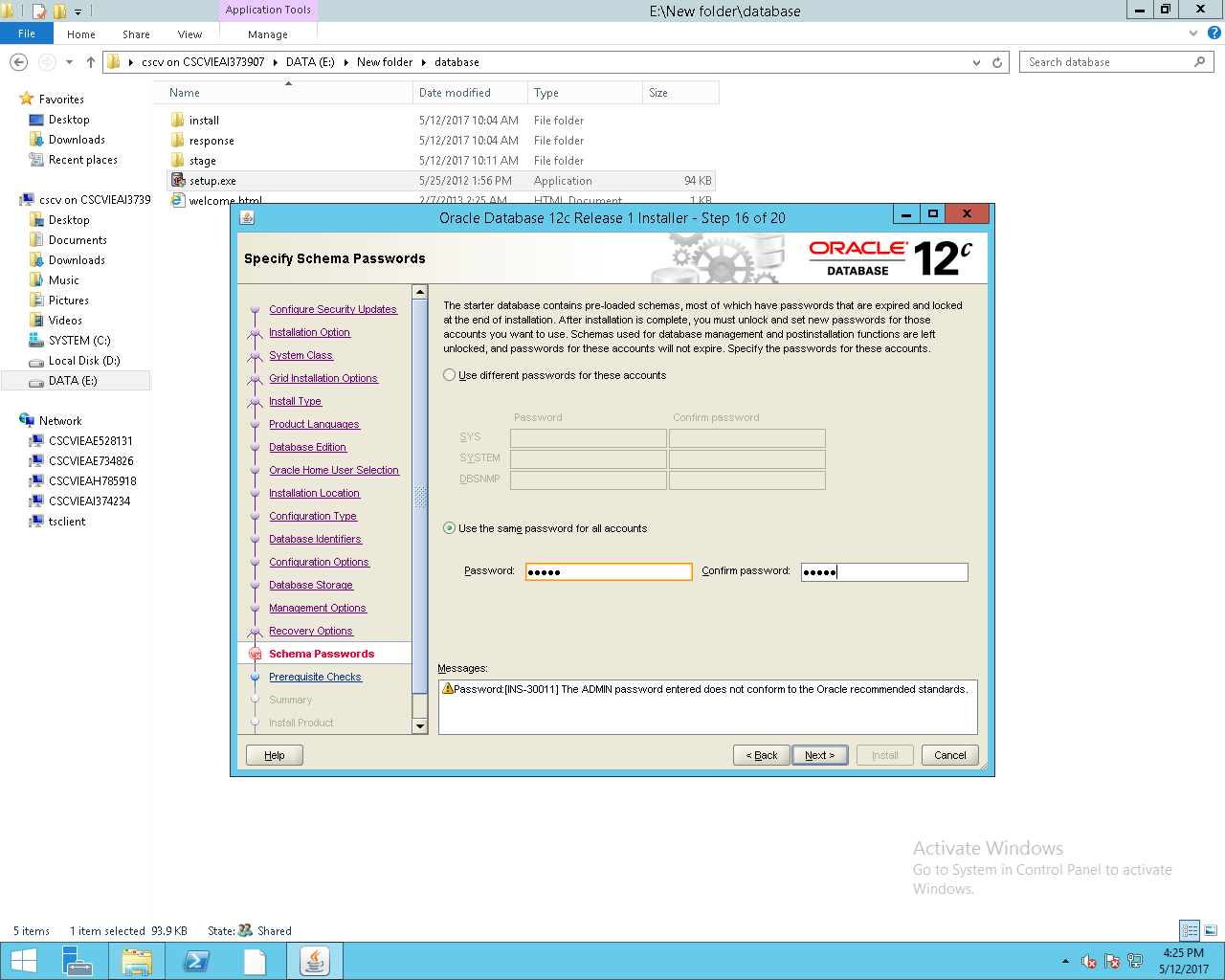




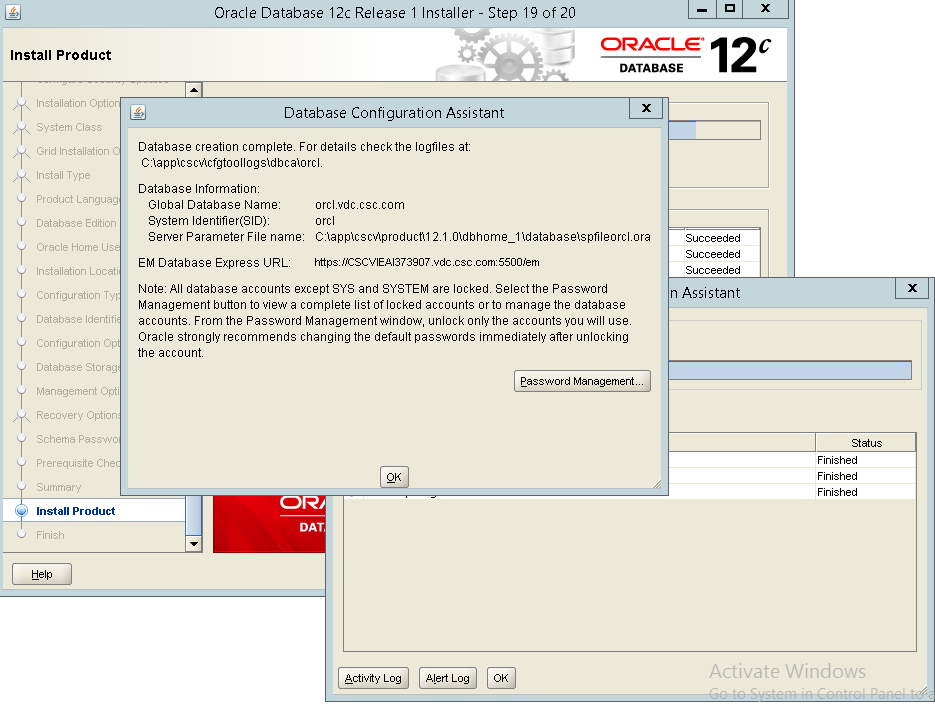
Nơi lưu database file:



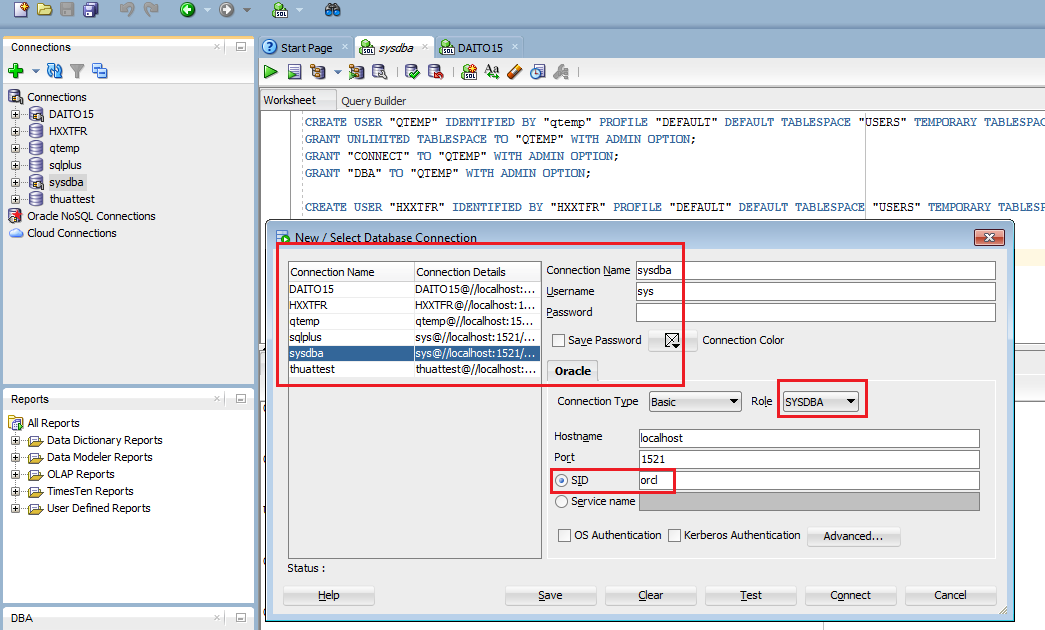
Password: daito



Sau khi setup:



Khi mới tạo Oracle:



## Chạy test daito-pilot trên server của Tiến:

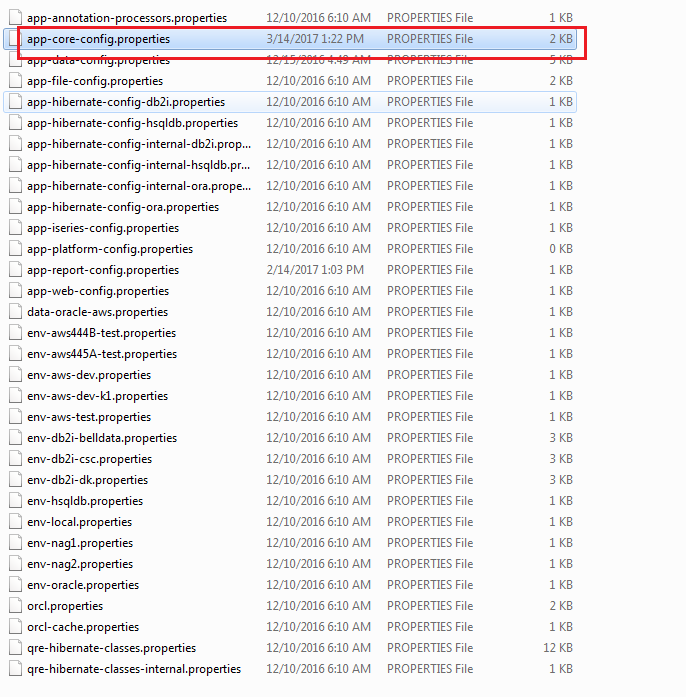
Máy server của Tiến: 146.88

Copy extract vào folder \_THUAT\_, sau đó copy vào file java bên pilot, rồi config 444 hoặc 445 như hình dưới.

Chạy 0.bat

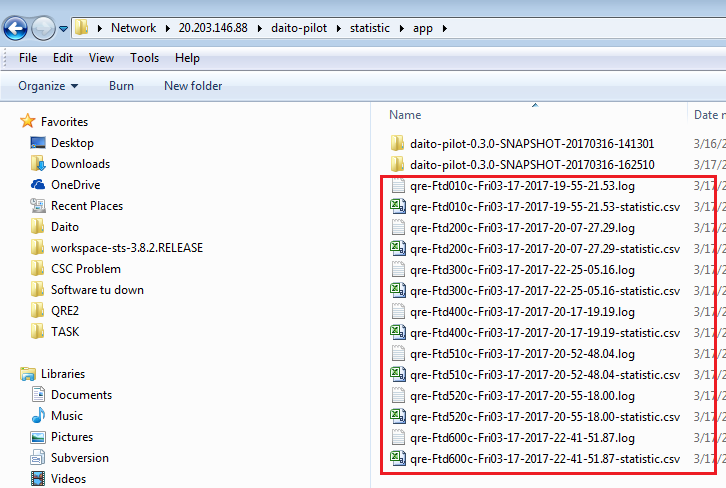
Chạy run\_444.bat

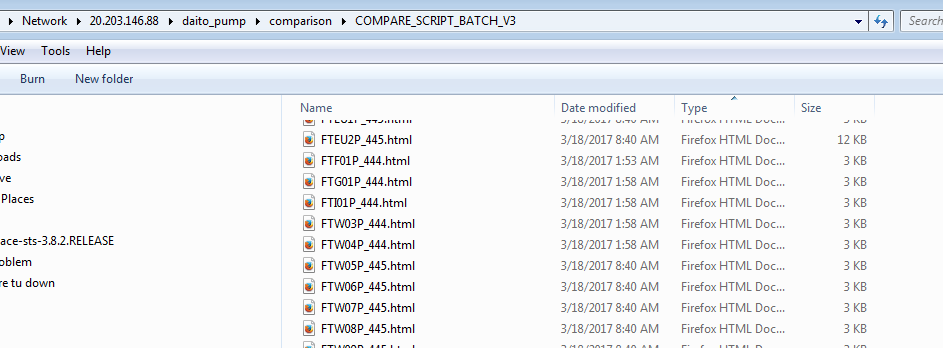
Config file config rồi chạy run\_445.bat



Sau đó sử dụng remote

Compare thì chạy run





## Cách để insert & vào oracle:

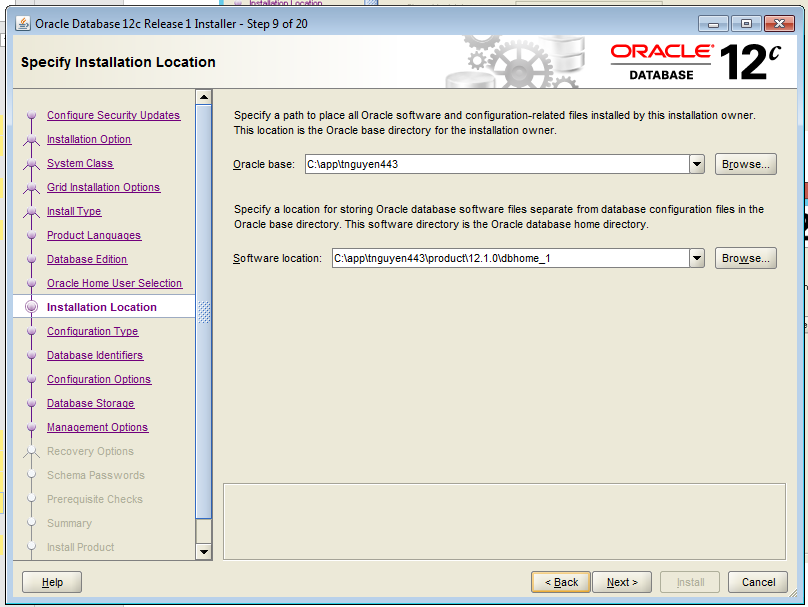
set define off;

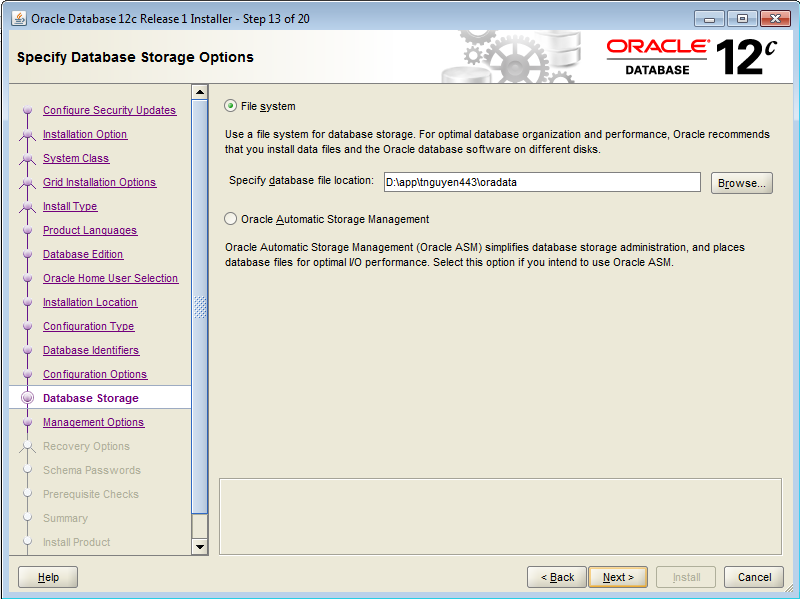
insert…

## Listener

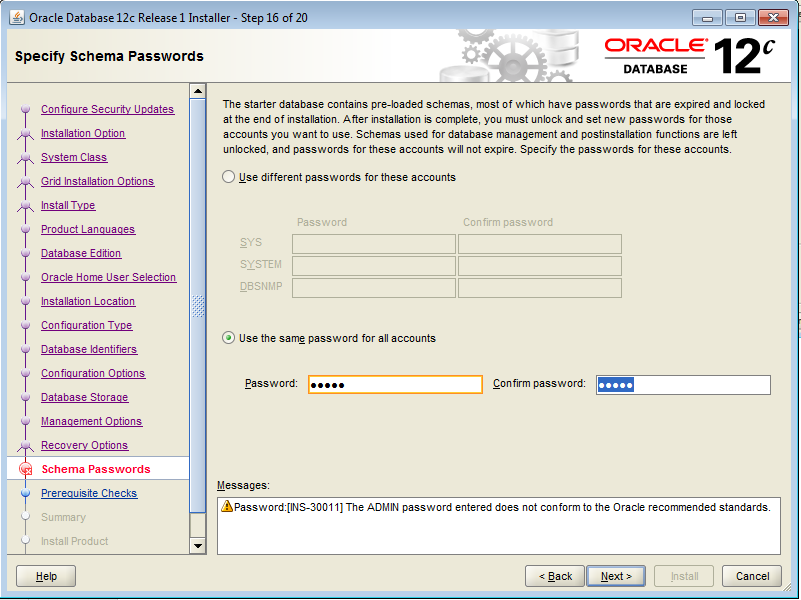
<https://docs.google.com/document/preview?id=15HAWsUXeOE_iORTxmxr3PhznKk4L2dZWl8tab1xnL1k>

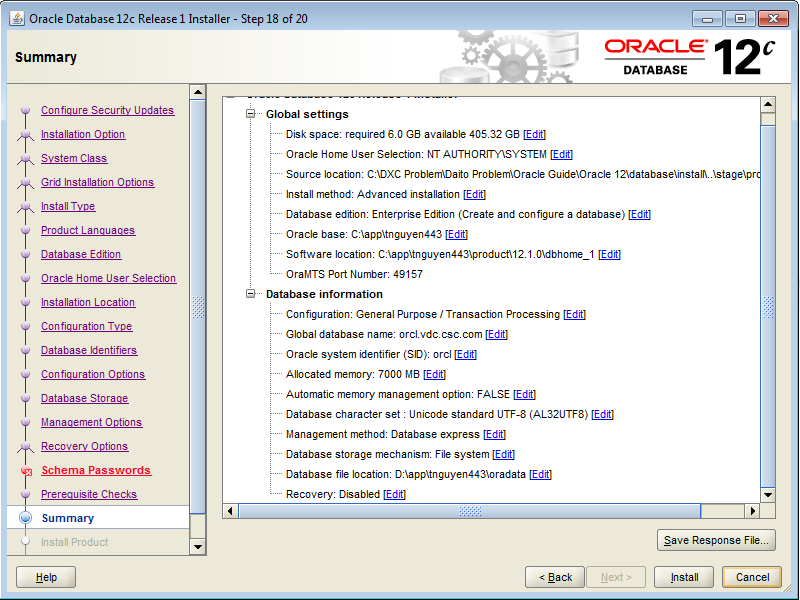
# Setup tren may minh:





Password: daito





# QTE-5859

Sử dụng địa chỉ sau đây để lấy file pump:

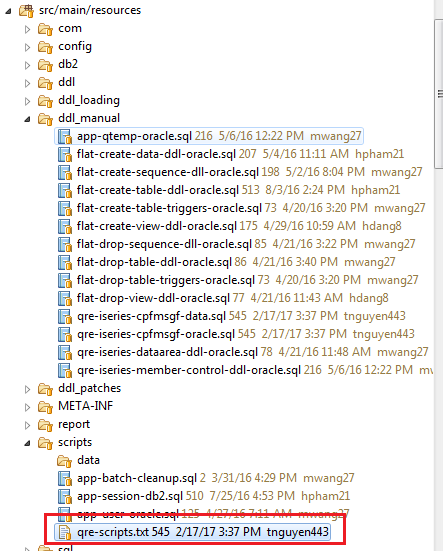
\\cscvieae528140.vdc.csc.com

Chạy qre-scripts trước khi tạo check point

Restore point của mình là PRE\_PUMP\_FULL15.

Nếu muốn revert (mỗi khi test cái mới) thì chạy cái 0, muốn tạo cái mới thì chạy cái 1

File script cần chạy trước khi chạy cái test nằm ở đây:



daito-as01-0.3.0-SNAPSHOT-20170321-115615.tar: build code mới nhất ngày 21/03/2017

daito-as01-0.3.0-SNAPSHOT-20170322-141112.tar: build ngày 22/03/2017 sau khi revert lại 1 version cho CommandLineBatchController( cái này Max sửa ngày 24/02, không biết có phải cái này gây lỗi k)

sửa cái env-aws-dev-a1.properties từ daitopd1 thành orcl

## Bỏ ngày tháng ở đầu file log:

([0-9].\*main])

## Ngày 03/23/207

### 1. Table FBUTUI thiếu data nên báo lỗi:

**Cpf2869: Empty member \*FIRST in file FBUTUI in library DAITO15 is**

**not copied.**

Thêm data vào table này:

028971108846970408187051000009999991187050397040897041802H60881001870501000000832900034770000000000000084119971219970408 0018705003 00809197042200999999 300

### 2. Table CTGB3300JG có vấn đề

Cpf9815: Member null file CTGB3300JG in library \*LIBL not found.

at com.daito.as01.hxx.clp.ctgb.procedure.CHashTgb3300j.process(CHashTgb3300j.java:877)

Lỗi do trong ChashTgb3300j

cmdExecutor.overrideDatabaseFile(**this**, OverrideDatabaseFileParameter.*build*().setFile("SYSIN").setToFile("CTGB3300JG").setMember("\*FIRST"));

### 3. Mismatch, nguyên nhân có thể do thiếu data

Cần nhờ anh Kiệt để create dump file:

Nguyen, Kiet Van <knguyen78@csc.com>

## 3. Tổng hợp cách làm:

1. Chạy 0.bat của Tiến

2. Chạy 0.sql, 1.sql trên sqldevloper, 2.sql, 3.sql cho daito15

3. Thêm data vào table này vào FBUTUI

028971108846970408187051000009999991187050397040897041802H60881001870501000000832900034770000000000000084119971219970408 0018705003 00809197042200999999 300

4. Tạo User QTEMP:

CREATE USER "QTEMP" IDENTIFIED BY "qtemp" PROFILE "DEFAULT" DEFAULT TABLESPACE "USERS" TEMPORARY TABLESPACE "TEMP" ACCOUNT UNLOCK;

GRANT UNLIMITED TABLESPACE TO "QTEMP" WITH ADMIN OPTION;

GRANT "CONNECT" TO "QTEMP" WITH ADMIN OPTION;

GRANT "DBA" TO "QTEMP" WITH ADMIN OPTION;

5. Chạy qtemp.sql

6. Tạo User DAITOEX: (chạy lần lượt 3 cái trong file create daitoex.txt trong QTE-5859)

CREATE TABLESPACE "EX\_DATA" DATAFILE 'D:\DAITO\_DATA\_FULL\EX\_DATA' SIZE 16G AUTOEXTEND ON NEXT 8G MAXSIZE UNLIMITED;

Commit;

CREATE TEMPORARY TABLESPACE "EX\_TEMP" TEMPFILE 'D:\DAITO\_DATA\_FULL\EX\_TEMP' SIZE 16G AUTOEXTEND ON NEXT 8G MAXSIZE UNLIMITED;

Commit;

CREATE USER "DAITOEX" PROFILE "DEFAULT" IDENTIFIED BY "daitoex" DEFAULT TABLESPACE "EX\_DATA" TEMPORARY TABLESPACE "EX\_TEMP" ACCOUNT UNLOCK;

GRANT UNLIMITED TABLESPACE TO "DAITOEX" WITH ADMIN OPTION;

GRANT "CONNECT" TO "DAITOEX" WITH ADMIN OPTION;

GRANT "DBA" TO "DAITOEX" WITH ADMIN OPTION;

GRANT read, write on directory daito\_pump to daitoex;

Commit;

7. Vào daitoex chạy 00.drop primary key in daitoex.sql

01

02

03

04 data

04 v2

8. chạy call-only.bat local

# Cách làm 5859 cho server 140

## Chạy 0.bat để restore database:

D:\daito\_pump\daito1

## Đổi password daito15

## Lỗi thiếu Table OUTQ

Caused by: org.springframework.jdbc.BadSqlGrammarException: PreparedStatementCal

lback; bad SQL grammar [SELECT LIBRARY FROM DAITO15.OUTQ WHERE NAME=?]; nested e

xception is java.sql.SQLSyntaxErrorException: ORA-00942: table or view does not

exist

Tuy nhiên không thể tạo bằng oracle 11 với câu lệnh này:

**DROP** **TABLE** OUTQ **CASCADE** CONSTRAINTS;

**CREATE** **TABLE** OUTQ (

ID **INTEGER** GENERATED **BY** **DEFAULT** **AS** IDENTITY (START **WITH** 1 INCREMENT **BY** 1) **NOT** **NULL**,

LIBRARY **VARCHAR**(10) **NOT** **NULL**,

NAME **VARCHAR**(10) **NOT** **NULL**,

**PRIMARY** **KEY** (ID)

);

Phải thay thế bằng trigger sau:

DROP TABLE OUTQ cascade constraints;

CREATE TABLE OUTQ (

ID INTEGER NOT NULL,

LIBRARY VARCHAR(10) NOT NULL,

NAME VARCHAR(10) NOT NULL,

PRIMARY KEY (ID)

);

DROP SEQUENCE OUTQ\_SEQ;

CREATE SEQUENCE OUTQ\_SEQ START WITH 1;

CREATE OR REPLACE TRIGGER TRIGGGER\_OUTQ\_SEQ

BEFORE INSERT ON OUTQ

FOR EACH ROW

BEGIN

SELECT OUTQ\_SEQ.nextval

INTO :new.ID

FROM dual;

END;

## Lỗi thiếu SPOOL\_FILES

-- create table

DROP TABLE SPOOL\_FILES cascade constraints;

CREATE TABLE SPOOL\_FILES (

ID INTEGER NOT NULL,

SPLFNAME VARCHAR(10) NOT NULL,

SPLFNBR INTEGER,

JOBNAME VARCHAR(10) NOT NULL,

JOBUSR VARCHAR(10) NOT NULL,

JOBNBR VARCHAR(6) NOT NULL,

INTJOBID VARCHAR(16),

INTSFID VARCHAR(16),

JOBSYSNAME VARCHAR(10),

PGMNAME VARCHAR(10),

PGMLIB VARCHAR(10),

STATUS VARCHAR(10) NOT NULL,

OUTQ VARCHAR(30),

OUTQLIB VARCHAR(30),

COPIES INTEGER,

CPI VARCHAR(10),

PAGELNGTH INTEGER,

PAGEWTH INTEGER,

TOTALPAGES INTEGER,

FORMTYPE VARCHAR(20),

FONT VARCHAR(20),

DEV VARCHAR(20),

LPI VARCHAR(10),

OUTPTY VARCHAR(10),

DIR\_NAME VARCHAR(250),

OVERFLOW INTEGER,

FILESEP VARCHAR(10),

SAVE CHAR(1), --BOOLEAN

CRTDATE TIMESTAMP,

EXPDATE TIMESTAMP,

LASTCLCOMMAND VARCHAR(30),

FRONTMGN VARCHAR(10),

HOLD CHAR(1), --BOOLEAN

IGCCPI VARCHAR(10),

IGCDTA CHAR(1), --BOOLEAN

IGCSOSI VARCHAR(10),

LVLCHK CHAR(1), --BOOLEAN

RPLUNPRT VARCHAR(10),

SECURE CHAR(1), --BOOLEAN

"SHARE" CHAR(1), --BOOLEAN

"FILE" VARCHAR(10),

TOFILE VARCHAR(10),

USRDTA VARCHAR(10),

XSLTEMPLATE VARCHAR(1000),

XMLFILEPATH VARCHAR(1000),

FILE\_CONTENT CLOB,

PRIMARY KEY (ID)

);

--create sequence

DROP SEQUENCE SPOOL\_FILES\_SEQ;

CREATE SEQUENCE SPOOL\_FILES\_SEQ START WITH 1;

-- create trigger using the sequence

CREATE OR REPLACE TRIGGER TRIGGGER\_SPOOL\_FILES\_SEQ

BEFORE INSERT ON SPOOL\_FILES

FOR EACH ROW

BEGIN

SELECT SPOOL\_FILES\_SEQ.nextval

INTO :new.ID

FROM dual;

END;

## Thêm message rpg9001

set define off;

insert into APP\_MESSAGE (MESSAGE\_ID,MESSAGE\_FILE,MESSAGE\_LONG\_DS) values ('RPG9001','CPFMSGF','Error &2 caused program &1 to stop.');

insert into APP\_MESSAGE\_VAR (MESSAGE\_ID,MESSAGE\_FILE,MESSAGE\_VAR\_NO,MESSAGE\_VAR\_LN,MESSAGE\_DEC\_PLACE,MESSAGE\_VAR\_TY) values ('RPG9001','CPFMSGF','1','10','10','CHAR');

## Copy file java vào đây

C:\daito-as01\java

## Chạy call-only.bat local ở đây

C:\daito-as01

## File statistic sẽ văng ra ở đây

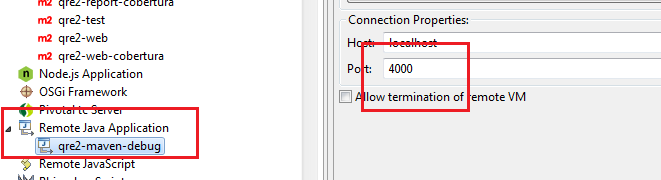
D:\daito-as01\statistic\app

## Vào đây để chạy cái comapare

D:\daito\_pump\comparison\_20160620

Chạy cái run\_comparison\_full.bat

# 41.11 Debug khi chạy test TED328



(debug->debug configuration)

Số ban đầu là 5005, muốn debug thì chỉnh lại thành 4000

Muốn so sánh cần tạo daitoex

Sau đó import file dump bằng i15\_daitoex.bat

# QTE-5880 (closed)

## ADD

If factor 1 is specified, the ADD operation adds it to factor 2 and places the sum in

the result field. If factor 1 is not specified, the contents of factor 2 are added to the

result field and the sum is placed in the result field. Factor 1 and factor 2 must be

numeric and can contain one of: an array, array element, constant, field name,

literal, subfield, or table name. For the rules for specifying an ADD operation, see

“Arithmetic Operations” on page 376.

Bình luận của Max:

When one of the factors is a field, a literal, or a figurative constant and the other  
factor and the result field are arrays, the operation is done once for every  
element in the shorter array

Định nghĩa về figurative constant: **Figurative constants** are reserved words that name and refer to specific **constant** values. The reserved words for **figurative constants** and their meanings are: ZERO, ZEROS, ZEROES. Represents the numeric value zero (0) or one or more occurrences of the alphanumeric character zero, depending on context.

Nghĩa là figurative constant hình như là ZERO

## SUB

If factor 1 is specified, factor 2 is subtracted from factor 1 and the difference is

placed in the result field. If factor 1 is not specified, the contents of factor 2 are

subtracted from the contents of the result field.

Result = factor1 – factor2

Result = result – factor2

## DIV

If factor 1 is specified, the DIV operation divides factor 1 by factor 2; otherwise, it

divides the result field by factor 2. The quotient (result) is placed in the result field.

If factor 1 is 0, the result of the divide operation is 0. Factor 2 cannot be 0. If it is,

an error occurs and the RPG IVexception/error handling routine receives control.

When factor 1 is not specified, the result field (dividend) is divided by factor 2

(divisor), and the result (quotient) is placed in the result field. Factor 1 and factor 2

must be numeric; each can contain one of: an array, array element, field, figurative

constant, literal, named constant, subfield, or table name.

## Z-ADD

Factor 2 is added to a field of zeros. The sum is placed in the result field. Factor 1

is not used. Factor 2 must be numeric and can contain one of: an array, array

element, field, figurative constant, literal, named constant, subfield, or table name.

The result field must be numeric, and can contain one of: an array, array element,

field, subfield, or table name.

Half-adjust can be specified.

## Z-SUB

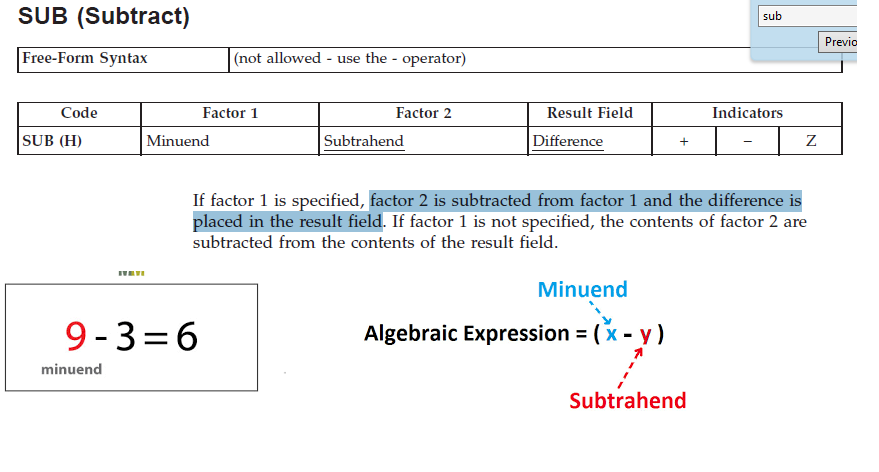
Factor 2 is subtracted from a field of zeros. The difference, which is the negative of

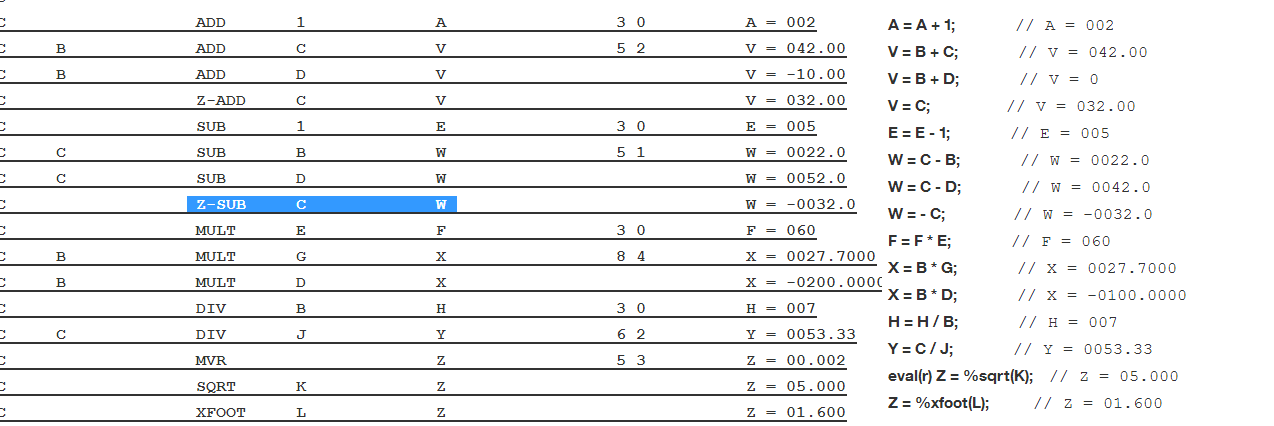
factor 2, is placed in the result field. You can use the operation to change the sign

of a field. Factor 1 is not used. Factor 2 must be numeric and can contain one of

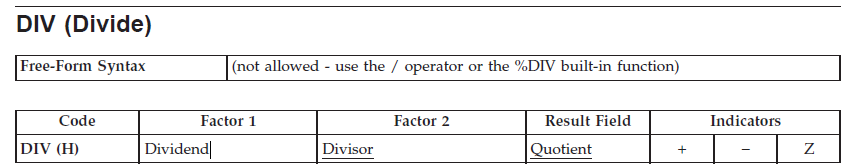
the following: an array, array element, field, figurative constant, literal, named

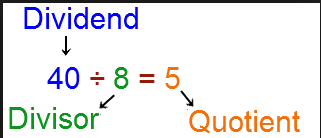
constant, subfield, or table name.





## DIV





## Chat with Max

[‎4/‎7/‎2017 10:59 AM] Wang, Jinhai:

Factor 2 is subtracted from a field of zeros. The difference, which is the negative of  
factor 2, is placed in the result field.

this is definitely not correct for Z-SUB.

z-ADD is ok.

so, request QTE change to use addArray(result, factor2)

to be consistent

Z-ADD --> addArray(result, factor2)

[‎4/‎7/‎2017 11:00 AM] Wang, Jinhai:

Z-SUB --> subArray(result, factor2)

this require transformation change for Z-ADD and Z-SUB with arrays

clear?

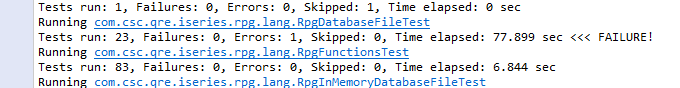
need to grab a lunch before meeting

## Chat with anh Thanh

Về Z-ADD mà factor2 không phải array

# Class not found

## Lỗi iseries-rpg



# 42. QTE-5982 (closed)

## 1. Các file Tiến và Max đã sửa:

RpgFunction

RpgFunctionTest

## 2. Các file đã thêm mới cho cái task này:

RpgFunctionDateTime

RpgFunctionArrray

## 3. Mail

Hi Max,

To add to the below, valid values for variable **P@FYMD** should bein format **‘YYYYMMDD’.**

If this statement is not monitored, the program fails as mentioned below. If this exception is monitored using opcode MONITOR, then we can assign some default to the result field. This is something like the “try - catch” exception handler in java.

Thanks,

Arun

Hi Max, Tien Vinh,

Apologize for the delay.

|  |  |  |
| --- | --- | --- |
| **If**  **P@FYMD**  **(8 Chars)** | **Then**  **W@FYMD**  **(Date as**  **YYYY-MM-DD)** | **Comments** |
| 20171022 | 2017-10-22 | Value assigned to variable W@FYMD |
| 2017102b | 2017-10-02 | ‘b’ means blank.  Value assigned to variable W@FYMD |
| 20172210 | Error | Program fails with a “Date, Time or Timestamp value is not valid” error |
| 2017-10- | Error | Program fails with a “Date, Time or Timestamp value is not valid” error |
| 12345678 | Error | Program fails with a “Date, Time or Timestamp value is not valid” error |
| 17-01-01 | Error | Program fails with a “Date, Time or Timestamp value is not valid” error |

Here is what happens when input value varies in “**C     \*ISO0         MOVE      P@FYMD        W@FYMD**” :

Let me know if you have any questions.

Thanks,

Arun

# 43 JDBC

## \*Tham khảo:

IseriesFileUtils trong qre2-iseries

## 1. Update Project bị lỗi 1.5

<properties>

<java-version>1.8</java-version>

</properties>

<build>

<pluginManagement>

<plugins>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.1</version>

<configuration>

<source>${java-version}</source>

<target>${java-version}</target>

</configuration>

</plugin>

</plugins>

</pluginManagement>

</build>

## 2. Datasource là gì

Datasource là một config kết nối đến database, nó chứa **username, password, db driver, db url**. Ví dụ như mình cần tạo ra nhiều database: dba1, dba2. Thì mình cần tạo 2 datasource để kết nối đến từng database, cần lầy data ở database nào thì ta gọi datasource tương ứng.

## 3. Khai báo file pom.xml để chạy kết nối jdbc

<spring-version>4.3.7.RELEASE</spring-version>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-core</artifactId>

<version>${spring-version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-beans</artifactId>

<version>${spring-version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>${spring-version}</version>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>6.0.6</version>

<scope>test</scope>

</dependency>

# 44. QTE-5819 (closed)

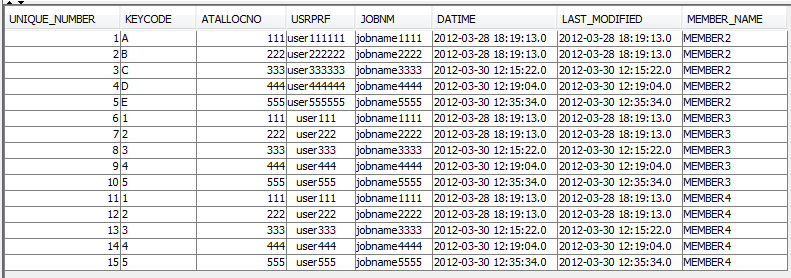
## 0. Link

Create Duplicate Object:

<https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_73/cl/crtdupobj.htm>

## 1. testCreateDuplicateObjectWithAndWithoutData():

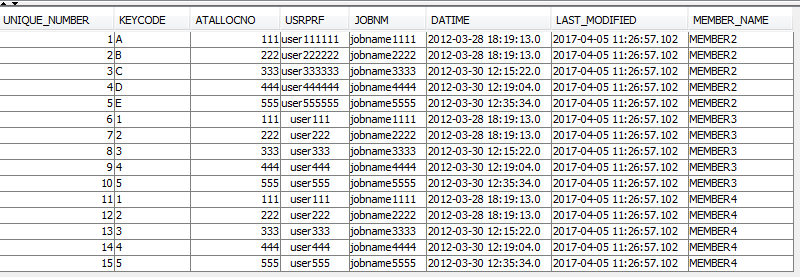
From: ISERIESPOC. ATNMPF6



To: ISERIESPOC. ATNMPF9

Table này không tồn tại

Sau khi excute:



Nghĩa là copy lại table này (duplicate lại table)

## 2. Làm cái QreExceptions

|  |  |  |
| --- | --- | --- |
| [**OBJ**](https://www.ibm.com/support/knowledgecenter/en/ssw_ibm_i_73/cl/crtdupobj.htm?view=kc#CRTDUPOBJ.OBJ) | From object | Generic name, name, \*ALL |

Hiện tại All thì quăng ra cái QreExpceptions

## 3. Làm cái Cpf2106

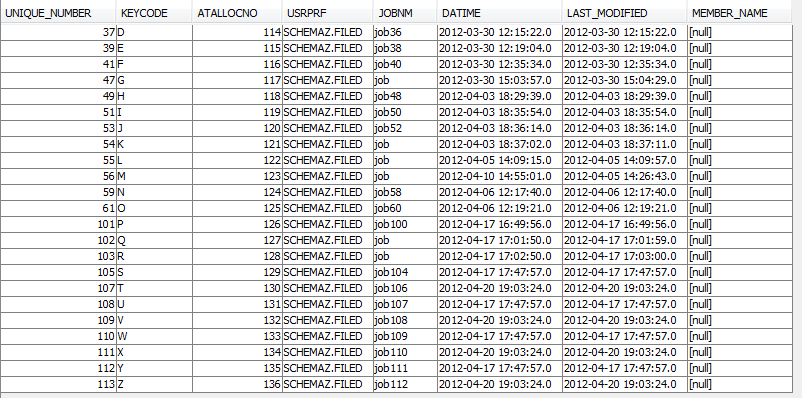
GenericName:

param.setFromLibrary("SCHEMAX");

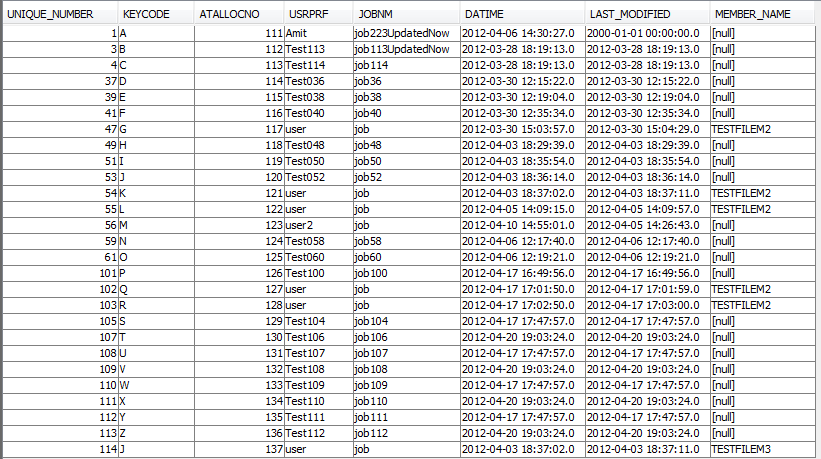
param.setObject("AXXJG\*");

AXXJG\* gọi là generic name vì có dấu \* cuối cùng

## 4. Làm cái QTEMP



## 5. Làm cái null library



## 6. Không thể rơi vào cpf2185 được vì

isGenericObjectName must be true. In this case, new object must be “\*SAME” to not throw exception Cpf2109 as logic in these line codes:

**final** **boolean** isGenericObjectName = IseriesUtils.*isGenericName*(object);

**if** ((isAllObject || isGenericObjectName) && !ClpConstants.***SAME***.equalsIgnoreCase(newObject)) {

**throw** **new** Cpf2109("CPF2109 - NEWOBJ must be \*SAME when OBJ parameter is \*ALL or generic name");

}

Because new object is “\*SAME”, so toSameName is true as logic this line code

**final** **boolean** toSameName = StringUtils.*isEmpty*(newObject) || newObject.startsWith(IseriesConstants.***ASTERISK***);

So implement can not throw exception for line code 140 as below

**if** (toSameName) {

**…** } **else** **if** (isGenericObjectName) {

**throw** **new** Cpf2185();

}

## 7. plugin check coverage unit test lấy trên marketplace

eclemma

# 45. QTE-6008 (closed)

## 1. Các program mô tả trong ticket:

BXX260R

BXX270R

Trong file [KOD551-RPG-programs.xlsx](https://vdc-jira.asia.csc.com/jira/secure/attachment/64838/64838_KOD551-RPG-programs.xlsx) thì cái này 2 program (chứ không phải include\_program)

# 46. QTE-5892

## Ticket liên quan

QTE-5556: cái này của Quân có dính đến date, đến ALL\_TAB\_MODIFICATIONS

## Các file đã sửa

RetrieveMemberDescriptionImpl

RetrieveMemberDescriptionTest

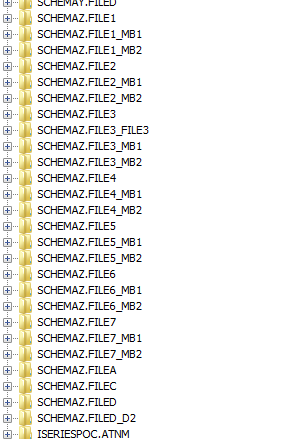
RetrieveMemberDescriptionParameter

qre\_iseries\_all\_tab\_modifications\_hsqldb.sql

## Tìm hiểu

RTVMBRD with NBRDLTRCD & CHGDATE

RTVMBRD nghĩa là Retrieve Member Description



T Thấy member đầu tiên là FILE1 và cuối cùng là

## Phân tích hàm Impl

parameter.setFile("\*LIBL/FILE1");

Như vậy:

fileQualifier = \*LIBL/FILE1

library = \*LIBL

theFile = FILE1

theSchema = SCHEMAZ; cái này khó hiểu thật, sao nó lại biêt được chỉ bằng method IseriesFileUtils.*resolveSchema*

foundMember = FILE1; đây là cái quan trọng nhất, cái này tìm được thông qua hàm private retrieveMemberName

### CL var for RTNMBR (10) (RTNMBR)

|  |  |  |  |
| --- | --- | --- | --- |
| [**RTNMBR**](https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/cl/rtvmbrd.htm?view=kc#RTVMBRD.RTNMBR) | CL var for RTNMBR (10) | *Character value* | Optional |

Specifies the name of a variable used to retrieve the name of the file member whose description is being retrieved. In CL programs, this should be a 10-character variable.

### CL var for NBRDLTRCD (10 0) (NBRDLTRCD) – hiện thực hóa là numberDeleteRecords

|  |  |  |  |
| --- | --- | --- | --- |
| [**NBRDLTRCD**](https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/cl/rtvmbrd.htm?view=kc#RTVMBRD.NBRDLTRCD) | CL var for NBRDLTRCD (10 0) | *Decimal number* | Optional |

Specifies the name of a variable used to retrieve the current number of deleted records in this file member. In CL programs, this should be a 10-position decimal variable. Zero (0) is returned for keyed logical files. Remote non-AS/400 and non-S/38 files return a value of 0. For nonkeyed logical files, the number of deleted records in the based-on physical file member is returned.

### CL var for NBRCURRCD (10 0) (NBRCURRCD) – hiện thực hóa là numberCurrentRecords

|  |  |  |  |
| --- | --- | --- | --- |
| [**NBRCURRCD**](https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/cl/rtvmbrd.htm?view=kc#RTVMBRD.NBRCURRCD) | CL var for NBRCURRCD (10 0) | *Decimal number* | Optional |

Specifies the name of a variable used to retrieve the current number of nondeleted records in this file member. In CL programs, this should be a 10-position decimal variable.

If the member is a keyed logical member, the number of index entries is returned. For nonkeyed logical members, the number of nondeleted records in the based-on physical file member is returned.

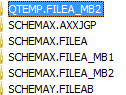
For a join logical file, the number of records returned is a total of all the nondeleted records in the files being joined. This number includes those records that exist in the secondary file which do not have matching records in the primary file. If omit or select criteria is used, the number of records returned will be reduced or increased by the number of records meeting the selection criteria in the specified files.

## Trường hợp rơi vào retrieveNumberCurrentRecords

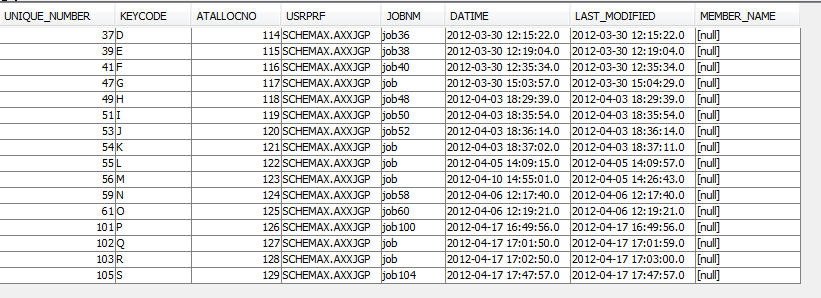
testRetrieveNumberOfRecordsFromMultipleMemberTable

Schema: SCHEMAX

fileName: AXXJGP



SCHEMAX chỉ có 4 table



AXXJGP có 16 member

Vì thế numberCurrentRecords là 16

## Làm cái CHGDATE

Có 1 câu sql trong ticket của Quân trong **AbstractSimpleDbDialectImpl** cần chú ý:

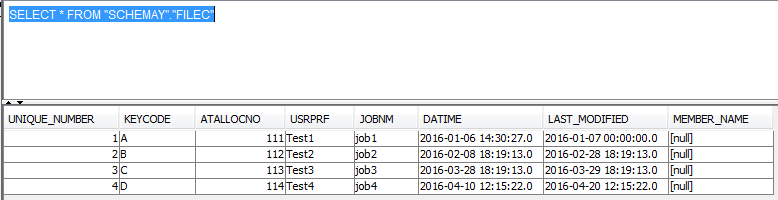
@Override

**public** String getTableStatistic(){

**return** String.format("SELECT \* FROM QRE.all\_tab\_modifications WHERE TABLE\_OWNER = ? AND TABLE\_NAME = ?");

}

## Chạy cái testRetrieveNumberOfRecordsFromSingleMemberTable



## TableStatistic

Cái này đơn giản là kết nối với

@Override

**public** String getTableStatistic(){

**return** String.*format*("SELECT \* FROM QRE.all\_tab\_modifications WHERE TABLE\_OWNER = ? AND TABLE\_NAME = ?");

}

# 47. Tìm hiểu rebase

[10:36:44 AM] Thông Võ: Đang ngồi fix conflict tè le nè.

[10:36:49 AM] Thông Võ: Fix hoài nó ếu hết là sao ta. :(

[10:37:03 AM] Thuat Nguyen: conflict lúc commit lên git hả?

[10:37:24 AM] Thông Võ: Lúc rebase á.

[10:37:26 AM] Thông Võ: :(

[10:38:07 AM] Thuat Nguyen: rebase là gì @@

[10:39:12 AM] Thông Võ: Rebase nó giống merge á, có điều là nó sẽ gắn những cái mình commit vào đuôi cái branch mà mình rebase.

[10:39:33 AM] Thông Võ: Nên coi trên history commit thì nó liên tục khá là đẹp.

[10:40:14 AM] Thông Võ: Còn merge thì nó copy lại hết những gì commit trên nhánh mình merge bỏ vào nhánh của mình nên nó gây ra dư thừa và nhìn log khá là xấu.

[10:41:09 AM] Thuat Nguyen: @@

[10:41:15 AM] Thuat Nguyen: k hiểu gì luôn :D

[10:41:36 AM] Thuat Nguyen: copy lại hết những gì commit trên nhánh mình merge bỏ vào nhánh của mình

[10:41:52 AM] Thuat Nguyen: nhánh của mình mà lại copy bỏ vào nhánh của mình là sao @@

[10:42:21 AM] Thông Võ: Nhánh mình merge với nhánh của mình khác nhau mà. :|

[10:42:39 AM] Thuat Nguyen: à

[10:42:47 AM] Thông Võ: Nghĩa là ông làm trên nhánh A, ông muốn merge nhánh B vào, ông có 2 cách là merge và rebase.

[10:43:50 AM | Edited 10:46:57 AM] Thông Võ: Merge thì ông copy hết những gì có trên nhánh B, ông bỏ vào trong nhánh A của ông, nghĩa là lúc này ông có y hệt bên nhánh A (log, commit,…). Nên khi ông tạo pull request lên nhánh B, nó sẽ bị duplicate.

[10:44:29 AM] Thuat Nguyen: uh hiểu

[10:44:32 AM] Thuat Nguyen: còn rebase?

[10:44:52 AM] Thông Võ: Rebase, là ông chỉ lấy cái phần của ông, ông gắn vào commit cuối cùng của nhánh B, nghĩa là lúc này mọi người sẽ thấy mọi thứ đang được liên tục và không bị duplicate.

[10:46:01 AM] Thuat Nguyen: uh vậy nhánh A và B riêng biệt rồi

[10:46:09 AM] Thuat Nguyen: đâu liên quan gì nhau lúc rebase

[10:46:23 AM] Thuat Nguyen: chỉ khi nào merge thì mới có chuyện thôi hả?

[10:46:52 AM] Thông Võ: Thì ông làm việc là ông tạo nhánh của ông từ 1 cái nhánh nào đó mà mọi người làm chung mà.

[10:47:03 AM] Thông Võ: Thường là ông có cái nhánh dev với master chẳng hạn.

[10:47:14 AM] Thông Võ: Dev là để code, master là để production.

[10:47:24 AM] Thông Võ: Ông tạo nhánh của ông từ nhánh dev, ông làm trên đó.

[10:47:44 AM] Thông Võ: Cuối ngày ông commit lên nhánh của ông, đồng thời ông cũng muốn lấy code của mọi người từ nhánh dev về.

[10:48:04 AM] Thông Võ: Thì ông có 2 cách là merge hoặc rebase nhánh dev vào nhánh của ông.

[10:48:32 AM] Thông Võ: Khi ông chạy lệnh merge thì lúc này nhánh của ông có thông tin y hệt nhánh dev.

[10:48:52 AM] Thông Võ: Khi ông rebase thì nhánh của ông chỉ chứa thông tin của thằng commit cuối cùng trên nhánh dev.

[10:49:23 AM | Edited 10:51:59 AM] Thông Võ: Sự khác nhau sẽ xảy ra khi chủ của nhánh dev nhận được pull request của ông để gộp nhánh ông vào nhánh dev.

[10:49:54 AM] Thông Võ: Nếu ông đã chạy lệnh merge thì pull request của ông chứa luôn những commit của mọi người khác.

[10:50:13 AM] Thông Võ: Còn ông chạy rebase thì pull request chỉ chứa thông tin commit của ông thôi.

[10:50:28 AM] Thuat Nguyen: uh hiểu rồi

[10:50:36 AM] Thông Võ: Ừ.

[10:50:37 AM] Thuat Nguyen: kết quả cuối cùng thì code k khác nhau

[10:50:47 AM] Thuat Nguyen: chỉ khác nhau thông tin history log này nọ thôi hả?

[10:50:52 AM] Thông Võ: Chỉ là nó làm cho cái cây phả hệ của git đẹp hơn thôi. :D

[10:51:06 AM] Thông Võ: Và giúp việc trace log nó đơn giản.

[10:51:17 AM] Thông Võ: Lỡ sau này cần revert lại cũng dễ.

[10:51:23 AM] Thông Võ: Vì mọi thứ nó tiếp diễn liên tục.

[10:51:30 AM] Thuat Nguyen: uh hiểu

[10:51:34 AM] Thuat Nguyen: bên t làm svn :D

[10:51:59 AM] Thông Võ: SVN thì nó không chia nhánh như vậy, và repo thì làm chung.

# 48. QTE-6027 (closed)

## Tìm hiểu hàm get f1.set(o1, p1);

f1: FixedLengthString

o1: TestIseriesDto

o2: FixedLengthStringProxy

Program có exlcludename: Ace726r

File sửa của Max:

**public** **static** **void** linkFields(**final** IseriesDto o1, **final** IseriesDto o2, **final** String... excludeFields){

**final** Class<?> t2 = o2.getClass();

**final** Field[] fields = o1.getClass().getFields();

**final** **boolean** hasExcludes = !StringUtils.*isEmpty*(excludeFields);

**final** Set<String> excFields;

**if** (hasExcludes) {

excFields = **new** HashSet<>(excludeFields.length);

**for** (String s:excludeFields){

excFields.add(s.toUpperCase());

}

} **else** {

excFields = **null**;

}

**for** (**final** Field f1 : fields) {

// only process legacy data types

**if** (!LegacyData.**class**.isAssignableFrom(f1.getType()) || Modifier.*isFinal*(f1.getModifiers())) {

**continue**;

}

// check if is exclude field

**if** (hasExcludes && excFields.contains(f1.getName().toUpperCase())) {

**continue**;

}

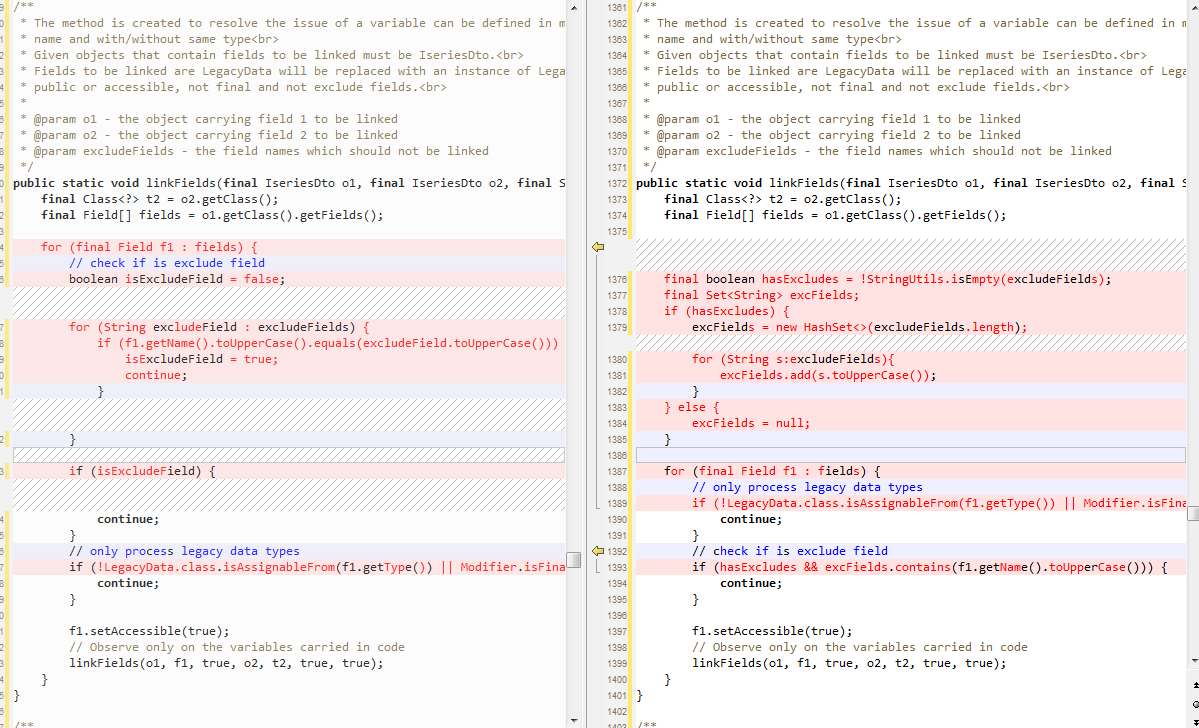
f1.setAccessible(**true**);

// Observe only on the variables carried in code

*linkFields*(o1, f1, **true**, o2, t2, **true**, **true**);

}

}



# 49. QTE-6015

Ibm: <https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_72/cl/sbmjob.htm>

JOBNUM-> LOGLEVEL int

USERNAME-> LOGSEVERITY int

JOBNAME-> LOGTEXT String

|  |  |  |  |
| --- | --- | --- | --- |
| [**LOG**](https://www.ibm.com/support/knowledgecenter/en/ssw_ibm_i_72/cl/sbmjob.htm?view=kc#SBMJOB.LOG) | Message logging | Element list | Optional |
| Element 1: Level | 0-4, **\*JOBD** |
| Element 2: Severity | 0-99, **\*JOBD** |
| Element 3: Text | **\*JOBD**, \*MSG, \*SECLVL, \*NOLIST |

# 50. QTE-6065 (closed)

Regular expression to search setTofile: setToFile\(\w.\*\)[.\*]{1}

setToFile\(\w.\*\)

setMember\([a-z]

# 51. QTE-6066 (closed)

# 52. Apply pattern regular expression

Dic100r

pDatabaseFile

hoặc l0, l1, l2, l3, l4, l5, l6

pattern: (p|l0|l1|l2|l3|l4|l5|l6)DatabaseFile

phải là 2 loại trở lên, 1 loại thì k chấp nhận được.

# 53. QTE-5888

## Các file đã sửa

DataAreaManagerImpl

DataAreaManagerTest

Package com.csc.qre.iseries.objects.impl

TableObjectHelperTest

TableObjectHelper

## Các file đã sửa 2

qre-iseries-cl.xml

DataAreaManagerImpl

DataAreaManagerTest

Package com.csc.qre.iseries.objects.impl

TableObjectHelperTest

TableObjectHelper

## testCreateDataAreaFromPersistTable

với TEST\_SOURCE = TEST/SOURCE, ta được câu insertSql trong JdbcDataAreaDaoImpl là:

INSERT INTO QRE.DATAAREA (NAME,LIBRARY,DATA,TYPE,LENGTH,DEC,DESCRIPTION,IS\_TEMPORARY) VALUES (:name,:library,:data,:type,:length,:decimal,:description,:temporary)

## testDeleteDataAreaFromTemporaryTable

Câu insertSql khi create:

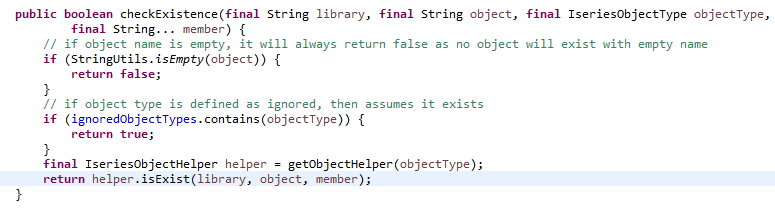
INSERT INTO QTEMP.DATAAREA (NAME,LIBRARY,DATA,TYPE,LENGTH,DEC,DESCRIPTION,IS\_TEMPORARY) VALUES (:name,:library,:data,:type,:length,:decimal,:description,:temporary)

dataAreaParam:

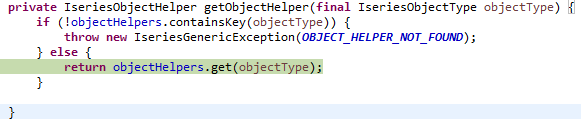
{temporary=0, library=MOCKED, data=ABCD, name=SOURCE, length=4, description=TEST, type=\*CHAR, decimal=0}

## Hàm isExist trong TableObjectHelper

IseriesObjectManager sử dụng hàm này

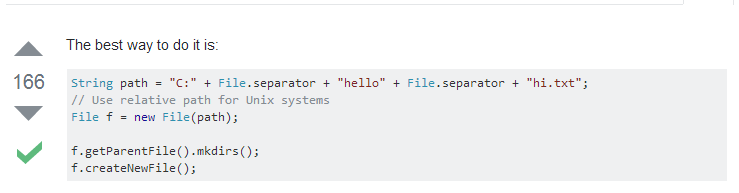


Trong đó helper được lấy từ objectHelpers:



[\*FILE=com.csc.qre.iseries.objects.impl.TableObjectHelper@7d0cd23c](mailto:*FILE=com.csc.qre.iseries.objects.impl.TableObjectHelper@7d0cd23c)

## Hàm tạo file



The best way to do it is:

String path = "C:" + File.separator + "hello" + File.separator + "hi.txt";

// Use relative path for Unix systems

File f = new File(path);

f.getParentFile().mkdirs();

f.createNewFile();

# 54. TED328 04\_24\_2017

## 1. Địa chỉ share trên local để access máy 140 vào lấy

\\CSCVIEAI374839\Users\tnguyen443\Desktop\Daito\TASK\TED328 04\_24\_2017

# 55. Dic100r

## 0. Các file đã sửa

env-local.properties : file này chỉ sửa local, nếu có submit thì k submit file này

## 1. invalid username password

java.sql.SQLException: ORA-01017: invalid username/password; logon denied

Sửa file env-local.properties đổi password lại thành daito15 chứ không phải DAITO15

## 2. null của ChashTgb3300j

com.daito.as01.hxx.clp.ctgb.procedure.CH

ashTgb3300j is not resolved as in type of null

vào call-only.bat sửa lại đường dẫn

## 3. Flat record size for file DAITO15.QPRMSRC is zero

Flat record size for file DAITO15.QPRMSRC is zero

Cái này có thể do thiếu QprmsrcDto and QprmsrcDo. Trong iteration 2 chưa có 2 file này.

# 56. QTE-5411

## Các file đã sửa

1. Ajh110r (sửa retrieve)

2. Ajh115r (sửa retrieve)

3. Exm050r (sửa retrieve)

4. Fdx250r (sửa retrieve)

5. Hda110r (sửa retrieve)

6. Hda500r (sửa retrieve)

7. Hex140r (sửa retrieve)

8. Hex145r (sửa retrive)

9. Hex150r (sửa retrieve)

10. Hex155r (sửa retrieve)

11. Hex170r (sửa retrieve)

12. Hex175r (sửa retrieve)

13. DataArea (delete 2 method)

14. DataAreaImpl (delete 2 method)

## 6.28.7 Unlock a Data Area or Release a record (UNLOCK)

This code transformed as in **RPG to JAVA Transformation QRE2.** There is only 1 program in iteration-2 use this one is **Asxt10r,** the legacy code:

{code}

C UNLOCK DKENKNML0 91

{code}

Transform to:

{code}

dkenknml0DatabaseFile.unlock(indArea.get(***IND91***));

{code}

## 6.10.2 \*DTAARA DEFINE

This case has been checked and fixed as patch file.

## 6.8.1 Data Area Operations

As **RPG to JAVA Transformation QRE2** page 79,the code is transformed as below

Legacy code:

{code}

C IN @TODA2

{code}

Transform to:

{code}

RpgFunctions.retrieveDataArea(**this**, atToda2, "@TODA2");

{code}

But in program EXM050R, I see

Legacy code:

{code}

C IN ZZZP00

{code}

Transform to:

{code}

*retrieveDataArea*(**this**, zzzp00, DataArea.***LDA***);

{code}

This is not same as **RPG to JAVA Transformation QRE2,** I think the right transformed code must be:

{code}

*retrieveDataArea*(**this**, zzzp00, “ZZZP00”);

{code}

The same issue occurs in programs: Hda110r, Hda500r, Hex140r, Hex145r, Hex150r, Hex155r, Hex170r, Hex175r

## 6.8.2 Data Area Operations OUT

Legacy code:

{code}

C OUT ZZZP00

{code}

This is not transformed at the moment. This code appear in programs: HDA110R, HDA500R.

## 5.4.4

FLS091R: cái này đúng

This is transformed as expected as in **\*RPG to JAVA Transformation QRE2**

# 57. QTE-5922

## Các file liên quan:

CopyFromImportFileParameter

# 58. Parameter option

## 1. memberOption

|  |  |  |  |
| --- | --- | --- | --- |
| [**MBROPT**](https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/cl/cpyfrmimpf.htm?view=kc#CPYFRMIMPF.MBROPT) | Replace or add records | **\*ADD**, \*REPLACE, \*UPDADD | Optional |

### \*UPADD

Không support \*UPADD now

**if** (memberOption.equals("\*UPDADD")) {

**throw** **new** InvalidOperationException("Member Option \*UPDADD is not supported for now.");

}

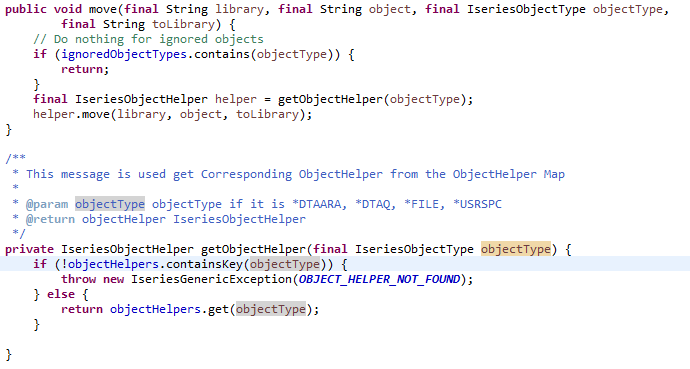
### \*REPLACE

Replace target member.

## 2. objectType

### FILE - IseriesObjectType.***FILE***

Nếu objectType thì sẽ chọn “**TableObjectHelper”,** because:

  
với objectHelpers:

{\*LIB=com.csc.qre.iseries.objects.impl.LibraryObjectHelper@43deba42, \*FILE=com.csc.qre.iseries.objects.impl.TableObjectHelper@e795a1d, \*PGM=com.csc.qre.iseries.objects.impl.ProgramObjectHelper@38c10190, \*USRSPC=com.csc.qre.iseries.userspace.impl.UserSpaceManagerImpl@1cc41b77, \*DTAQ=com.csc.qre.iseries.dataqueue.impl.DataQueueManagerImpl@2a7ebe07, \*DTAARA=com.csc.qre.iseries.dataarea.impl.DataAreaManagerImpl@6e93b0e7, \*OUTQ=com.csc.qre.iseries.printing.OutputQueueManager@5eeee124}

## 3. outputMemberOptions

|  |  |  |  |
| --- | --- | --- | --- |
| [**OUTMBR**](https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/cl/dspusrprf.htm?view=kc#DSPUSRPRF.OUTMBR) | Output member options | *Element list* | Optional |
| Element 1: Member to receive output | *Name*, **\*FIRST** |
| Element 2: Replace or add records | **\*REPLACE**, \*ADD |

## 4. Char(1)

|  |  |  |  |
| --- | --- | --- | --- |
| 4 | Initial value | Input | Char(1) |

Trong file Parameter:

**private** String initialValue;

# 59. Table sử dụng trong test

## Cách lấy table và schema từ File

fromQualifier = IseriesObjectQualifier.*forQualifiedName*(fromFile);

fromTable = fromQualifier.getName();

fromSchema = IseriesFileUtils.*resolveSchema*(fromQualifier.getLibrary(), fromTable);

Nếu table có member thì sử dụng hàm sau:

String resolvedFromTableName = *resolveTableName*(fromTable, resolvedFromMemberName);

## Check SingleTable

isFromSingleMemberTable = memberManager == **null** || *isSingleMemberTable*(fromSchema, fromTable);

# 60. QTE-6133 (Multiple thread) (Closed)

## Các file đã sửa

Cpf2986 (added)

IseriesErrors (edited)

ClearPhysicalFileMemberImpl (edited)

ClearPhysicalFileMemberTest (edited)

qre-iseries-cl.xml (edited)

qre-iseries-cpfmsgf-data.sql (edited): ( 5 files)

\_ qre2-iseries

\_ daito-common

\_ daito-as01 ( 2 files, 1 ở ddl and 1 ở ddl/db2)

\_ daito-pilot

## Tài liệu

<http://www.iseriesworld.net/?messages=cl&prefix=cpf&code=2986>

<https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/cl/clrpfm.htm>

# 61. QTE-6134 - Cpd0170

## Các file đã sửa:

Cpd0170 (edited)

IseriesErrors (edited)

// chưa làm nhưng sẽ làm

qre-iseries-cpfmsgf-data.sql (edited): ( 5 files)

\_ qre2-iseries

\_ daito-common

\_ daito-as01 ( 2 files, 1 ở ddl and 1 ở ddl/db2)

\_ daito-pilot

//

Tài liệu:

<http://www.iseriesworld.net/?messages=cl&prefix=cpd&code=0170>

## Các exception liên quan:

RPG9001

## Tự tìm hiểu

1. com.csc.qre.core.exception.LegacyObjectNotFoundException: XJE200C is not resolved as in type of null

2. Cái invoke nó bay vào class ProgramInvokerImpl

3. Class ProgramInvokdeImpl

program.mainline(args); dòng code này chỉ để print log và chạy process.

/////

/\*

\* It is revised to set return code to calling program after called another program.

\*/

**if** (source != **null** && source **instanceof** AbstractCodeModel) {

((AbstractCodeModel) source).setReturnCode(program.getReturnCode());

}

**return** program.getReturnCode();

không hiểu đoạn này để làm gì vì retrunCode luôn bằng 0 mà

## Các file đã sửa 2:

com.csc.qre.iseries.lang.cl (package)

ClCommandExecutorTest

## Các ticket liên quan:

5914 – ticket của Max, liên quan đến callProgram sử dụng LegacyString, trong đó có các program: XMX000C, XMY000C

## Tìm hiểu 2

Syso ra tên class: com.csc.qre.iseries.lang.cl.ClCommandExecutorTest$SampleClp

Length is 59

# 62. QTE-6174 moveLeft testcase

## Tìm hiểu testMoveLeftFunction

### 1. moveleft với date Euro (HH.mm.ss)

rpgTime = 00.00.00

rpgTimeNum = 11121314L

after command: rpgTime = 11.12.13

### 2. moveLeft với timeStamp

resultTimestamp = 0001-01-01-00.00.00.000000

timeStampTest = 1823-03-23-12.34.20.123456

after command: 1823-03-23-12.34.20.123456

### 3. moveleft với Binary

Cách tạo số Binary:

**double** num = 333258425;

Binary var9 = *createBinary*(9, 0, num);

Command:

var9 = 333258425

var5 = 56784

after command:

var5 = 33325

### 4. moveleft từ Binary sang FixedLengthString

var9 = 903178425

var5String = AKT4D

after command:

var5String = 90317

### 5. moveLeft từ FixedLengthString sang Binary

var9String = 296379999

var5Num = 56784

after command:

var5Num = 29637

### 6. moveLeft từ FixedLengthString sang FixedLengthString

var9String = BRWCXH4SN

var5String = AKT4D

after command:

var5String = BRWCX

### 7. moveLeft với Binary với from nhỏ hơn to

var5Num = 78425

var9Num = 130943210

after command:

var9Num = 784253210

… còn nhiều nữa liên quan true false

### 8. moveLeft với ISO\_FORMAT date

(ISO\_DATE = yyyy-MM-dd)

rpgDate = today = 09.05.2017

var10String = “”

after command:

var10String = 2017-05-09

đồng thời rpgDate cũng sẽ biến đổi luôn rpgDate = 2017-05-09

### 9. moveLeft với PredefinedConstant

var5String = LLL6N

PredefinedConstant.***HIGH\_VALUE = 0xFF***

***After command with false value:***

var5String = ÿÿÿÿÿ

after command with true value:

var5String = "ÿ "

## Tìm hiểu testMoveLeftLegacyNumberArrayToLegacyStringArray

# 63. QTE-6146

[Daito][ITER3.3][QRE] Implement setIgcCpi and setShare in OverridePrinterFileParameter (Cbd800c)

<https://www.ibm.com/support/knowledgecenter/ssw_i5_54/cl/ovrprtf.htm#OVRPRTF.IGCDTA>

## Các file đã sửa

OverridePrinterFileParameter

OverridePrinterFileTest

OverridePrinterFileImpl

SpoolFileManagerJdbcTest

RpgReportFileTest

## Các file sử dụng setShareOpenDataPath

Các này wrong, because setShare mới đúng

OverridePrinterFileParameter

SpoolFileManagerJdbcImpl (cái này không liên quan)

setShareOpenDataPath (cái này không liên quan)

OverridePrinterFileTest (cần sửa lại)

## Các file sử dụng isShareOpenDataPath

OverridePrinterFileImpl (đã sửa)

OverridePrinterFileTest (không liên quan @@)

SpoolFileManagerJdbcTest(đã sửa)

## IgcCpi

**ShareOpenDataPath**

Old: dbcsCharacterPerInch

### DBCS characters per inch (IGCCPI)

Specifies the printer character density of double-byte character set (DBCS) data, in characters per inch (cpi).

**\*CPI**

DBCS density is based on the values specified for the **Characters per inch (CPI)** parameter. The system prints one double-byte character for every two alphanumeric characters. (\*CPI is the default value for this parameter on the CRTPRTF command.)

* For CPI(10), DBCS characters print at 5 characters per inch.
* For CPI(12), DBCS characters print at 6 characters per inch.
* For CPI(13.3), DBCS characters print at 6.7 characters per inch (same as IGCCPI(\*CONDENSED)).
* For CPI(15), DBCS characters print at 7.5 characters per inch.
* For CPI(18), DBCS characters print at 9 characters per inch.
* For CPI(20), DBCS characters print at 10 characters per inch.

**\*CONDENSED**

Condensed printing, where the system prints 20 double-byte characters every three inches, is used.

**5**

DBCS density is 5 characters per inch.

**6**

DBCS density is 6 characters per inch.

**10**

DBCS density is 10 characters per inch.

## Tìm hiểu thêm

# 64. QTE-6185 – assign qua cho Mẫn

<https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/cl/dspusrprf.htm>

## Các file đã sửa

DisplayUserProfileParameter

# 65. QTE-6159

## Các file đã sửa

SaveLibrary

SaveLibraryParameter

## Tài liệu:

<https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_71/cl/savlib.htm#SAVLIB.LIB>

# 65B Enviroment:

Path:

D:\app\oracle\product\12.1.0\dbhome\_1\bin;C:\ProgramData\Oracle\Java\javapath;%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\;C:\Program Files\TortoiseSVN\bin;C:\Program Files\Java\jdk1.8.0\_131\bin;C:\apache-maven-3.3.9\bin

# 66. TEA410 05\_18\_2017

## Sql

select \* from HDDDBF.EBF60P order by UNIQUE\_NUMBER;

SELECT \* FROM HDDDBF.EBF60P order by UNIQUE\_NUMBER OFFSET 0 ROWS FETCH NEXT 10000 ROWS ONLY;

SELECT ebf56d FROM OUTDATA.EBF60P\_\_0517 order by UNIQUE\_NUMBER OFFSET 10000 ROWS FETCH NEXT 20000 ROWS ONLY;

SELECT EBF57S FROM OUTDATA.EBF60P\_\_0517 order by UNIQUE\_NUMBER OFFSET 20 ROWS FETCH NEXT 10000 ROWS ONLY;

SELECT count(\*) FROM OUTDATA.ECNE0P\_\_0517 order by UNIQUE\_NUMBER OFFSET 20 ROWS FETCH NEXT 10000 ROWS ONLY;

SELECT count(\*) FROM OUTDATA.ECNE0P\_\_0517;

SELECT count(\*) FROM HMMDBF.ECNE0L2;

SELECT ecmabc FROM HMMDBF.ECMD0L6 WHERE ecmabc LIKE ' ' order by UNIQUE\_NUMBER;

SELECT ecnacc FROM HMMDBF.ecne0l2 order by UNIQUE\_NUMBER;

SELECT EBFDBB,EBF42M FROM OUTDATA.EBF60P\_\_0517 order by UNIQUE\_NUMBER OFFSET 0 ROWS FETCH NEXT 10000 ROWS ONLY;

SELECT \* FROM HMMDBF.EJISML6 ORDER BY EJISTC;

SELECT UNIQUE\_NUMBER,EBFDBB FROM HDDDBF.EBF60P WHERE EBFDBB LIKE '%85%' order by UNIQUE\_NUMBER;

SELECT COUNT(\*) FROM HDDDBF.EBF60P WHERE EBFDBB LIKE '%85%' order by UNIQUE\_NUMBER;

SELECT COUNT(\*) FROM HDDDBF.EBF60P WHERE EBFDBB LIKE '60020423320601%' order by UNIQUE\_NUMBER;

SELECT COUNT(\*) FROM HDDDBF.EBF60P WHERE EBFDBB LIKE '6002042385320601%' order by UNIQUE\_NUMBER;

SELECT UNIQUE\_NUMBER,EBFDBB FROM OUTDATA.EBF60P\_\_0517 WHERE EBFDBB LIKE '%85%' order by UNIQUE\_NUMBER;

SELECT COUNT(\*) FROM OUTDATA.EBF60P\_\_0517 WHERE EBFDBB LIKE '%85%' order by UNIQUE\_NUMBER;

SELECT EBFDBB,UNIQUE\_NUMBER FROM OUTDATA.EBF60P\_\_0517 WHERE EBFDBB LIKE '%33%' order by UNIQUE\_NUMBER;

SELECT COUNT(\*) FROM HDDDBF.EBF60P WHERE EBFDBB LIKE '%33%' order by UNIQUE\_NUMBER;

SELECT COUNT(\*) FROM OUTDATA.EBF60P\_\_0517 WHERE EBFDBB LIKE '%33%' order by UNIQUE\_NUMBER;

SELECT COUNT(\*) FROM OUTDATA.EBF60P\_\_0517 WHERE EBFDBB LIKE '60020423320601%' order by UNIQUE\_NUMBER;

SELECT COUNT(\*) FROM OUTDATA.EBF60P\_\_0517 WHERE EBFDBB LIKE '6002042385320601%' order by UNIQUE\_NUMBER;

SELECT ecn20d FROM HMMDBF.ECNE0P order by UNIQUE\_NUMBER OFFSET 10000 ROWS FETCH NEXT 90000 ROWS ONLY;

SELECT ecn21q FROM HMMDBF.ECNE0P order by UNIQUE\_NUMBER OFFSET 0 ROWS FETCH NEXT 100000 ROWS ONLY;

SELECT ecn21q FROM OUTDATA.ECNE0P\_\_0517 order by UNIQUE\_NUMBER OFFSET 0 ROWS FETCH NEXT 100000 ROWS ONLY;

SELECT ebf18q FROM OUTDATA.EBF60P\_\_0517 order by UNIQUE\_NUMBER;

SELECT ecn20d FROM OUTDATA.ECNE0P\_\_0517 order by UNIQUE\_NUMBER OFFSET 10000 ROWS FETCH NEXT 90000 ROWS ONLY;

SELECT edpabc,edpacc,EDP07S FROM OUTDATA.EDPHYP

ORDER BY edpabc;

SELECT edpabc,edpacc,EDP07S FROM HDDDBF.EDPHYP

ORDER BY edpabc;

SELECT \* FROM S\_HDDDBF.EDPHYP

where edpabc='4820901';

SELECT \* FROM OUTDATA.EDPHYP

where edpabc='4820901';

SELECT \* FROM HDDDBF.EDPHYP

where edpabc='4820901';

SELECT COUNT(\*) FROM HMMDBF.ECNE0L2 ORDER BY UNIQUE\_NUMBER;

SELECT \* FROM HMMDBF.EAA10P ORDER BY Eaaawc;

SELECT \* FROM HMMDBF.EAA10P ORDER BY Eaaawc;

SELECT Eaa11q FROM outdata.EAA10P ORDER BY Eaaawc;

SELECT ecn15q,ecn16q,ecn21q FROM HMMDBF.ecne0l2 WHERE ECNABC='009459902';

SELECT \* FROM OUTDATA.EAA10P WHERE EAA11Q=13.44;

SELECT

Table1.EBF18Q as Legacy\_EBF18Q, Table2.EBF18Q as Oracle\_EBF18Q,(CASE WHEN Table1.EBF18Q IS NOT NULL AND Table2.EBF18Q IS NOT NULL AND Table1.EBF18Q = Table2.EBF18Q THEN 'TRUE' ELSE 'FALSE' END) as "Compare"

from (select rownum as STT, Table1xx.\* from (select Table1x.\* from OUTDATA.EBF60P Table1x order by EBFABC, EBFACC, EBF55S) Table1xx) Table1

full outer join

(select rownum as STT, Table2xx.\* from (select Table2x.\* from HDDDBF.EBF60P Table2x order by EBFABC, EBFACC, EBF55S) Table2xx) Table2

ON Table1.STT = Table2.STT;

SELECT COUNT(\*) FROM OUTDATA.EBF60P;

SELECT COUNT(\*) FROM HDDDBF.EBF60P;

SELECT

Table1.EBF45M as Legacy\_EBF45M, Table2.EBF45M as EBF45M,(CASE WHEN Table1.EBF45M IS NOT NULL AND Table2.EBF45M IS NOT NULL AND Table1.EBF45M = Table2.EBF45M THEN 'TRUE' ELSE 'FALSE' END) as "Compare"

from (select rownum as STT, Table1xx.\* from (select Table1x.\* from OUTDATA.EBF60P Table1x order by EBFABC, EBFACC, EBF55S) Table1xx) Table1

full outer join

(select rownum as STT, Table2xx.\* from (select Table2x.\* from HDDDBF.EBF60P Table2x order by EBFABC, EBFACC, EBF55S) Table2xx) Table2

ON Table1.STT = Table2.STT;

SELECT EBE59M FROM HDDDBF.EBE50P WHERE UNIQUE\_NUMBER = 346169;

SELECT EBE67M, EBE59M FROM HDDDBF.EBE50P WHERE UNIQUE\_NUMBER = 346170;

SELECT

Table1.EBF35A as Legacy\_EBF35A, Table2.EBF35A as ORACLE\_EBF35A,(CASE WHEN Table1.EBF35A IS NOT NULL AND Table2.EBF35A IS NOT NULL AND Table1.EBF35A = Table2.EBF35A THEN 'TRUE' ELSE 'FALSE' END) as "Compare"

from (select rownum as STT, Table1xx.\* from (select Table1x.\* from OUTDATA.EBF60P Table1x order by EBFABC, EBFACC, EBF55S) Table1xx) Table1

full outer join

(select rownum as STT, Table2xx.\* from (select Table2x.\* from HDDDBF.EBF60P Table2x order by EBFABC, EBFACC, EBF55S) Table2xx) Table2

ON Table1.STT = Table2.STT;

SELECT COUNT(\*) FROM HDDDBF.ECC30P;

SELECT \* FROM OUTDATA.ECC30P;

SELECT

Table1.ECNACC as Legacy\_ECNACC, Table2.ECNACC as ORACLE\_ECNACC,(CASE WHEN Table1.ECNACC IS NOT NULL AND Table2.ECNACC IS NOT NULL AND Table1.ECNACC = Table2.ECNACC THEN 'TRUE' ELSE 'FALSE' END) as "Compare"

from (select rownum as STT, Table1xx.\* from (select Table1x.\* from OUTDATA.ECNE0P Table1x order by ECNABC, ECNACC, ECN55S) Table1xx) Table1

full outer join

(select rownum as STT, Table2xx.\* from (select Table2x.\* from HMMDBF.ECNE0P Table2x order by ECNABC, ECNACC, ECN55S) Table2xx) Table2

ON Table1.STT = Table2.STT;

SELECT \* FROM OUTDATA.ECC30P;

SELECT

Table1.ECC14N as Legacy\_ECC14N, Table2.ECC14N as ORACLE\_ECC14N,(CASE WHEN Table1.ECC14N IS NOT NULL AND Table2.ECC14N IS NOT NULL AND Table1.ECC14N = Table2.ECC14N THEN 'TRUE' ELSE 'FALSE' END) as "Compare"

from (select rownum as STT, Table1xx.\* from (select Table1x.\* from OUTDATA.ECC30P Table1x order by ECCBHN, ECC15N, ECC24C) Table1xx) Table1

full outer join

(select rownum as STT, Table2xx.\* from (select Table2x.\* from HDDDBF.ECC30P Table2x order by ECCBHN, ECC15N, ECC24C) Table2xx) Table2

ON Table1.STT = Table2.STT;

SELECT COUNT(\*) FROM HDDDBF.EAC30P;

SELECT

Table1.EAC05N as Legacy\_EAC05N, Table2.EAC05N as Oracle\_EAC05N, Table1.EAC03D as Legacy\_EAC03D, Table2.EAC03D as ORACLE\_EAC03D,(CASE WHEN Table1.EAC03D IS NOT NULL AND Table2.EAC03D IS NOT NULL AND Table1.EAC03D = Table2.EAC03D THEN 'TRUE' ELSE 'FALSE' END) as "Compare",

Table1.EAC04H as Legacy\_EAC04H, Table2.EAC04H as ORACLE\_EAC04H,(CASE WHEN Table1.EAC04H IS NOT NULL AND Table2.EAC04H IS NOT NULL AND Table1.EAC04H = Table2.EAC04H THEN 'TRUE' ELSE 'FALSE' END) as "Compare"

from (select rownum as STT, Table1xx.\* from (select Table1x.\* from OUTDATA.EAC30P Table1x order by EAC01D, EAC02H, EAC05N) Table1xx) Table1

full outer join

(select rownum as STT, Table2xx.\* from (select Table2x.\* from HDDDBF.EAC30P Table2x order by EAC01D, EAC02H, EAC05N) Table2xx) Table2

ON Table1.STT = Table2.STT WHERE Table1.EAC05N LIKE 'EBF020R%';

select \* from outdata.EAC30P\_EAC03D\_DIFF;

SELECT \* FROM S\_HDDDBF.EDPHYP;

SELECT

Table1.ECC14N as Legacy\_ECC14N, Table2.ECC14N as ORACLE\_ECC14N,(CASE WHEN Table1.ECC14N IS NOT NULL AND Table2.ECC14N IS NOT NULL AND Table1.ECC14N = Table2.ECC14N THEN 'TRUE' ELSE 'FALSE' END) as "Compare"

from (select rownum as STT, Table1xx.\* from (select Table1x.\* from OUTDATA.ECC30P Table1x order by ECCBHN, ECC15N, ECC24C) Table1xx) Table1

full outer join

(select rownum as STT, Table2xx.\* from (select Table2x.\* from HDDDBF.ECC30P Table2x order by ECCBHN, ECC15N, ECC24C) Table2xx) Table2

ON Table1.STT = Table2.STT;

SELECT \* FROM OUTDATA.JRN\_OR\_EDPHYP;

SELECT \* FROM OUTDATA.enxo1p;

Dữ liệu ban đầu java: S\_...

Dữ liệu sau khi chạy: HDDDBF

Dữ liệu legacy: nằm trong OUTDATA nhưng có \_\_0517 hoặc tương tự thì là database được backup của oracle theo ngày

## EBF60P

Chain: chỉ đọc

Openmode.input chỉ đọc

Openmode.both: đọc và ghi

Biến điều kiện rất quan trọng:

60: 浴室: bathroom

02: シャワー: shower

04: エアコン: air condition

23: 追焚機能 : Reheating function

85: メモリーキー: memory key

32: 台所: kitchen

06: エレベーター : elevator

01: 給湯: Hot water supply

### EBF18Q

Sql thần thánh:

SELECT

Table1.EBFABC as Legacy\_EBFABC, Table2.EBFABC as Oracle\_EBFABC,(CASE WHEN Table1.EBFABC IS NOT NULL AND Table2.EBFABC IS NOT NULL AND Table1.EBFABC = Table2.EBFABC THEN 'TRUE' ELSE 'FALSE' END) as "Compare",

Table1.EBFACC as Legacy\_EBFACC, Table2.EBFACC as Oracle\_EBFACC,(CASE WHEN Table1.EBFACC IS NOT NULL AND Table2.EBFACC IS NOT NULL AND Table1.EBFACC = Table2.EBFACC THEN 'TRUE' ELSE 'FALSE' END) as "Compare",

Table1.EBF55S as Legacy\_EBF55S, Table2.EBF55S as Oracle\_EBF55S,(CASE WHEN Table1.EBF55S IS NOT NULL AND Table2.EBF55S IS NOT NULL AND Table1.EBF55S = Table2.EBF55S THEN 'TRUE' ELSE 'FALSE' END) as "Compare",

Table1.EBFBQB as Legacy\_EBFBQB, Table2.EBFBQB as Oracle\_EBFBQB,(CASE WHEN Table1.EBFBQB IS NOT NULL AND Table2.EBFBQB IS NOT NULL AND Table1.EBFBQB = Table2.EBFBQB THEN 'TRUE' ELSE 'FALSE' END) as "Compare"

from (select rownum as STT, Table1xx.\* from (select Table1x.\* from OUTDATA.EBF60P Table1x order by EBFABC, EBFACC, EBF55S) Table1xx) Table1

full outer join

(select rownum as STT, Table2xx.\* from (select Table2x.\* from HDDDBF.EBF60P Table2x order by EBFABC, EBFACC, EBF55S) Table2xx) Table2

ON Table1.STT = Table2.STT;

# 67. QTE-6276

**Scenario: Assigning Figurative Constants to character variables or Arrays**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Source** | **Result** | | | | | | | | | |  |
|  | **Constants** | **CHAR05** | | | | | **CARR2(5)** | | | | |  |
| **MOVE** | \*HIVAL | **FF** | **FF** | **FF** | **FF** | **FF** | **FFFF** | **FFFF** | **FFFF** | **FFFF** | **FFFF** | <<< Hexa Value |
| **MOVEL** | \*LOVAL | **00** | **00** | **00** | **00** | **00** | **0000** | **0000** | **0000** | **0000** | **0000** | <<< Hexa Value |
| **MOVEL** | \*ZERO | 0 | 0 | 0 | 0 | 0 | 00 | 00 | 00 | 00 | 00 |  |
| **MOVEL** | \*BLANK | ␢ | ␢ | ␢ | ␢ | ␢ | ␢␢ | ␢␢ | ␢␢ | ␢␢ | ␢␢ |  |

**Scenario: Assigning character variables to character variables**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Source** | **Result** | |
|  | **CHAR05** | **CHAR05** | **CHAR10** |
| **MOVE** | VWXYZ | VWXYZ | ␢␢␢␢␢VWXYZ |
| **MOVEL** | VWXYZ | VWXYZ | VWXYZ␢␢␢␢␢ |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Source** | **Result** | |
|  | **CHAR05** | **CHAR05** | **CHAR10** |
| **MOVE** | ABCDEFGHIJ | FGHIJ | ABCDEFGHIJ |
| **MOVEL** | ABCDEFGHIJ | ABCDE | ABCDEFGHIJ |

# 68. TEA410 ECC30P:

Select log của legacy:

SELECT \* FROM OUTDATA.JRN\_AS\_ECC30P;

UB: before

UP: after

table=HDDDBF.ECC30L0, total bulk: 134, create: 0, update: 127, delete: 7

ECQ560R, program này delete 7 record.

# Pcus86Ds – viết unit test

@Test

**public** **void** testClearDataStructureFixedLengthStringArray() {

**final** FixedLengthString st03YoyakuJh = **new** DataStructure<FixedLengthString>(8);

FixedLengthString[] st03CkinsihaJh = *initArray*(**new** FixedLengthString[3], 1, st03YoyakuJh, 1);

st03CkinsihaJh[1].set("1");

st03CkinsihaJh[2].set("2");

st03YoyakuJh.clear();

System.***out***.println("end of TEA410");

}

# 72 Setup Man’s PC for server TEA410

Hi All,

Please use my PC ( **CSCVIEAI374809**) as a DB for FT **TEA410**.

Cause Db used TNS name, so we need to change the access url (in env-local.properties ) to :

**qre.dataaccess.url=jdbc:oracle:thin:@(DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = CSCVIEAI374809.vdc.csc.com)(PORT = 1521))(CONNECT\_DATA = (SERVER = DEDICATED) (SERVICE\_NAME = TEA410.vdc.csc.com) ) ).**

Then using your STS to run.

# Test moveArray với index – anh Mậu

moveArray từ array sang field:

In the following example, N=3. Array-to-field move with variable indexing.

 ARRX FIELD

|01|0A|02|0B|03|0C| Before |0|1|0|A|

C  MOVEA ARRX,N FIELD

|01|0A|02|0B|03|0C| After |0|2|0|B|

dest[i].set(Arrays.*copyOfRange*(source, from, to));

**private** **static** **void** doMoveArray(**final** **char**[] source, **final** LegacyString[] dest, **final** **int** start, **final** **int** end,

**final** **int** elementSize) {

**int** from = 0;

**int** to;

**for** (**int** i = start; i <= end; i++) {

to = from + elementSize;

dest[i].set(Arrays.*copyOfRange*(source, from, to));

from = to;

}

}

Với

RpgFunctions.*moveArray*(array1,4, array2,1);

Start = 1

End = 1

Elementsize = 3

Source = [7,8,9,0]

From = 0;

To = 3

Với

Start = 2

End = 2

Elementsize = 3

Source = [7,8,9,0]

From = 0;

To = 3

Hàm bên trên không sai

# EBF45M

“追焚設備完備　　　　　　　　　 “

“追焚設備完備　　　　　　　　　 “

"公　" với space 2 bytes

Thì ra hex là:

e, 5, 8, 5, a, c,

e, 3, 8, 0, 8, 0

# EAC30P.EAC03D

Bug này của Mẫn

# DBCS

Địa chỉ code oracle mê hoặc:

<http://alvinalexander.com/java/jwarehouse/openjdk-8/jdk/test/sun/nio/cs/OLD/DBCS_IBM_EBCDIC_Encoder.java.shtml>

Charset: charset is the set of characters you can use, nghĩa là tập hợp các ký tự bạn có thể sử dụng

Encoding: encoding is the way these characters are stored into memory, nghĩa là cách mà các character này được lưu trong bộ nhớ

## Trail byte range

in MS932 (CP932), Trail-Byte Ranges is 0x40 - 0xFC (except 0x7F)<br>

## Leading byte range

Lead-Byte Ranges is 0x81 - 0x9F & 0xE0 - 0xFC

/\*\*

\* in MS932 (CP932), <br>

\* Lead-Byte Ranges is 0x81 - 0x9F & 0xE0 - 0xFC<br>

\* Trail-Byte Ranges is 0x40 - 0xFC(except 0x7F)<br>

\*

\* **@param** c - char

\* **@return** - true if it's in the leading byte range

\*/

**public** **static** **boolean** isDbcsLeadingChar(**char** c) {

**return** c >= 0x81 && c <= 0x9F || c >= 0xE0 && c <= 0xFC;

}

## Web chuyển đổi raw hex

<http://dev.networkerror.org/utf8/?start=240&end=241&cols=4&search=&show_uni_int=on&show_uni_hex=on&show_html_ent=on&show_raw_hex=on&show_raw_bin=on>

## EDP07S

table=HDDDBF.EDPHYL0, total bulk: 146, create: 126, update: 20, delete: 0, elapsedTime: 165

PT: insert

UB: before update

UP: update thành cái này

AbstractRpgDatabaseFileTest

## Theo wiki

Các quy định chính xác của UTF-8 như sau (các số bắt đầu bằng *0x* là các số biểu diễn trong hệ [thập lục phân](https://vi.wikipedia.org/w/index.php?title=Th%E1%BA%ADp_l%E1%BB%A5c_ph%C3%A2n&action=edit&redlink=1))

* Các ký tự có giá trị nhỏ hơn 0x80, sử dụng 1 byte có cùng giá trị.
* Các ký tự có giá trị nhỏ hơn 0x800, sử dụng 2 byte: byte thứ nhất có giá trị 0xC0 cộng với 5 bit từ thứ 7 tới 11 (*7th-11th least significant bits*); byte thứ hai có giá trị 0x80 cộng với các bit từ thứ 1 tới thứ 6 (*1st-6th least significant bits*).
* Các ký tự có giá trị nhỏ hơn 0x10000, sử dụng 3 byte: byte thứ nhất có giá trị 0xE0 cộng với 4 bit từ thứ 13 tới 16; byte thứ hai có giá trị 0x80 cộng với 6 bit từ thứ 7 tới 12; byte thứ ba có giá trị 0x80 cộng với 6 bit từ thứ 1 tới thứ 6.
* Các ký tự có giá trị nhỏ hơn 0x200000, sử dụng 4 byte: byte thứ nhất có giá trị 0xF0 cộng với 3 bit từ thứ 19 tới 21; byte thứ hai có giá trị 0x80 cộng với 6 bit từ thứ 13 tới 18; byte thứ ba có giá trị 0x80 cộng với 6 bit từ thứ 7 tới thứ 12; byte thứ tư có giá trị 0x80 cộng với 6 bit từ thứ 1 tới thứ 6.

Hiện nay, các giá trị khác ngoài các giá trị trên đều chưa được sử dụng. Tuy nhiên, các chuỗi ký tự dài tới 6 byte có thể được dùng trong tương lai.

* Chuỗi 5 byte sẽ lưu trữ được mã ký tự chứa đến 26 bit: byte thứ nhất có giá trị 0xF8 cộng với 2 bit thứ 25 và 26, các byte tiếp theo lưu giá trị 0x80 cộng với 6 bit có ý nghĩa tiếp theo.
* Chuỗi 6 byte sẽ lưu trữ được mã ký tự chứa đến 31 bit: byte thứ nhất có giá trị 0xFC cộng với bit thứ 31, các byte tiếp theo lưu giá trị 0x80 cộng với 6 bit có ý nghĩa tiếp theo.

Nghĩa là thấy chữ c (ví dụ c3b0) là 2 byte, e là 3 byte, …

# KOD423

Config cho máy 100:

qre.core.applicationDate=20170406

app-core-config do

qre.iseries.initLibraries=QTEMP,HDURL,HDDDBF,HMMDBF,HXXWRK

thang nay la iseries config

trong daito common

# 72. Phân quyền delete cho người dùng

Right click -> Properties -> security -> edit ->add -> nhập tên and pass -> ok

# 73. Các lỗi không build được ra class target

Lỗi file pom, maven:

# 74. @Qualifier

<https://www.mkyong.com/spring/spring-autowiring-qualifier-example/>

# 75. @Dependon

@DependsOn in spring is used with @Configuration. @DependsOn sets bean that container will initialize one bean after given bean. IoC guarantees that depending bean will be created before the bean which depends on this bean. We can also achieve the same using XML application context.

# 76. CEDMEP

## Program

Cej010r

Cek010r

Các program chỉ update:

Cej020r

Giá trị ban đầu của CEDMEP: 3.253.138

# 77. BKW13P

Program:

Bkc140r

# 78. BKRKSP

QTE-6374

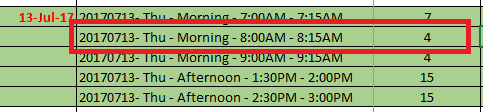
BKC200C calls Bkc210r

# 79. Japanese

CETX7P sai từ CETX3P hoặc CETX5P. Chính xác là CETX5P

Cek010c

# 80. Khám sức khỏe



# 80. Câu hỏi phỏng vấn:

Interviewcake.com

# 81. QTE-6142

## 1. Related file or document

1257 hdang8 Modified the QTE-3860

CreateUserSpace: <https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/apis/quscrtus.htm>

Listjob: <https://www.ibm.com/support/knowledgecenter/ssw_i5_54/apis/qusljob.htm>

DeleteUserSpace: https://www.ibm.com/support/knowledgecenter/ssw\_ibm\_i\_72/apis/qusdltus.htm

RetrieveUserSpace: <https://www.ibm.com/support/knowledgecenter/en/ssw_i5_54/apis/qusrtvus.htm>

## 2. Created file

CreateUserSpaceParameter (edit)1

CreateUserSpaceImpl (edit)

qre-iseries-userspace-ddl-hsqldb.sql(edit)1

qre-iseries-userspace-ddl-mssql.sql(edit)1

qre-iseries-userspace-ddl-mysql.sql(edit)1

UserSpaceManager (edit)1

UserSpaceManagerImpl (edit)1

UserSpaceDom (edit)1

UserSpaceDao (edit)

UserSpaceDaoImpl (edit)1

RetrieveUserSpaceParameter(edit)1

IseriesApiExecutor (edit) – delete mấy cái comment 6142

UserSpaceApiTest (edit)

ListJobParameter (edit)

## Làm branch

1. CreateUserSpaceParameter (edit)1

2. CreateUserSpaceImpl (edit)

3. qre-iseries-userspace-ddl-hsqldb.sql(edit)1 (bỏ cái này đi)

4. qre-iseries-userspace-ddl-mssql.sql(edit)1(bỏ cái này đi)

5. qre-iseries-userspace-ddl-mysql.sql(edit)1(bỏ cái này đi)

6. UserSpaceManager (edit)1

7. UserSpaceManagerImpl (edit)1

8. UserSpaceDom (edit)1 (bỏ cái này đi)

9. UserSpaceDao (edit) (bỏ cái này đi)

10. UserSpaceDaoImpl (edit)1 (bỏ cái này đi)

11. RetrieveUserSpaceParameter(edit)1

12. IseriesApiExecutor (edit) – delete mấy cái comment 6142

13. UserSpaceApiTest (edit)

14. ClpConstants (edit) – them constant yes no

15. ErrorCode (create) – tạo mới

16. ErrorCodeFormatErrC0100 (create) – tạo mới

17. ErrorCodeFormatErrC0200 (create) – tạo mới

18. Cpf3C2b (create) – tạo mới

19. Cpf3C2c (create) – tạo mới

20. Cpf3C2d (create) – tạo mới

21. IseriesErrors (edit)

22. qre-iseries-cpfmsgf-data.sql (edit)

23. DeleteUserSpaceImpl - edit

24. DeleteUserSpaceParameter - edit

25. ErrorCodeType – tạo mới

26. RetrieveSpaceImpl - edit

## Comment anh Nguyen

1. Create user space:  
1.1. Replace parameter:

* Use legacy data type(Legacy String) instead of java data type
* Add validation for this parameter

1.2. Error code parameter:

* Return error information with structure instead of string. Please refer to errorCode property in RetrieveSpoolFileAttributesParameter.
* Check if error code is valid or not by using "CPF3CF1 E Error code parameter not valid."

1.3.Below are parametes that need to be changed: (Not sure it is required to be changed for this ticket)

* Change "Qualified user space name" parameter as Legacy String
* Change "Extended attribute" parameter as Legacy String
* Change "Initial size" parameter as LegacyNumber
* Change "Initial value" parameter as Legacy String
* Change "Public authority" parameter as Legacy String
* Change "Text description" parameter as Legacy String
* Add validation for required parameter group.

2. Delete user space;  
2.1. Change "Qualified user space name" parameter as Legacy String  
2.2. Error code parameter: change the same as Create user space.  
2.3. Add validation for required parameter group.

3. Retrieve user space:  
3.1. Error code parameter: it has not been implemented. change the same as Create user space.  
3.2. Handle error indicator.  
3.3. Change "Starting position" parameter as LegacyNumber  
3.4. Change "Length of data" parameter as LegacyNumber

3.5. Below are parametes that need to be changed: (Not sure it is required to be changed for this ticket)

* Change "Qualified user space name" parameter as Legacy String
* Add validation for required parameter group.

4. List Job:  
4.1. Error code parameter: it has not been implemented. change the same as Create user space.  
4.2. Implement Job type, Number of fields to return, Key of fields to return parameter.

4.3. Below are parametes that need to be changed: (Not sure it is required to be changed for this ticket)

* Change "Qualified user space name" parameter as Legacy String
* Change "Qualified job name" parameter as JobQualifier
* Add validation for required parameter group.

# 82. QTE-6475 – log enhancement

AbstractIseries

AbstractCobol

AbstractCl

## Làm cái read cho anh Ngôn:

Theo anh Ngôn thì cứ 10000 thì đọc ra cái read, nhưng cái cuối cùng k in ra, vì nó dưới 2000 hay sao ấy

## Các file liên quan đến đọc dữ liệu:

RpgDatabaseFile (Cái này hình như sẽ bị bỏ)

RpgInMemoryDatabaseFileImpl

AbstractRpgDatabaseFileTest (cái này quan trọng)

## Config 10k:

app-data-config.properties – nhưng cái này chỉ có config cho daito

HibernateSessionManagerImpl: cái này để chỉnh 10.000 thì phải, cái này không phải

DataAccessConfig: cái này để chỉnh 10.000, statistic enable <- chính xác

## Các file có thể nên sửa và suggest cho Max:

InMemoryDatabaseFileImpl: kiểm tra mode debug cho log

RpgInMemoryDatabaseFileImpl

## Bắt đầu làm theo comment của Max:

RpgWorkstationFileImpl – open và close của cái này không ai xài cả

### File đã sửa:

1. InMemoryDatabaseFileImpl – log info cho open file, log trace update, delete, write

2. RpgInMemoryDatabaseFileImpl – log info cho close file, log debug

3. RpgJdbcDatabaseFileImpl – log debug

~~4. IseriesDatabaseFileImpl –~~ sửa từ debug sang info open file, close file (theo anh ngôn là k cần làm)

5. JdbcDatabaseFileImpl – log info cho open file, close file, log trace update, delete, write – Test chạy cho file này: RpgJdbcDatabaseFileTest

6. AbstractLegacyFile – log info cho open file, close file in project qre2-file

7. CobolInMemoryDatabaseFileImpl – log info, có thể sẽ làm log trace cho chỗ này

8. ClpInMemoryDatabaseFileImpl – log info

~~9. RpgDatabaseFileImpl – log debug (theo anh ngôn là k cần làm)~~

## Comment anh Ngôn:

* Qualified name should be printed rather than separate schema and file name since it contains more info such as resolved member - done
* Should include progarm name if possible, program name get in current init program in context - context.getCurrentInitProgram().get
* Pattern of data access logs should have prefix

<ACTION> file ... - program={}, qualifiedName={},

* I think no need to update for deprecated files IseriesDatabaseFileImpl, RpgDatabaseFileImpl, CobolDatabaseFileImpl. Other devs can think they are continue supported when reviewing SVN log
* CHAIN, SETLL, SETGT can be DEBUG level but READxxx are huge and should be TRACE level.
* For record before update/delete, we can add trace in fetchCurrentDto for InMemory and in getCurrentCachedDto for JDBC.
* More on debug log should be record key change happen (check changedKey in doUpdate, doDelete), record omit happen (in omitRecord())

## File config 10k:

AbstractIseriesDatabaseFileImpl

## Các file sửa lại comment anh Ngôn – hi vọng đây là lần nhúng tay cuối cùng vào:

1. AbstractIseriesDatabaseFileImpl – sửa private thành protected để có thể kế thừa

2. ClpInMemoryDatabaseFileImpl – File test: ClpInMemoryDatabaseFileTest

3. InMemoryDatabaseFileImpl – File test: RpgInMemoryDatabaseFileTest, mới chỉ sửa cho Open

4. CobolInMemoryDatabaseFileImpl

5. RpgInMemoryDatabaseFileImpl

6. JdbcDatabaseFileImpl – Làm open và close cho jdbc

7. RpgJdbcDatabaseFileImpl

## Sửa lại theo yêu cầu của Max

**QRE2-ISeries**

* AbstractIseriesDatabaseFileImpl
  + should use getCurrentProgram as it should be more accurrate than getCurrentInitProgram  
    Need to understand why getCurrentInitProgram is required - it's only set when program is loaded, but current program is set whenver a program is called
* InMemoryDatabaseFileImpl/JdbcDatabaseFileImpl
  + line 582  
    Need to add isTraceEnabled when message has method call like **IseriesDaoHelper.getPfName**, **Arrays.toString**,   
    Indeed, a property can be added in parent class as **traceEnabled**.   
    then add if (traceEnabled) for all logging statement - to minimise impact to performance considering construction of variable length array for method call.
  + also, need to include status of output operation

**QRE2-ISERIES-RPG**

* all change to **TRACE level**

## File sửa theo yêu cầu của Max:

1. RpgInMemoryDatabaseFileImpl

# 82\* 6032, 6055,6045

# 83. QTE-6592 – Run KOD551 daily

Share tar:

\\CSCVIEAI374839\Users\tnguyen443\.m2\repository\daito\daito-as01-iteration-2\0.3.0-SNAPSHOT

DB: 20.203.146.95

App:

Địa chỉ lưu file dump máy 95:

C:\daito\_pump

Các file lỗi của anh Mậu:

Hga050r

Fyb070r

## Câu check và open mode KOD551:

sqlplus /nolog

conn /as sysdba

sho pdbs

alter pluggable database kod551 open;

## Chạy latest code:

Config app-core-context.xml:

<import resource=*"classpath:META-INF/spring/qre-dbcs-support-convert.xml"*/>

app-core-config.properties: bỏ 2 cái này ra vì sử dụng dbcs-support

qre.core.stringConfig.dbcsConversionRequired=true

qre.core.stringConfig.dbcsSpaceEmulated=true

# 84. Build iter2 từ a-z bằng 1 câu command:

mvn clean install -Pqre -Piter2 -DskipTests -l ../3-build-tar-iter2.log

# 85. QTE-6595 (closed)

## File đã sửa

qre-test-iseries-config.properties

# 86. QTE-6477

Ticket liên quan cache: QTE-6520 của anh Ngôn

# 86. Config ram maven

MAVEN\_OPTS  = -Xmx8g -Xms4g

# 87.6613

Check sequence:

----------get sequence name from oracle

SELECT DATA\_DEFAULT

FROM ALL\_TAB\_COLUMNS

WHERE OWNER         = 'HMMDBF'

AND TABLE\_NAME      = 'XXLDAP'

AND IDENTITY\_COLUMN = 'YES';

---------get current sequence value

select "HMMDBF"."ISEQ$$\_91390".nextval from dual;

Check skip column:

select \* from FACTORYFILE where FILE\_NAME='BJDKMP';

select \* from FILECOLUMN where FILE\_ID='124865';

Anh Truong new’s config:

DataAccessConfig

@Autowired

**public** **void** setUidIncrementSize(@Value("${qre.data.iseries.uidIncrementSize:1}") **final** **int** uidCacheIncrementSize) {

**this**.uidIncrementSize = uidCacheIncrementSize;

}

## Detail:

If you don’t want to use factory you can only produce the row level comparison report.

1. Login to atlassian server

IP: 20.197.20.57

USR/PWD: atlassian/atlassian

1. Go to path “/home/atlassian/FactoryAutomation/scripts/db”
2. Execute the script below

Just change the DB name if you want to get the comparison for KOD551\_DBG02

./executeComparisonScripts.sh /home/atlassian/FactoryAutomation/scripts/db ec1-user Umisango KOD551\_DBG01 /home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords DIKSSP

./executeComparisonScripts.sh /home/atlassian/FactoryAutomation/scripts/db ec1-user Umisango KOD551\_DBG01 /home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords BKTBNP

./executeComparisonScripts.sh /home/atlassian/FactoryAutomation/scripts/db ec1-user Umisango KOD551\_DBG01 /home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords DIW14P

./executeComparisonScripts.sh /home/atlassian/FactoryAutomation/scripts/db ec1-user Umisango KOD551\_DBG01 /home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords XXLDAP

./executeComparisonScripts.sh /home/atlassian/FactoryAutomation/scripts/db ec1-user Umisango KOD551\_DBG01 /home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords DINWBP

./executeComparisonScripts.sh /home/atlassian/FactoryAutomation/scripts/db ec1-user Umisango KOD551\_DBG01 /home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords LADUTP

./executeComparisonScripts.sh /home/atlassian/FactoryAutomation/scripts/db ec1-user Umisango KOD551\_DBG01 /home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords KVCDGP

./executeComparisonScripts.sh /home/atlassian/FactoryAutomation/scripts/db ec1-user Umisango KOD551\_DBG01 /home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords DIW13P

./executeComparisonScripts.sh /home/atlassian/FactoryAutomation/scripts/db ec1-user Umisango KOD551\_DBG01 /home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords DIVSSP

./executeComparisonScripts.sh /home/atlassian/FactoryAutomation/scripts/db ec1-user Umisango KOD551\_DBG01 /home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords KVBDGP

./executeComparisonScripts.sh /home/atlassian/FactoryAutomation/scripts/db ec1-user Umisango KOD551\_DBG01 /home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords DIMWHP

1. Get the report from “/home/atlassian/FactoryAutomation/scripts/db/UnmatchRecords”

## Lệnh kill program:

kill -9 17469

## Lệnh compare row count:

Hi Alistair,

I have attached the necessary scripts to create the row level reports in Vietnam servers

Please perform the following steps to run the row level report scripts

1. Login as system user or any user with dba privilege
2. Run GRANT\_PRIV\_TO\_RESET\_NUMBER.SQL script once
3. Run SP\_RESET\_UNIQUE\_NUMBER.SQL script once
4. Login using QRE in target Vietnam DB server
5. Run row level report script for a particular business code in row\_level\_comparison.sql

Note: for tables that is compared using unique\_number, run the renumber script first before running the row level report scripts

Used the following scripts to create the row level scripts

with tmp1 as

(

select BUSINESS\_CODE, target\_schema, target\_table\_name, FN\_ROWLEVEL\_UNMATCH\_QRY\_RETURN(BUSINESS\_CODE, target\_table\_name)||';' as qry

from table\_keys

where ignore\_flag = 'N'

)

select BUSINESS\_CODE, target\_schema, target\_table\_name,case when instr(qry,'REVERSE(''"UNIQUE\_NUMBER',1) > 0 then 'exec "QRE"."SP\_RESET\_UNIQUE\_NUMBER"('''||target\_schema||''','''||target\_table\_name||''');' else null end, qry

from tmp1 a

order by 1,3

;

Best regards,

Andres

## Revision cho anh Ngôn:

daito-as01 560

daito-as01-data 65

daito-as01-iteration-2 348

daito-as01-report 34

daito-common 183

daito-parent 45

qre2-cobol 613

qre2-core 1830

qre2-core-test 32

qre2-data 495

qre2-file 268

qre2-iseries 2916

qre2-iseries-cobol 88

qre2-iseries-parent 37

qre2-iseries-rpg 708

qre2-iseries-web 44

qre2-parent 278

qre2-report 155

qre2-test 151

qre2-web 599

# 88. FlashBack

Lệnh xóa log:

rman target /

backup archivelog all delete input;

## Tạo flashback:

# 89. QTE-6471

## Edited/Created file:

LegacyNumberTest – thêm method testFillWithBigDecimal

# 90. QTE-6719 Delete Object

## Các file đã sửa:

1. DeleteObject – new creation

2. DeleteObjectParameter – new creation

3. DeleteObjectImpl – new creation

4. DeleteObjectTest – new creation

5. qre-iseries-cl.xml – adding bean

6. iseries\_schema\_mapping\_hsqldb.sql – edit

7. iseries\_script\_hsqldb.sql – edit

8. qre-test-iseries-member-attributes\_hsqldb.sql - edit

# 91. Thông phỏng vấn Ekino

[9:03:11 AM] Thông Võ: 1 System.out.println(123 + 321 + "00");

2 System.out.println("00" + 321 + 123);

3 System.out.println(123 + 321);

[9:03:18 AM] Thông Võ: 3 dòng này theo ông nó in ra sao?

[9:07:40 AM] Thuat Nguyen: thứ 3 chắc chắn là 444

[9:07:50 AM] Thuat Nguyen: thứ 2 t nghĩ là 00321123

[9:07:55 AM] Thuat Nguyen: thứ 1 t nghĩ báo lỗi

[9:07:59 AM] Thuat Nguyen: để check lại koi :v

[9:08:25 AM] Thông Võ: 2 và 3 ông nói đúng rồi đó, còn số 1 không lỗi nó vẫn in ra được.

[9:08:46 AM] Thông Võ: Hôm qua tui sai cái 1 và 2. :(

[9:13:19 AM] Thuat Nguyen: 44400

[9:13:20 AM] Thuat Nguyen: @@

[9:13:31 AM] Thuat Nguyen: nghĩa là 2 cái đầu nó cộng lại

[9:13:39 AM] Thuat Nguyen: sao đó nó lấy toString

[9:13:45 AM] Thuat Nguyen: rồi nối chuỗi với 00

[9:13:46 AM] Thuat Nguyen: :v

[9:13:47 AM] Thuat Nguyen: vãi

[9:14:02 AM] Thuat Nguyen: còn hỏi câu gì nữa k ông?

[9:14:15 AM] Thông Võ: Nghĩa là nếu đứng đầu là chuỗi thì nó cộng chuỗi toàn bộ.

[9:14:24 AM] Thông Võ: Còn đứng đầu là số thì cộng số trước.

[9:14:41 AM] Thông Võ: Ông thử xen kẽ chuỗi với số xem. :v

[9:14:53 AM] Thông Võ: Còn 2 câu nữa, để tui gõ vô IDE rồi paste cho ông.

[9:17:30 AM] Thuat Nguyen: ok

[9:17:41 AM] Thuat Nguyen: sao ông nhớ câu hỏi hay vậy

[9:18:16 AM] Thông Võ: Mới chiều qua mà ông, vẫn còn nhớ chứ, đâu phải tháng trước hay nửa năm gì đâu. =]]]

[9:22:36 AM] Thuat Nguyen: công ty ông mấy giờ vào?

[9:22:39 AM] Thuat Nguyen: sao ông đi sớm vậy

[9:24:02 AM] Thông Võ: byte b = 1;

b += 1;

b = b \* 2;

System.out.println(b\*2);

[9:24:12 AM] Thông Võ: Đoạn code này có lỗi hay in ra nhiêu?

[9:24:19 AM] Thông Võ: Công ty tui vào 9h. :v

[9:24:31 AM] Thông Võ: Tui vô đúng giờ chứ có sớm đâu.

[9:24:54 AM] Thuat Nguyen: ra 8 hả @@

[9:25:02 AM] Thuat Nguyen: t thấy có gì bất thường đâu :v

[9:25:16 AM] Thông Võ: ;V

[9:25:19 AM] Thông Võ: Sai rồi.

[9:25:23 AM] Thuat Nguyen: b = b\*2;

[9:25:27 AM] Thuat Nguyen: sao cái này bị lỗi ta?

[9:26:50 AM] Thông Võ: Theo như giải thích hôm qua của ông kia thì nó thấy số 2 là Integer nên nó sẽ ép kiểu thằng b từ byte lên integer để thực hiện phép nhân, sau đó nó gán kết quả vô lại b. Mà lúc này kết quả là Integer trong khi b là byte, nó không tự ép từ lớn xuống nhỏ được mà mình phải làm bằng tay cho nó. :v

[9:27:39 AM] Thuat Nguyen: chuẩn rồi

[9:27:52 AM] Thông Võ: Còn câu cuối.

[9:27:54 AM] Thông Võ: Đợi xíu.

[9:29:24 AM] Thông Võ: public class A {

public A(int i) {

System.out.println(1);

}

}

public class B extends A {

public B() {

System.out.println(2);

}

}

[9:29:33 AM] Thông Võ: Mời ông phán.

[9:29:43 AM] Thuat Nguyen: có làm vậy được k hả?

[9:29:55 AM] Thông Võ: Ừ, có error không hay compile thành công.

[9:30:15 AM] Thuat Nguyen: để suy nghĩ xíu

[9:30:26 AM] Thông Võ: Đừng gõ vô IDE nhé.

[9:30:28 AM] Thông Võ: =)))

[9:30:58 AM] Thuat Nguyen: đang học mà

[9:31:03 AM] Thuat Nguyen: gõ thì chơi làm gì :v

[9:31:22 AM] Thuat Nguyen: t nghĩ A là abstract class thì chắc là ok

[9:31:31 AM] Thuat Nguyen: còn class bình thường thì, từ từ đã

[9:31:44 AM] Thông Võ: :v

[9:31:53 AM] Thuat Nguyen: t phán đại là B phải có public B(int i)

[9:32:27 AM] Thuat Nguyen: cần overwrite lại cái constructor có tham số int i

[9:32:36 AM] Thuat Nguyen: móa mà cũng méo biết

[9:32:42 AM] Thuat Nguyen: t trả lại đại

[9:32:50 AM] Thuat Nguyen: chính xác là kết quả sao?

[9:32:58 AM] Thông Võ: Ông ghi tui câu trả lời cuối cùng của ông là gì?

[9:33:12 AM] Thuat Nguyen: t phán đại là B phải có public B(int i)

[9:33:17 AM] Thông Võ: Ok.

[9:33:25 AM] Thông Võ: Ý của ông gần đúng, nhưng bị ngược rồi.

[9:33:41 AM] Thông Võ: Chính xác là thằng A phải có constructor không tham số như thằng B.

[9:34:04 AM] Thông Võ: Vì thằng B là con, nếu nó gọi constructor không tham số của nó thì nó sẽ tự gọi constructor không tham số của thằng cha là A.

[9:34:14 AM] Thông Võ: Mà A lúc này không có constructor không tham số nên sẽ bị lỗi.

[9:34:45 AM] Thông Võ: Ông biết từ khóa super() đúng không?

[9:34:52 AM] Thuat Nguyen: biết

[9:35:00 AM] Thuat Nguyen: gọi lại method của cha thôi đúng k

[9:35:34 AM] Thuat Nguyen: cái constructor nó đặc biệt nhỉ

[9:35:46 AM] Thuat Nguyen: nó bắt buộc phải super thằng cha 1 cái ak @@

[9:36:04 AM] Thuat Nguyen: nếu t để là public void abc(){} trong B thì k vấn đề gì cả

[9:36:19 AM] Thông Võ: Vì ông đang kế thừa nó mà, nó phải khởi tạo thằng A rồi nó mới khởi tạo thằng B chứ. :v

[9:36:28 AM] Thuat Nguyen: oh hiểu rồi

[9:36:43 AM] Thông Võ: Câu đó tui cũng phán sai. :(((

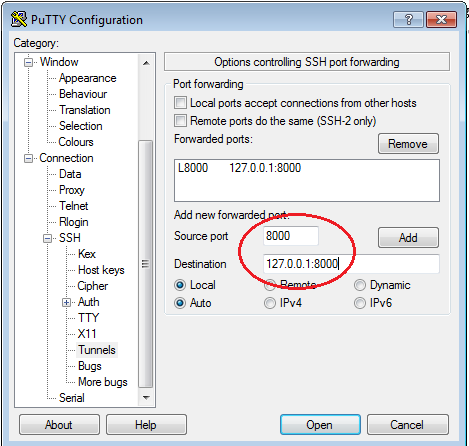
[9:36:54 AM] Thông Võ: Nhưng ổng gợi ý thì tui nói ra được.

[9:37:03 AM] Thông Võ: Rầu vl.

[9:37:04 AM] Thuat Nguyen: t cứ nghĩ constructor nó như method bình thường chứ :v

# 95.Chạy debug remote java

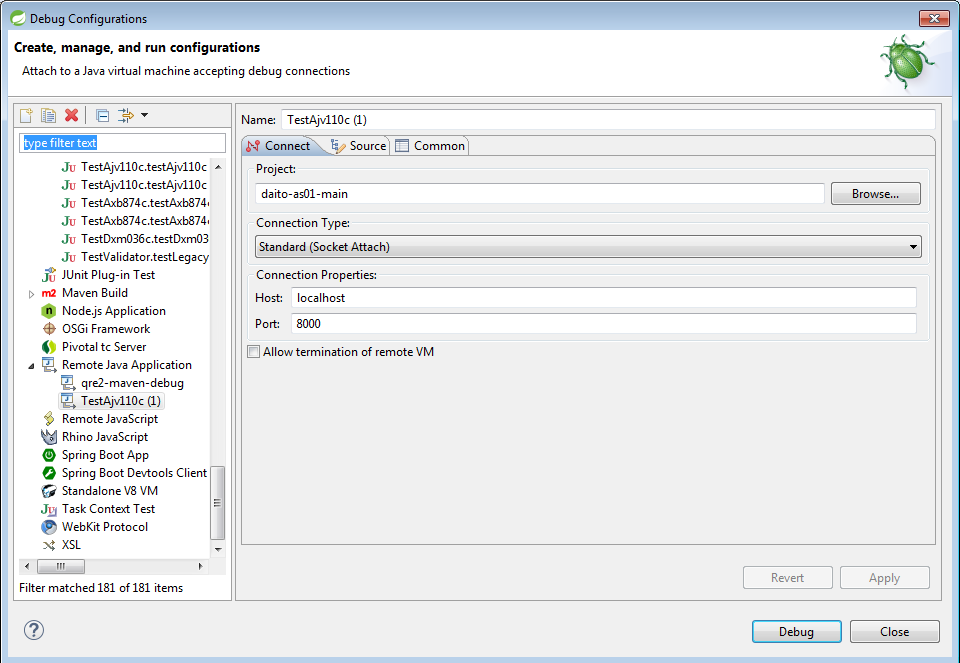
1. Chỉnh putty



2. Chỉnh sửa execute.sh

-Xmx32g -Xms32g -XX:NewRatio=1 -Xdebug -Xrunjdwp:transport=dt\_socket,server=y,suspend=y,address=8000

3. Chạy Debug Configuration



# 96. Update Revision for multiple projects

svn update -r 59 --force daito-as01-data

# 97. Commit scenario for unit test case:

/\*\*

\* Unit test for large number conversion from String.

\*

\* <pre>

\* Scenario 1: source string is "123456789012345675"

\* Expected output: target number is "123456789012345675"

\*

\* Scenario 2: Retrieve data area without ER indicator where data area name is not exist

\* Expected output: ??? is raisded as data area is not exits

\*

\* Scenario 3: Retrieve data area with ER indicator, where data area is exist

\* Expected output: the variable gets data area value as {@value #TEST} and the ER indicator is off

\*

\* Scenario 4: Retrieve data are with ER indicator, where data area is not exist

\* Expected output: the variable's value is not changed and ER indicator is on

\*

\*

\* Scenario 5: Retrieve data area without ER indicator, where data area name = \*DTAARA

\* Expected output: {@link InvalidOperationException} is raised as \*DTAARA is not supported yet

\*

\* Scenario 6: retrieve data area with lock, where the data area exits

\* Expected output: the data area can be retrieved and the data area can't be locked by another user.

\* </pre>

\*/

# 98. Cách viết code truyền tham số dạng extends từ cha

@SuppressWarnings("unchecked")

**public** **static** **final** <N **extends** LegacyNumber> N[] initArray(**final** N[] array, **final** **int** length, **final** **int** decimal,

**final** DataStructureSupport parent, **final** **int** offset, **final** LegacyDataInitializer<N> initializer) {

**final** **int** startIndex = LegacyDataConfig.*getArrayIndexBase*();

**final** **int** endIndex = array.length;

**final** LegacyNumberCreator<N> creator = *getCreator*((Class<N>) array.getClass().getComponentType());

**final** **boolean** toInitialies = initializer != **null**;

**final** **boolean** hasParent = parent != **null** && offset >= 0;

**int** lengthInBytes = -1;

**int** workingOffset = offset;

**for** (**int** i = startIndex; i < endIndex; i++) {

array[i] = creator.create(length, decimal);

**if** (i == startIndex) {

lengthInBytes = array[i].lengthInBytes();

}

**if** (hasParent) {

((DeclarationSupport<N>) array[i]).isAPartOf(parent, workingOffset);

workingOffset += lengthInBytes;

}

**if** (toInitialies) {

initializer.initialize(array[i]);

}

}

**return** array;

}

# 99. Cách comment theo link

/\*\*

\* Test method {@link ArrayDeclarationFunctions#initArray(LegacyNumber[][], int, LegacyConstant)}

\*/

@Test

**public** **void** testInitZonedDecimalArrayWithLegacyConstant(){

}

# 101. Display Hsql

org.hsqldb.util.DatabaseManagerSwing.main(**new** String[] {

"--url", "jdbc:hsqldb:mem:loadDB;sql.syntax\_ora=true", "--noexit"

});

# 100. Bonus

## 1. Set line size

Sql> set linesize 32767

## 2. Remove space line

^\s\*

## 3. Đổi user svn

