

Phát triển với PL/SQL trong IBM Data
Studio 2.2 và Optim Development Studio
2.2

Phát triển với PL/SQL trong IBM Data Studio 2.2 và Optim Development Studio 2.2

Tạo, gỡ rối và triển khai các thường trình cho DB2 và Oracle

Thomas Sharp, Kiến trúc sư công cụ phần mềm, IBM

Tóm tắt: Bài viết này mô tả cách bạn có thể sử dụng IBM® Data Studio 2.2 and Optim™ Development Studio 2.2 để phát triển các thường trình sử dụng Oracle PL/SQL. Nó chỉ cho bạn cách tạo, soạn thảo, triển khai và gỡ rối các gói, các thủ tục và các hàm PL/SQL cho các cơ sở dữ liệu sau đây: Phiên bản 9.7 của DB2® for Linux®, UNIX® và Windows®, Oracle 10g và Oracle 11g.

Tổng quát

Bài viết này cung cấp thông tin về việc sử dụng Optim Development Studio (ODS-Công cụ phát triển Optim) để tạo, soạn thảo, triển khai và gỡ rối các thường trình PL/SQL cho Phiên bản 9.7 của DB2 Linux, UNIX và Windows (LUW), các cơ sở dữ liệu Oracle 10g và Oracle 11g. Giả định bạn có kiến thức cơ bản về DB2 và Oracle và đã quen thuộc với các công việc trong một môi trường phát triển phần mềm dựa trên Eclipse.

Khả năng để phát triển, gỡ rối và triển khai khi sử dụng PL/SQL cũng có sẵn trong sản phẩm miễn phí của IBM, Data Studio 2.2. Tuy nhiên, chỉ có Data Studio hỗ trợ phát triển và triển khai dựa vào cơ sở dữ liệu DB2 9.7 trong chế độ tương thích. Để phát triển và triển khai dựa vào cơ sở dữ liệu Oracle, bạn phải sử dụng Optim Development Studio 2.2 hoặc bản phát hành mới hơn. Để thử các khả năng được mô tả trong bài viết này, xem phần Tài nguyên để tải về một bản dùng thử Optim Development Studio hoặc một bản sao sử dụng-đầy đủ của Data Studio.

Optim Development Studio là gì?

Optim Development Studio (tên mới của IBM cho Data Studio Developer) cung cấp một môi trường phát triển tích hợp dựa trên Eclipse để tăng tốc độ phát triển ở giữa dữ liệu, nhắm vào các cơ sở dữ liệu DB2, Informix và Oracle.

Ngoài các khả năng quản lý cơ sở dữ liệu cơ bản và năng lực phát triển các thường trình cơ sở dữ liệu, Optim Development Studio đưa việc phát triển cơ sở dữ liệu lên mức độ tiếp theo với các khả năng nâng cao của nó để phát triển và tối ưu hóa các ứng dụng cơ sở dữ liệu Java đối với các cơ sở dữ liệu không đồng nhất. Để biết thêm thông tin về sự hỗ trợ phát triển Java, hãy xem các trang sản phẩm Optim Development Studio và Trung tâm thông tin quản lý dữ liệu tích hợp (Integrated Data Management Information Center), từ phần Tài nguyên của bài viết này có thể liên kết đến cả hai trang đó.

Trong phiên bản trước của Optim Development Studio, bạn có thể phát triển cả các ứng dụng phía máy khách trong Java và cả logic nghiệp vụ phía máy chủ trong Java hoặc SQL khi sử dụng các trình thủ thuật tiên tiến, các trình soạn thảo, các công cụ triển khai và các trình gỡ rối. Phiên bản 2.2 cũng hỗ trợ logic nghiệp vụ phía máy chủ trong PL/SQL. Nó cho phép bạn quản lý các đối tượng PL/SQL trong Data Source Explorer (Trình thám hiểm nguồn dữ liệu) và phát triển các gói, các thủ tục và các hàm PL/SQL trong Data Project Explorer (Trình thám hiểm dự án dữ liệu).

Do Optim Development Studio 2.2 dựa trên Eclipse 3.4.2, bạn có thể cài đặt nó trong cùng một "vỏ" Eclipse với các sản phẩm khác dựa trên Eclipse 3.4.2.

PL/SQL là gì?

PL/SQL là sự mở rộng ngôn ngữ thủ tục của Oracle cho SQL. Giống như SQL bản địa cho DB2, PL/SQL là một phương ngữ của SQL cung cấp các cấu trúc 3GL để triển khai thực hiện logic chạy trên máy chủ cơ sở dữ liệu.

Bạn có thể sử dụng PL/SQL để tạo các thường trình (các thủ tục và các hàm) và để tạo các gói PL/SQL có chứa các định nghĩa kiểu dữ liệu, các khai báo biến, các thủ tục và các hàm.

Một thường trình PL/SQL có thể làm quá tải thường trình khác cùng kiểu (thủ tục hoặc hàm) chỉ khi cả hai ở trong cùng một gói PL/SQL. Nội dung của gói PL/SQL được soạn thảo cùng nhau, được triển khai cùng nhau và được gỡ rối cùng nhau. Chúng được quản lý như một đơn vị trên máy chủ cơ sở dữ liệu và được sửa đổi theo tên của gói PL/SQL và lược đồ của nó.

Một gói PL/SQL có hai phần:

- Đặc tả xác định các thường trình và khai báo ngoại lệ.
- Phần thân bao gồm các khai báo biến và con trỏ, các phần thân của các thường trình của nó và phần khởi tạo tùy chọn.

Đặc tả này là một khai báo công khai của các giao diện; phần thân là sự thực hiện riêng của nó.

Làm rõ nghĩa thuật ngữ gói (package)

Không nên nhầm lẫn một gói PL/SQL với một gói DB2 (SQL). Một gói DB2 là một đối tượng cơ sở dữ liệu. Đây là kết quả của sự ràng buộc một ứng dụng SQL hoặc thường trình để chứa đựng các câu lệnh SQL tĩnh và kế hoạch truy cập dữ liệu của DB2. Một gói PL/SQL là một cấu trúc lập trình. Nó là một thùng chứa của các câu lệnh PL/SQL.

Khi bạn triển khai một gói PL/SQL trên DB2 LUW 9.7, DB2 biên dịch nó thành một mô-đun DB2, mô-đun này là một đối tượng SQL riêng tương tự như một gói PL/SQL. Giống như một gói PL/SQL, một mô-đun DB2 chứa các điều kiện, các hàm, các thủ tục, các kiểu do người dùng định nghĩa và các biến. Không giống

như một gói PL/SQL, một mô-đun DB2 theo logic chỉ chứa các phần tử của nó. Nó không có tệp nguồn xác định nội dung của nó. Thay vào đó, sau khi bạn tạo ra một mô-đun DB2, bạn viết một chuỗi các câu lệnh ALTER để thêm các phần tử vào nó.

Để tạo một mô-đun DB2, bạn có thể sử dụng trình soạn đối tượng dữ liệu (Data Object) từ Object Editor Explorer hoặc bạn có thể gõ các câu lệnh CREATE và ALTER trong một kịch bản lệnh SQL trong dự án của bạn và sau đó chạy kịch bản lệnh đó.

Tóm tắt chức năng PL/SQL

Hỗ trợ các mô-đun DB2 và PL/SQL cho DB2 LUW 9.7. Nó cũng hỗ trợ PL/SQL cho Oracle 10g và 11g. Để làm việc với PL/SQL, hãy sử dụng phối cảnh dữ liệu ODS.

Bảng 1. Chức năng PL/SQL mới

Xem—Kiểu kết nối	Chức năng mới
Trình thám hiểm nguồn dữ liệu — DB2 LUW 9.7	<ul style="list-style-type: none">• Hiển thị một gói PL/SQL trong thư mục PL/SQL Packages của một lược đồ. Các gói này không mở rộng để hiển thị nội dung của chúng.• Hiển thị một thủ tục hoặc hàm PL/SQL trong thư mục của một lược đồ.• Xem một gói, thủ tục hoặc hàm PL/SQL trong

khung nhìn Properties.

- So sánh hai đối tượng PL/SQL cùng kiểu.
- Chạy một thủ tục hoặc hàm PL/SQL, dù nó là độc lập hay trong một gói PL/SQL và xem các kết quả của nó trong khung nhìn SQL Results.
- Hiển thị một gói PL/SQL trong thư mục PL/SQL Packages của một lược đồ. Các gói này không mở rộng để hiển thị nội dung của chúng.
- Hiển thị một thủ tục hoặc hàm PL/SQL trong thư mục của một lược đồ.
- Xem một gói, thủ tục hoặc hàm PL/SQL trong khung nhìn Properties.
- So sánh hai đối tượng PL/SQL cùng kiểu.
- Chạy một thủ tục hoặc hàm PL/SQL, dù nó là độc lập hay trong một gói PL/SQL và xem các kết quả của nó trong khung nhìn SQL Results.

Trình thám hiểm nguồn
dữ liệu — Oracle 10g
hoặc 11g

Trình thám hiểm dự án
dữ liệu — DB2 LUW
9.7

- Tạo một gói PL/SQL.
- Tạo một thủ tục PL/SQL độc lập.
- Tạo một hàm PL/SQL độc lập.
- Triển khai các đối tượng PL/SQL trong một thư

mục.

- Thử nghiệm một thủ tục hoặc hàm PL/SQL bằng cách chạy nó và xem các kết quả của nó trong khung nhìn SQL Results.
- Gỡ rỗi một thủ tục hoặc hàm trong một gói PL/SQL.
- Gỡ rỗi một thủ tục PL/SQL độc lập.
- Kéo hoặc sao chép một gói hoặc thường trình PL/SQL từ một kết nối Oracle hoặc DB2 trong Data Source Explorer vào dự án của bạn.
- Kéo hoặc sao chép một gói hoặc thường trình PL/SQL từ một dự án vào trong một dự án khác.
- Tạo một gói PL/SQL.
- Tạo một thủ tục PL/SQL độc lập.
- Tạo một hàm PL/SQL độc lập.
- Triển khai các đối tượng PL/SQL trong một thư mục.
- Thử nghiệm một thủ tục hoặc hàm PL/SQL bằng cách chạy nó và xem các kết quả của nó trong khung nhìn SQL Results.
- Gỡ rỗi một thủ tục hoặc hàm trong một gói

Trình thám hiểm dự án
dữ liệu — Oracle 10g
hoặc 11g

PL/SQL.

- Gỡ rỗi một thủ tục PL/SQL độc lập.
- Kéo hoặc sao chép một gói hoặc thường trình PL/SQL từ một kết nối Oracle trong Data Source Explorer vào dự án của bạn.
- Kéo hoặc sao chép một gói hoặc thường trình PL/SQL từ một dự án vào trong một dự án khác.
- Xem các đầu vào tham số, các đầu ra tham số và trả về các tập kết quả sau khi chạy thành công hay gỡ rỗi một thường trình PL/SQL.
- Xem thông báo lỗi sau khi triển khai, chạy, gỡ rỗi hoặc loại bỏ không thành công một đối tượng từ cơ sở dữ liệu.

Khung nhìn kết quả SQL (SQL Results)

- Soạn thảo đặc tả trong tab Specification.

Trình soạn thảo gói PL/SQL

- Mô tả hoặc đổi tên gói PL/SQL để cập nhật nó trong Data Project Explorer.
- Soạn thảo phần thân trong tab Body.

Trình soạn thảo thủ tục PL/SQL

- Soạn thảo mã nguồn trong tab Source.
- Mô tả hoặc đổi tên thủ tục để cập nhật nó trong

Data Project Explorer.

Trình soạn thảo hàm
PL/SQL

- Soạn thảo các thuộc tính mô hình trong tab Source.
- Soạn thảo mã nguồn trong tab Source.
- Mô tả hoặc đổi tên hàm để cập nhật nó trong Data Project Explorer.
- Soạn thảo các thuộc tính mô hình trong tab Configuration.

Kết nối tới một cơ sở dữ liệu tương thích PL/SQL trong DB2 LUW 9.7

Để tạo một cơ sở dữ liệu có tên là *cdbname* với tính tương thích PL/SQL, hãy nhập các câu lệnh sau trong một cửa sổ lệnh DB2:

```
DB2SET DB2_COMPATIBILITY_VECTOR=FFF
```

```
DB2STOP && DB2START
```

```
DB2 CREATE DATABASE cdbname
```

Đối với cơ sở dữ liệu *cdbname*, DB2 hỗ trợ các kiểu dữ liệu PL/SQL-cụ thể. Nếu bạn kết nối cơ sở dữ liệu *cdbname* với cửa sổ lệnh DB2 và nhận được cấu hình cơ sở dữ liệu, bạn có thể xác nhận rằng các giá trị cài đặt tương thích sau đang có bằng cách sử dụng các lệnh sau đây (trên Linux sử dụng grep thay cho findstr):

DB2 CONNECT TO *cdbname*

DB2 GET DB CFG | findstr compatibility

Number compatibility = ON

Varchar2 compatibility = ON

Date compatibility = ON

DB2 DISCONNECT *cdbname*

Phím tắt của trình đơn hiện lên

Trong hầu hết trường hợp, bạn có thể sử dụng phím tắt **Shift+F10** để nhìn thấy trình đơn hiện lên cho khung nhìn hiện tại hoặc lựa chọn. Điều này có thể được sử dụng như là một sự thay thế cho việc nhấn chuột phải.

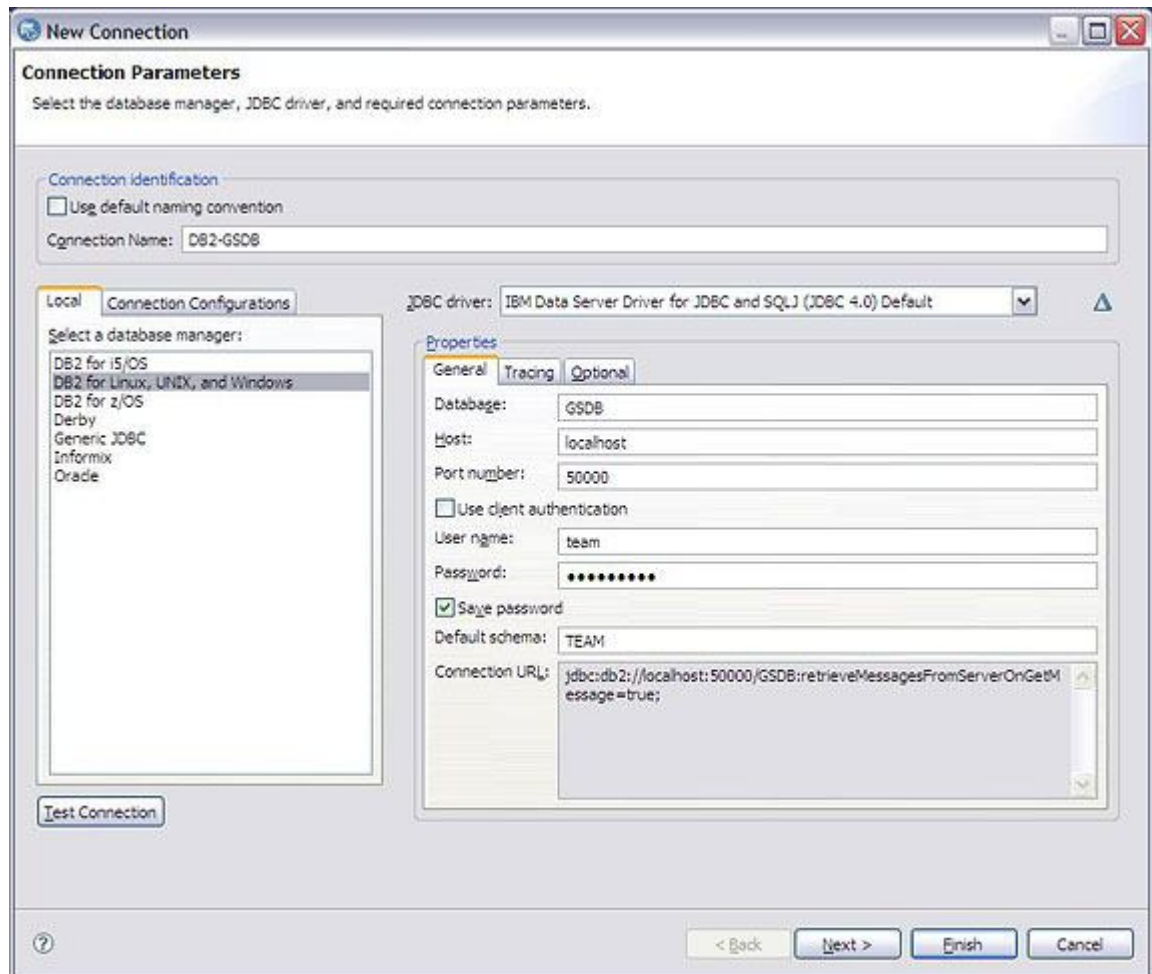
Để tạo một Hiện trạng kết nối cơ sở dữ liệu cho cơ sở dữ liệu mẫu GSDB có trong DB2 LUW 9.7, hãy làm như sau:

1. Từ Data Source Explorer, nhấn vào nút thanh công cụ Hiện trạng kết nối mới (New Connection Profile):



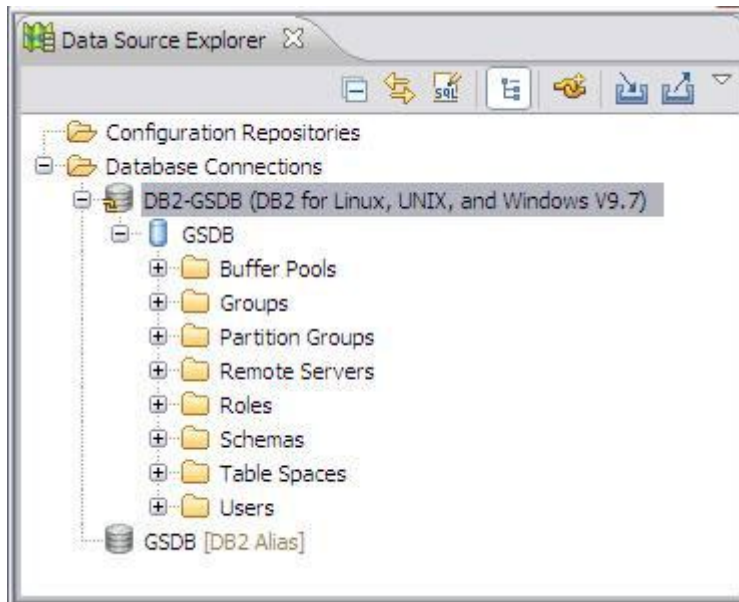
Hoặc, nhấn chuột phải vào thư mục Các kết nối cơ sở dữ liệu (Database Connections) và chọn **New...** từ trình đơn bật lên. Cái nào trong các hành động này cũng làm cho trình thủ thuật kết nối mới (New Connection) xuất hiện, như trong Hình 1.

Hình 1. Tạo một kết nối đến GSDB trên DB2 9.7



2. Chọn **DB2 for Linux, UNIX, and Windows** cho trình quản lý cơ sở dữ liệu của bạn.
3. Chọn trình điều khiển **IBM Data Server Driver for JDBC and SQLJ (JDBC 4.0) Default** và chỉ định tên cơ sở dữ liệu của bạn, tên máy chủ lưu trữ, số cổng kết nối, tên người dùng và mật khẩu người dùng. Định nghĩa trình điều khiển cho trình điều khiển IBM JDBC được định sẵn và các tệp JAR của nó được đóng gói chung với ODS 2.2.
4. Nhấn vào Kết thúc (**Finish**). Hiện trạng kết nối xuất hiện trong Data Source Explorer, như trong Hình 2.

Hình 2. Duyệt hiện trạng kết nối DB2-GSDB mới trong Data Source Explorer



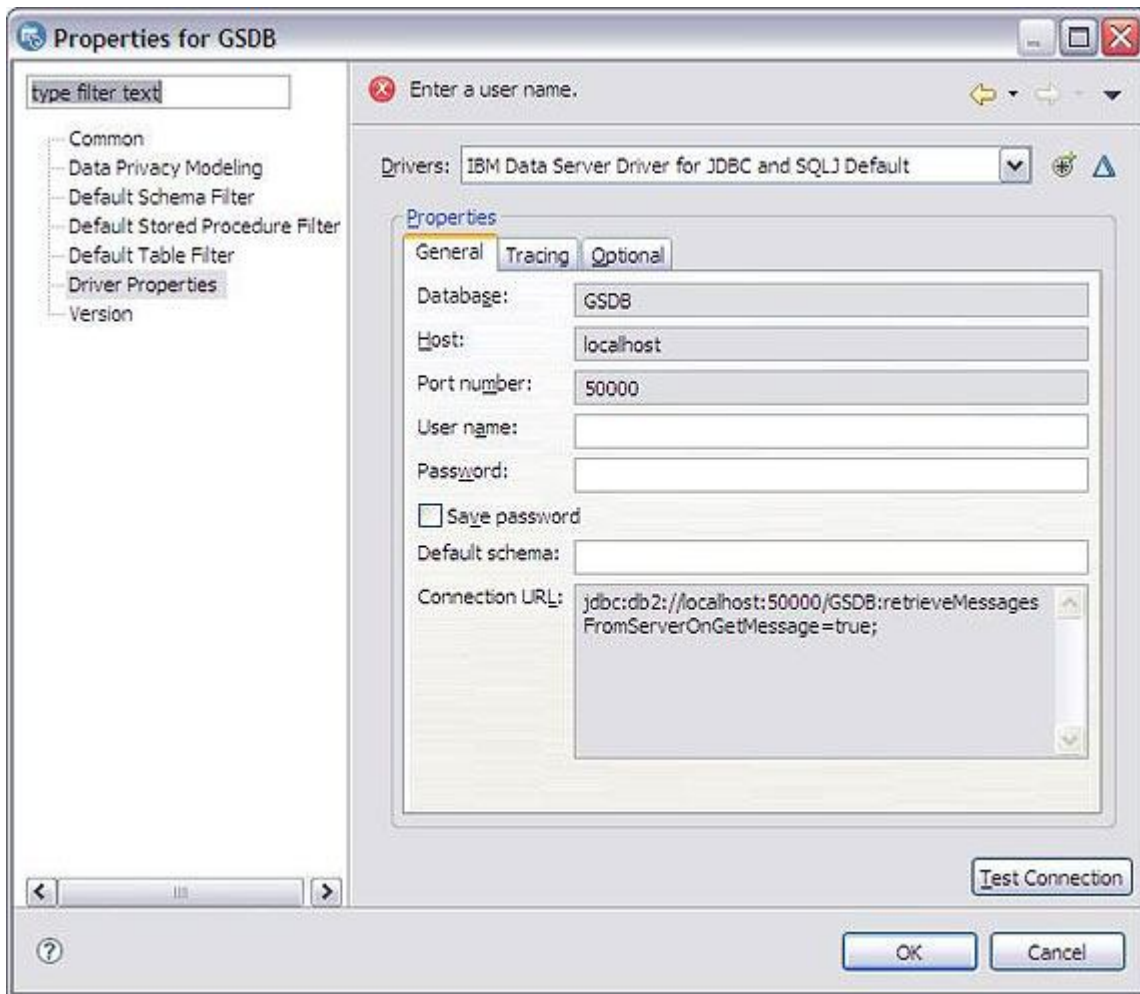
Chú ý rằng hiện trạng kết nối DB2-GSDB thể hiện trong Hình 2 được tiếp tục theo một kết nối có tên GSDB [DB2 Alias]. Nếu bạn đã cài đặt một máy khách DB2 và đã bổ sung một bí danh cơ sở dữ liệu DB2 bằng cách tạo ra cơ sở dữ liệu tại chỗ hay sử dụng Trợ giúp cấu hình (Configuration Assistant), thì bí danh cơ sở dữ liệu tự động xuất hiện trong Data Source Explorer khi bạn tạo hoặc mở một vùng làm việc.

Vì phiên bản cơ sở dữ liệu bị ẩn đi đối với người sử dụng các bí danh DB2, cấu hình bí danh được giả định là một kết nối tới DB2 LUW 9.1. Như vậy, ODS 2.2 không hiển thị cho bạn các thư mục hoặc hành động PL/SQL khi bạn tạo một dự án phát triển dữ liệu có liên kết với nó. Tuy nhiên, khi bạn chỉ định một ID và mật khẩu của người dùng có thẩm quyền cho cấu hình này và kết nối tới cơ sở dữ liệu, ODS 2.2 phát hiện ra nền tảng và phiên bản DB2 thực tế. Nếu nền tảng và phiên

bản thực tế là DB2 LUW và 9.7, sau đó ODS 2.2 cho bạn thấy các thư mục và các hành động PL/SQL.

Khi lần đầu tiên bạn cố gắng kết nối đến một bí danh DB2, ODS 2.2 sẽ nhắc bạn nhập một tên và mật khẩu người dùng, như trong Hình 3.

Hình 3. Nhập tên và mật khẩu người dùng để kết nối



Kết nối đến Oracle 10g hoặc 11g

ODS không đóng gói một trình điều khiển JDBC để kết nối với một máy chủ cơ sở dữ liệu Oracle. Vì thế, trước tiên bạn phải lấy được một trình điều khiển JDBC để hỗ trợ Oracle từ Oracle hoặc DataDirect. Các trình điều khiển JDBC này là các sản phẩm lập trình được cấp phép mà bạn phải trả tiền hoặc sử dụng có giấy phép.

Là một nhà phát triển của Oracle, bạn có thể đăng ký vào trang Web mạng công nghệ của Oracle, chấp nhận Hiệp định cấp phép phân phối và phát triển OTN và tải một trình điều khiển Oracle JDBC. Xem Tài nguyên để có một liên kết tới trang tải về trình điều khiển Oracle JDBC.

Oracle cung cấp các trình điều khiển JDBC mỏng và các trình điều khiển JDBC của Giao diện gọi Oracle (Oracle Call Interface-OCI). Các trình điều khiển OCI yêu cầu một máy khách Cơ sở dữ liệu Oracle (hoặc bạn có thể tải các thư viện cần thiết tương đương).

Kết nối đến Oracle bằng cách sử dụng một trình điều khiển JDBC mỏng của Oracle

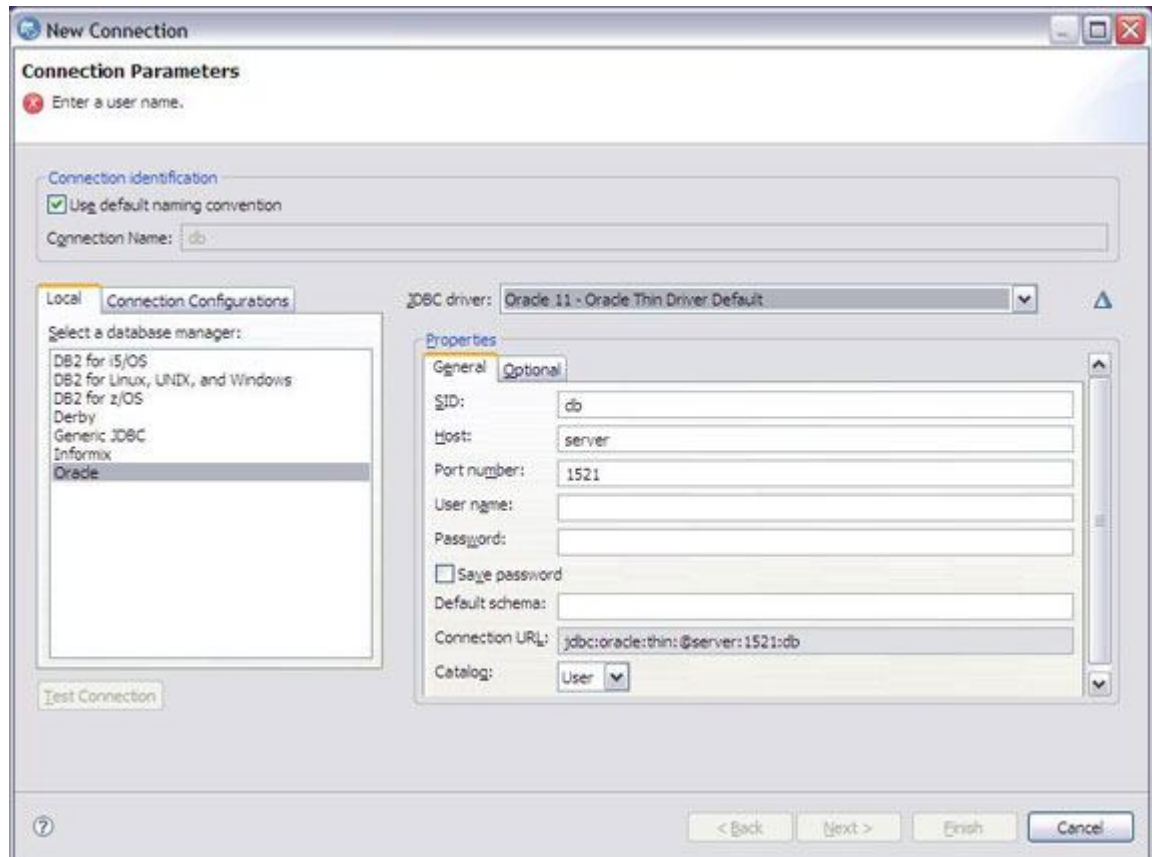
Oracle đã cung cấp một trình điều khiển JDBC mỏng mới cho các mức JRE từ 1.1.1. Bạn luôn luôn phải sử dụng trình điều khiển mới nhất cho các JRE có sẵn. Hiện nay, JRE mới nhất là 1.6 và JAR của nó là ojdbc6.jar. Phiên bản Eclipse mà ODS 2.2 mở rộng được xây dựng trên JRE 1.6.

Thực hiện theo các bước sau để kết nối với Oracle khi sử dụng trình điều khiển JDBC mỏng của Oracle:

1. Lấy về tệp Ojdbc6.jar và sao chép nó vào ổ cứng của bạn trong một thư mục như là C:\jdbcDrivers\Oracle.
2. Trong Data Source Explorer, nhấn chuột phải vào thư mục Các kết nối cơ sở dữ liệu (Database Connections) và chọn **New...** từ trình đơn hiện lên.

3. Trình thủ thuật New Connection sẽ xuất hiện như trong Hình 4.

Hình 4. Chọn một trình điều khiển JDBC mớng của Oracle



Ở đây bạn phải lựa chọn Oracle cho trình quản lý cơ sở dữ liệu và bạn có thể chọn một trong các trình điều khiển mớng, như **Oracle 11 - Oracle Thin Driver Default**.

Điền vào các thuộc tính chung để nhận biết máy chủ cơ sở dữ liệu Oracle của bạn.

4. Nhấn vào biểu tượng Soạn thảo danh sách Jar... (**Edit Jar List...**) bên phải của trình điều khiển JDBC:

Hộp thoại Edit Jar List xuất hiện, như trong Hình 5.

Hình 5. Xác định vị trí ojdbc6.jar

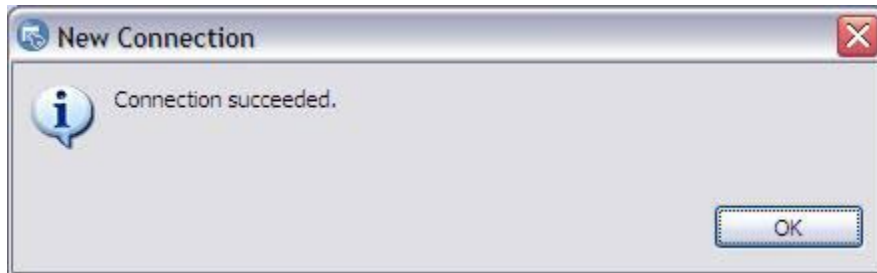


Theo mặc định tệp Ojdbc6.jar được liệt kê là một trình giữ chỗ (placeholder), bạn cần phải xác định đường dẫn đầy đủ của nó (ví dụ: C:\jdbcDrivers\Oracle\ojdbc6.jar).

5. Chọn trình giữ chỗ và nhấn vào **Edit JAR/Zip...** (Soạn thảo JAR/ZIP...). Một hộp thoại phân tệp xuất hiện.
6. Chuyển hướng đến bản sao ojdbc6.jar của bạn, chọn nó và nhấn **Open** (Mở). Đường dẫn xuất hiện trong hộp thoại Edit Jar List.
7. Nhấn **OK**.

8. Quay lại trình thủ thuật New Connection, nhấn vào **Test Connection** (Kiểm tra kết nối). Một thông báo xác nhận sẽ xuất hiện, như trong Hình 6.

Hình 6. Chấp nhận xác nhận kết nối



9. Nhấn **OK**.
10. Nhấn **Finish**. Hiện trạng kết nối xuất hiện trong Data Source Explorer, như trong Hình 7.

Hình 7. Duyệt hiện trạng kết nối mới của Oracle trong Data Source Explorer



Kết nối Oracle bằng cách sử dụng trình điều khiển JDBC kết nối của DataDirect

Thực hiện theo các bước sau để kết nối đến Oracle khi sử dụng trình điều khiển JDBC kết nối của DataDirect:

1. Hãy lấy về trình điều khiển JDBC kết nối đầy đủ của DataDirect hoặc trình điều khiển JDBC 4.0 kết nối của DataDirect cho riêng Oracle.

2. Nếu bạn có trình điều khiển JDBC 3.7 kết nối đầy đủ của DataDirect:

- a. Giải nén tệp connectjdbc.jar vào một thư mục mới tên là installdd. Tệp này sẽ cho bạn một giấy phép 15 ngày.

Lưu ý: Không tìm cách gia hạn giấy phép khi sử dụng LicenseTool.jar trong thư mục cài đặt. Các nỗ lực như vậy sẽ không thành công và làm hỏng.

- b. Từ thư mục installdd, chạy Trình cài đặt (Installer) và chọn một thư mục mà bạn muốn cài đặt trình điều khiển ở đó (ví dụ: thư mục C:\jdbcDrivers\Oracle\datadirect).

3. Nếu bạn có trình điều khiển JDBC kết nối của DataDirect chỉ cho Oracle, bạn có thể sử dụng tệp oracle.jar "như hiện tại" phụ thuộc vào các điều khoản sử dụng tùy theo giấy phép của bạn.

4. Trong ODS, mở **Window** -> **Preferences** và chọn **Data Management** -> **Connectivity** -> **Driver Definitions**.

5. Nhấn **Add...**

6. Các bảng sau đây cung cấp chi tiết về cách điền vào các trường trên hộp thoại New Driver Definition (Định nghĩa trình điều khiển mới), tùy theo cơ sở dữ liệu và trình điều khiển Oracle nào mà bạn đang sử dụng.

Tên/Kiểu:

	Oracle 10	Oracle 11
Bộ lọc Kiểu trình điều khiển	Cơ sở dữ liệu	Cơ sở dữ liệu
Bộ lọc nhà cung cấp	Oracle	Oracle
Các khuôn mẫu của trình điều khiển có sẵn	Trình điều khiển khác / Oracle / 10	Trình điều khiển khác / Oracle / 11
Tên trình điều khiển	Kết nối DataDirect cho Oracle 10	Kết nối DataDirect cho Oracle 11
Kiểu trình điều khiển	Trình điều khiển khác	Trình điều khiển khác

Các tệp trình điều khiển danh sách Jar:

**Trình điều khiển DataDirect
Connect 3.7 đầy đủ**

**Trình điều khiển cho riêng Oracle
(4.0)**

C:\jdbcDrivers\Oracle\datadirect\lib\ba
se.jar

C:\jdbcDrivers\Oracle\datadirect\ora
cle.jar

C:\jdbcDrivers\Oracle\datadirect\lib\util.jar
C:\jdbcDrivers\Oracle\datadirect\lib\oracle.jar

Các đặc tính:

Oracle 10

Oracle 11

Danh
mục TẤT CẢ (ALL)

TẤT CẢ (ALL)

URL
kết
nối jdbc:datadirect:oracle://habu.svl
.ibm.com:1521;SID=ora10g

jdbc:datadirect:oracle://chex.svl
.ibm.com:1521;SID=ORA11

Tên
cơ sở
dữ
liệu oracle

oracle

Lớp
trình
r com.ddtek.jdbc.oracle.OracleDriver
r

điều
khiển

Mật
khẩu
(tùy
chọn)

Mã
nhận
dạng

người
i

CRIOLLO1

CRIOLLO1

dùng
(tùy
chọn)

Lưu ý: Bạn có thể đặt SID hoặc vào URL kết nối, như ở trên, hoặc thêm nó vào như là một thuộc tính bổ sung trên tab Optional.

7. Sau khi bạn thực hiện tốt (OK) mọi thứ, bạn có thể tạo một hồ sơ kết nối mới với Oracle bằng cách sử dụng định nghĩa trình điều khiển này từ Data Source Explorer khi sử dụng Trình thủ thuật New Connection.

Các khung nhìn danh mục Oracle

Khi bạn tạo một kết nối của Oracle, bạn có thể chọn một trong ba khung nhìn danh mục để lọc những gì bạn nhìn thấy trong Data Source Explorer của chủ sở hữu:

All

Khung nhìn này mô tả tất cả các đối tượng có thể truy cập đến bạn.

DBA

Khung nhìn này mô tả tất cả các đối tượng trong cơ sở dữ liệu.

User

Khung nhìn này mô tả các đối tượng mà bạn sở hữu.

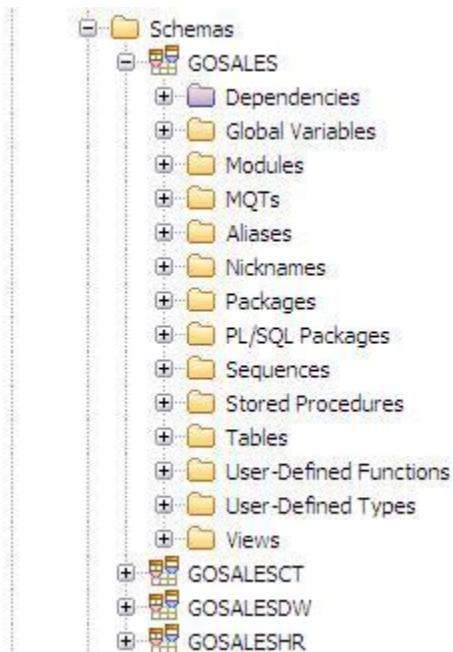
Duyệt cơ sở dữ liệu và các đối tượng của bạn

Sau khi bạn tạo một kết nối cơ sở dữ liệu, hiện trạng kết nối của nó xuất hiện trong Data Source Explorer.

Mở rộng kết nối DB2 đến GSDB để xem các thư mục chính của nó. Hầu hết các thông tin mà bạn quan tâm có trong thư mục Các lược đồ (Schemas). Mở rộng thư mục Schemas, cuộn để di chuyển đến tên lược đồ mặc định của bạn và mở rộng lược đồ đó để xem các thư mục của nó, như trong Hình 8.

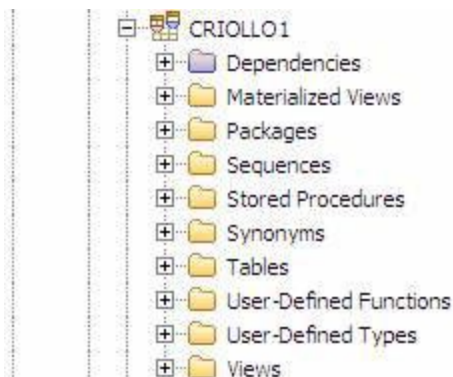
Hình 8. Mở rộng kết nối DB2 của bạn để xem các thư mục trong lược đồ

GOSALES



Mở rộng kết nối Oracle của bạn để xem các thư mục chính của nó. Mở rộng thư mục Schemas, cuộn để di chuyển đến tên lược đồ mặc định của bạn và mở rộng lược đồ đó để xem các thư mục của nó, như trong Hình 9.

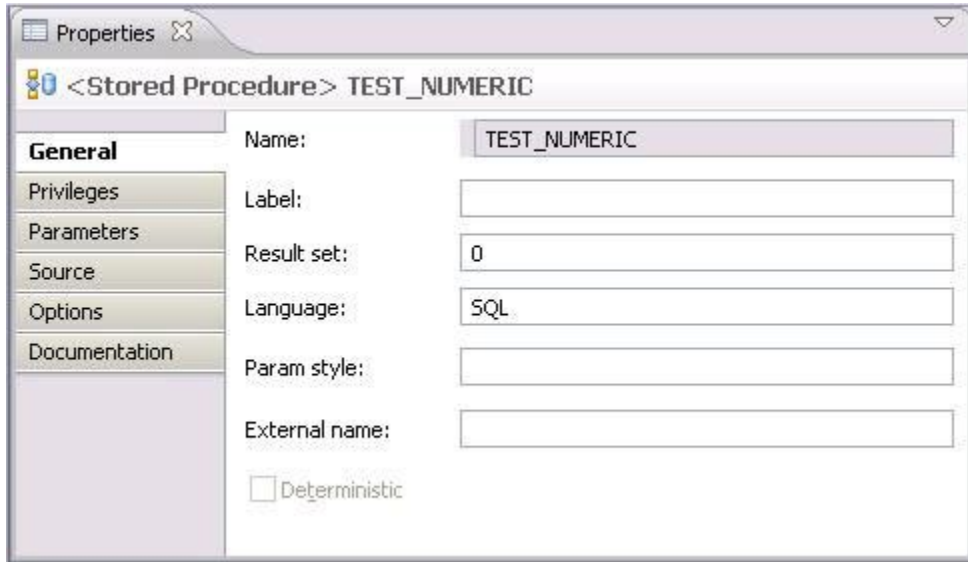
Hình 9. Mở rộng kết nối Oracle của bạn để xem các thư mục trong lược đồ CRIOLO1



Bạn có thể tìm thấy các đối tượng hiện có trong các thư mục của chúng. Khi bạn chọn một đối tượng hiện có, bạn có thể xem các thuộc tính của nó trong khung

nhìn Properties. Tab General trong khung nhìn Properties nhận biết đối tượng, như trong Hình 10.

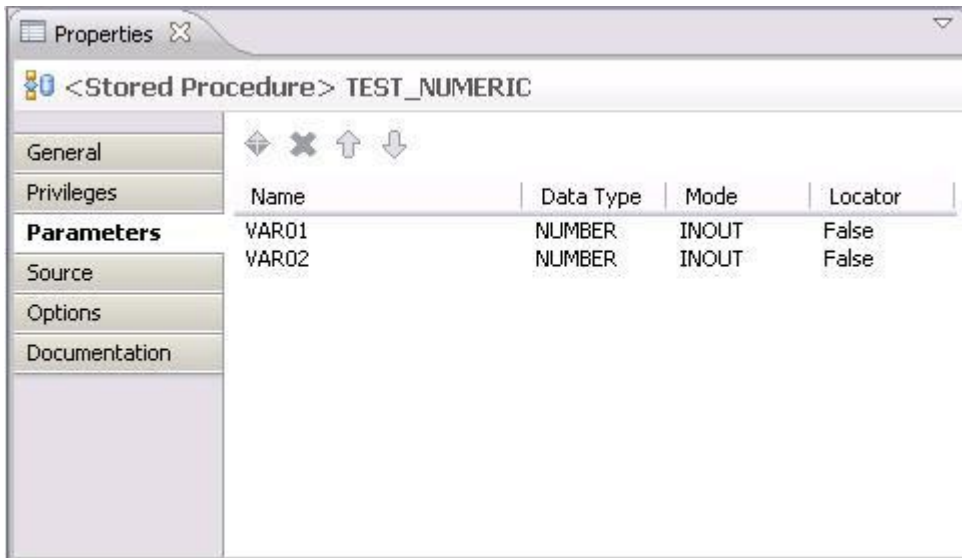
Hình 10. Duyệt các thuộc tính chung của một thủ tục lưu sẵn



Từ quan điểm của Oracle, ngôn ngữ thực hiện cho thủ tục PL/SQL là SQL, chứ không phải là PL/SQL. ODS 2.2 phân biệt giữa SQL và PL/SQL cho các mục đích phát triển.

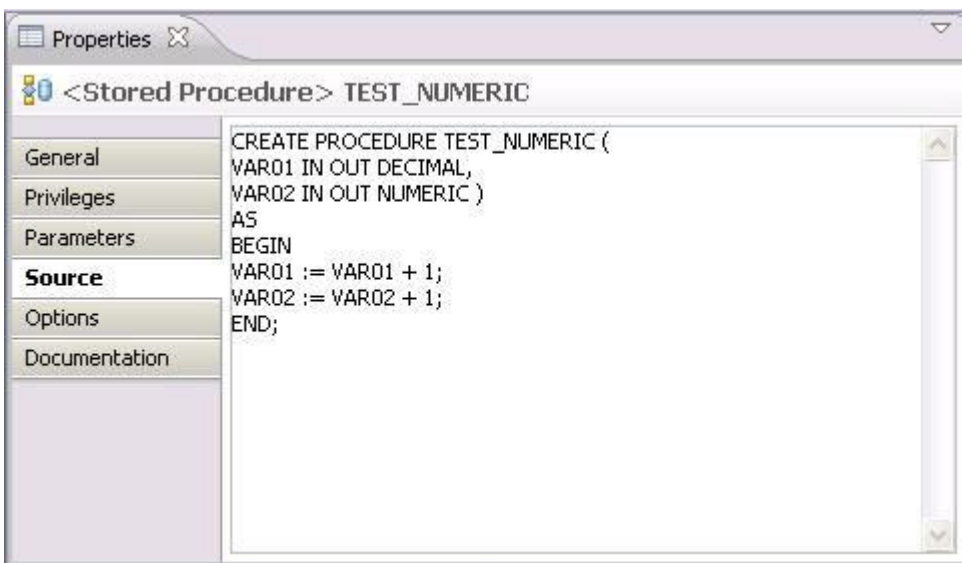
Tab Các tham số (Parameters) cho thấy các tham số đầu vào và đầu ra, như trong Hình 11.

Hình 11. Duyệt các tham số của một thủ tục lưu sẵn



Tab Nguồn (Source) cho thấy thủ tục này chỉ ra câu lệnh CREATE, như trong Hình 12.

Hình 12. Duyệt mã nguồn của một thủ tục lưu sẵn



Làm việc ngoại tuyến

Phối cảnh Dữ liệu (Data) được thiết kế để giúp bạn làm việc với các cơ sở dữ liệu của bạn; tuy nhiên, bạn vẫn có thể làm một số công việc khi bạn đang đi du lịch hoặc mạng bị ngắt. Cách làm này được gọi là *làm việc ngoại tuyến*.

Để lưu trữ trong bộ nhớ truy cập nhanh các thông tin danh mục cơ sở dữ liệu trong vùng làm việc của bạn để làm việc ngoại tuyến, hãy chọn những hành động này từ trình đơn hiện lên bằng cách nhấn chuột phải trên hiện trạng kết nối:

1. **Kết nối.**
2. **Lưu trữ ngoại tuyến.**
3. **Ngắt kết nối.**
4. **Làm việc ngoại tuyến.**

Bạn có thể biết bạn đang làm việc ngoại tuyến hay không khi trình đơn hiện lên về hiện trạng kết nối cho phép bạn kết nối hoặc ngắt kết nối, nhưng không cho phép bạn làm việc ngoại tuyến hoặc lưu trữ ngoại tuyến.

Khi bạn đang làm việc ngoại tuyến, bạn có thể:

- Mở rộng cơ sở dữ liệu để xem tất cả các đối tượng trong lược đồ của bạn và các thuộc tính của chúng.
- Tạo một dự án phát triển dữ liệu cho kết nối cơ sở dữ liệu đó.
- Kéo và thả (hoặc sao chép và dán) các gói và các thường trình PL/SQL từ Data Source Explorer tới dự án của bạn trong Data Project Explorer.
- Tạo các gói và các thường trình PL/SQL mới trong dự án của bạn.

- Soạn thảo các đối tượng trong dự án của bạn.

Tuy nhiên, nếu bạn chọn một hành động trong dự án có yêu cầu một kết nối, như Chạy (**Run**) hoặc Triển khai (**Deploy**), bạn có thể được nhắc tạo kết nối. Nếu bạn không chọn, thì hành động sẽ không hoàn thành.

Như bạn có thể nhìn thấy trong câu lệnh CREATE ở trên, thủ tục TEST_NUMERIC nhận một số thập phân DECIMAL và số NUMERIC và trả về các giá trị này được tăng thêm 1.

Bằng cách so sánh mã nguồn với các tham số, bạn cũng có thể thấy rằng mặc dù kiểu dữ liệu tham số VAR01 được định nghĩa là DECIMAL, những máy chủ cơ sở dữ liệu Oracle đã chuẩn hoá số thập phân này thành NUMBER.

Một NUMBER của Oracle có thể là một giá trị số nguyên, dấu phẩy động hoặc thập phân tùy thuộc vào độ chính xác và thang đo tùy chọn của nó.

Phát triển các đối tượng PL/SQL với Oracle

Các quá trình phát triển các gói và các thường trình PL/SQL với Optim Development Studio tương tự như các quá trình phát triển các gói và các thường trình DB2.

Do Optim Development Studio 2.2 hỗ trợ PL/SQL cho cả Oracle và cả DB2, nên bạn có thể dễ dàng chuyển giao công việc từ Oracle đến DB2 bằng cách kéo các đối tượng từ một dự án của Oracle trong Data Project Explorer hay từ một cơ sở dữ liệu Oracle trong Data Source Explorer, sang một dự án DB2 trong Data Project Explorer.

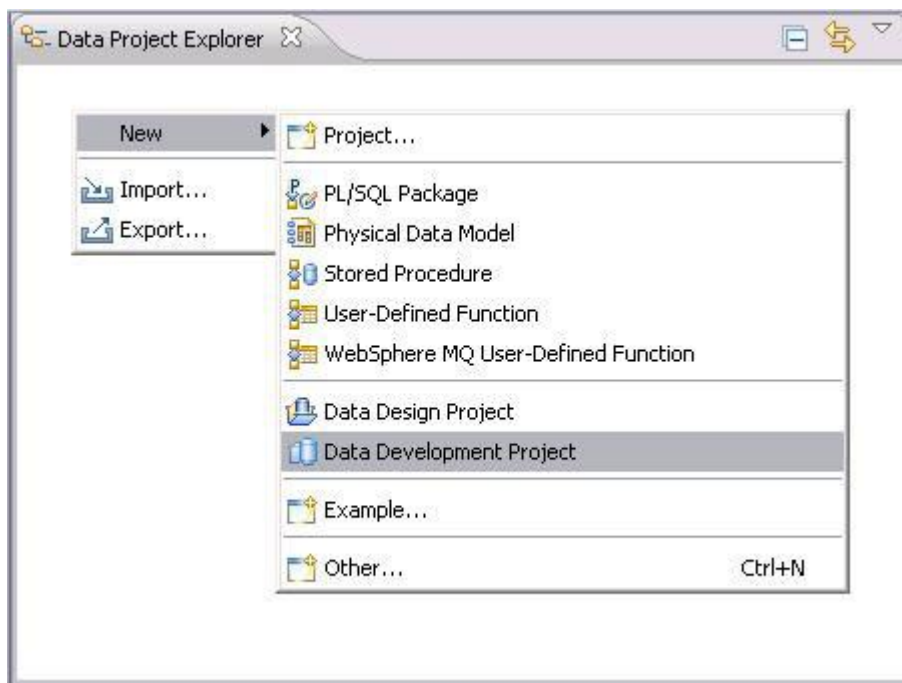
Tạo một dự án phát triển dữ liệu Oracle

Một dự án phát triển dữ liệu có chứa một tập các kịch bản lệnh SQL, các thường trình và các nguồn tài nguyên liên quan đến XML, được liên kết với một và chỉ một hiện trạng kết nối cơ sở dữ liệu trong Data Source Explorer. Một dự án phát triển dữ liệu có liên kết với một hiện trạng kết nối DB2 LUW 9.7 được gọi là dự án DB2 và một dự án phát triển dữ liệu có liên kết với một hiện trạng kết nối Oracle 10g hoặc 11g được gọi là một dự án của Oracle.

Thực hiện theo các bước sau để tạo ra một dự án phát triển dữ liệu trong Data Project Explorer:

1. Nhấn chuột phải vào một phần trống trên nền Data Project Explorer để mở một trình đơn hiện lên. Chọn **New -> Data Development Project**, như trong Hình 13.

Hình 13. Khởi chạy một trình thủ thuật để tạo ra một dự án phát triển dữ liệu mới



Trình đơn này chỉ xuất hiện khi bạn nhấn vào trong vùng trống trong khi không có gì trong khung nhìn được chọn.

2. Trang Dự án phát triển (Development Project) của trình thủ thuật xuất hiện. Nhập tên cho dự án, như trong Hình 14 và nhấn **Next >**.

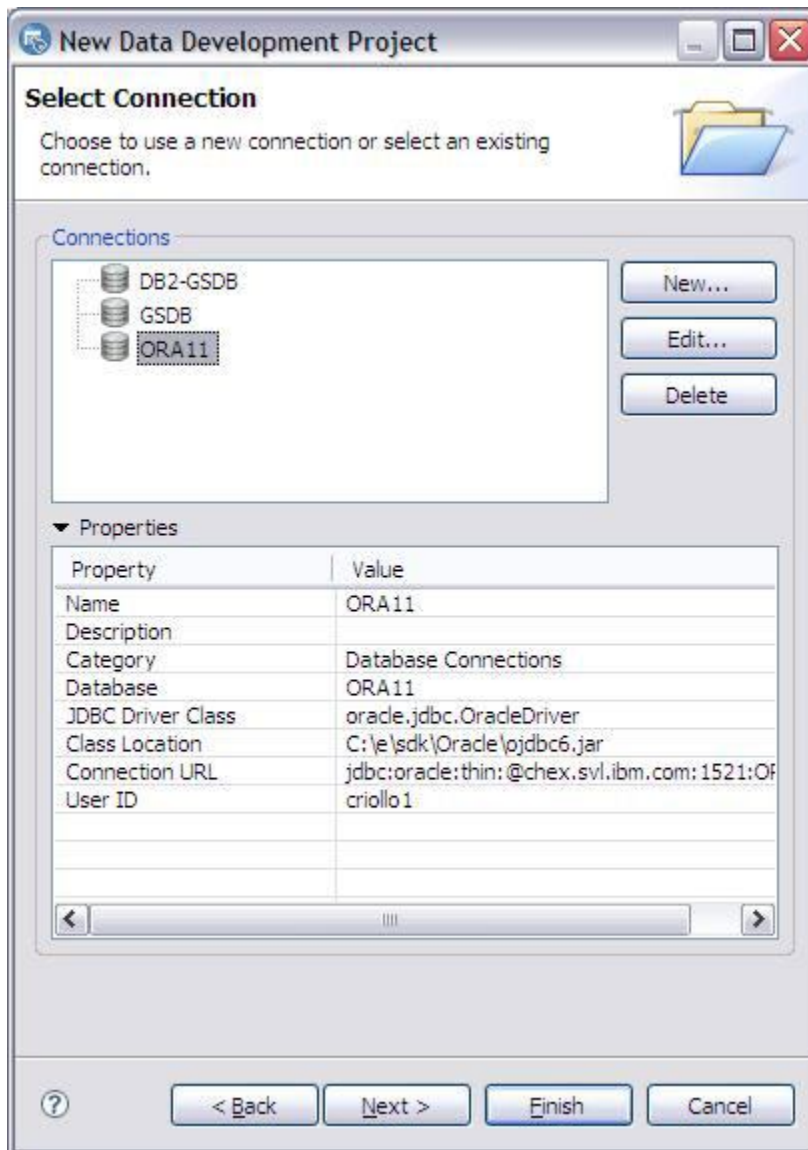
Hình 14. Nhập tên cho dự án phát triển dữ liệu của bạn



3. Từ trang Chọn kết nối (Select Connection), chọn một kết nối của Oracle hiện có, như trong Hình 15 hoặc tạo một kết nối mới.

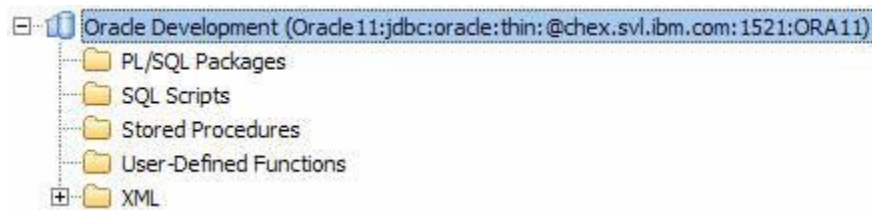
Hình 15. Chọn một hiện trạng kết nối cho dự án phát triển dữ liệu của

bạn



4. Nhấn **Finish**. Bây giờ dự án này xuất hiện trong Data Project Explorer.
5. Mở rộng dự án này. Hình 16 cho thấy một ví dụ về một dự án Oracle.

Hình 16. Mở rộng các thư mục của dự án Oracle của bạn



Mỗi thư mục chứa các tài nguyên của chỉ một kiểu. Bạn có thể nhấn chuột phải vào thư mục bất kỳ và sử dụng trình đơn hiện lên để gọi một trình thủ thuật, cho phép bạn tạo ra một nguồn tài nguyên của kiểu đó có chứa trong thư mục này.

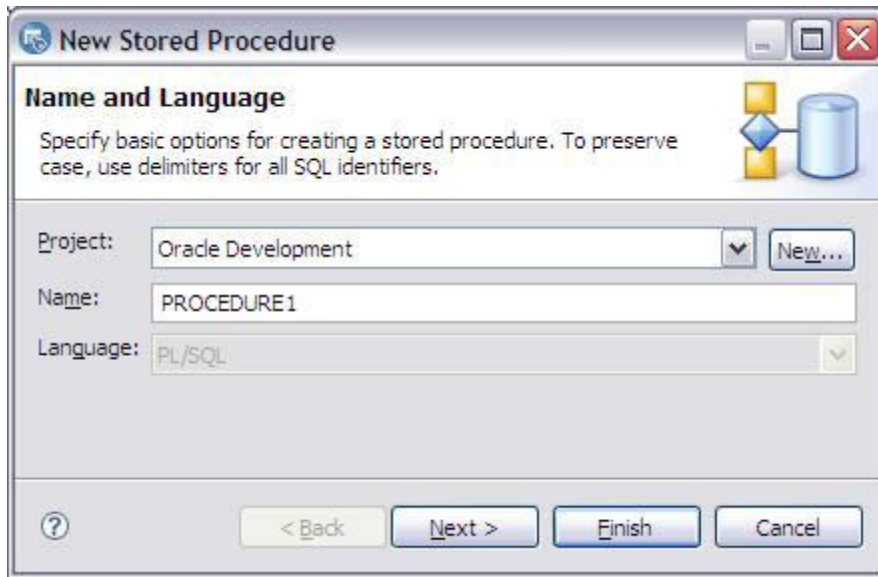
Mỗi hành động **New** trên một thư mục trong một dự án phát triển dữ liệu sẽ mở một trình thủ thuật để tạo một đối tượng kiểu ngầm định theo tên thư mục.

Tạo một thủ tục PL/SQL cho Oracle

Thực hiện các bước sau để sử dụng một trình thủ thuật Thủ tục lưu sẵn mới (New Stored Procedure), tạo một thủ tục PL/SQL:

1. Nhấn chuột phải vào thư mục Thủ tục lưu sẵn (**Stored Procedures**) của dự án phát triển Oracle. Từ trình đơn thả xuống, chọn **New -> Stored Procedure**. Trình thủ thuật New Stored Procedure xuất hiện, như trong Hình 17.

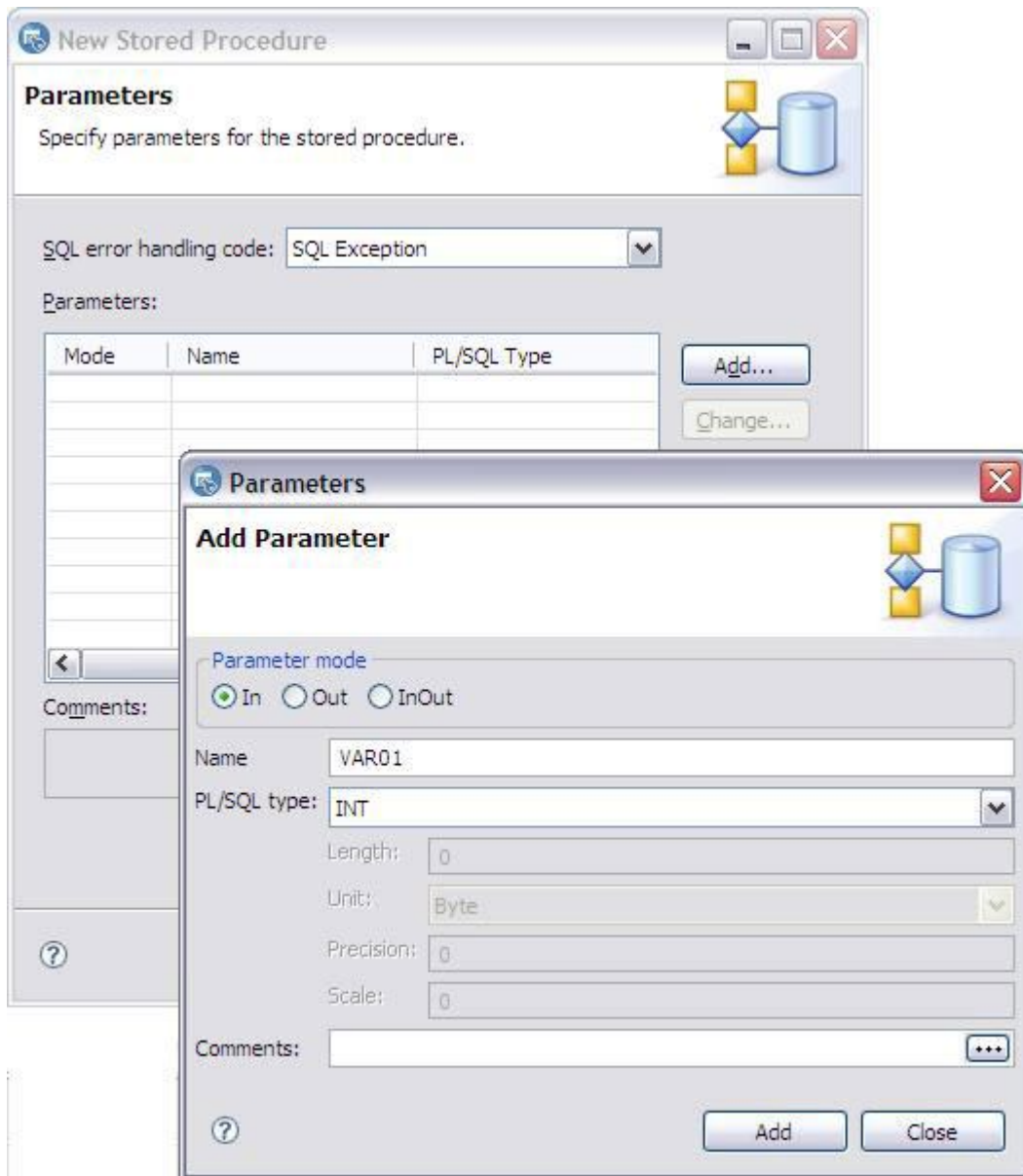
Hình 17. Đặt tên cho một thủ tục PL/SQL với Oracle



Đối với Oracle, PL/SQL là ngôn ngữ hỗ trợ duy nhất, vì thế trường Ngôn ngữ (Language) bị vô hiệu hoá.

2. Theo mặc định, giá trị cho trường Dự án (Project) là dự án có chứa sự lựa chọn của bạn. Bạn cũng có thể chọn một dự án đích khác cho thủ tục mới này hoặc sử dụng nút **New...** để tạo ra một dự án đích khác.
3. Nhập tên cho thủ tục đó. Bạn có thể mô tả tên này với một chủ sở hữu (ví dụ, CRIOLLO1.TESTPROCEDURE). Bạn cũng có thể phân định tên thủ tục cụ thể là các ký tự chữ thường và cho phép các khoảng trống và các ký tự đặc biệt (ví dụ, CRIOLLO1."Test.Procedure"). Đối với Oracle, tên này (không có chủ sở hữu) có thể dài đến 30 ký tự.
4. Nhấn **Next >** để hiển thị trang Parameters, như trong Hình 18.

Hình 18. Thêm các tham số cho một thủ tục PL/SQL với Oracle



Sử dụng **Add...** và hộp thoại Parameters để thêm các tham số của kiểu dữ liệu được hỗ trợ bất kỳ.

Về một danh sách các kiểu dữ liệu được hỗ trợ, xem Phụ lục A.

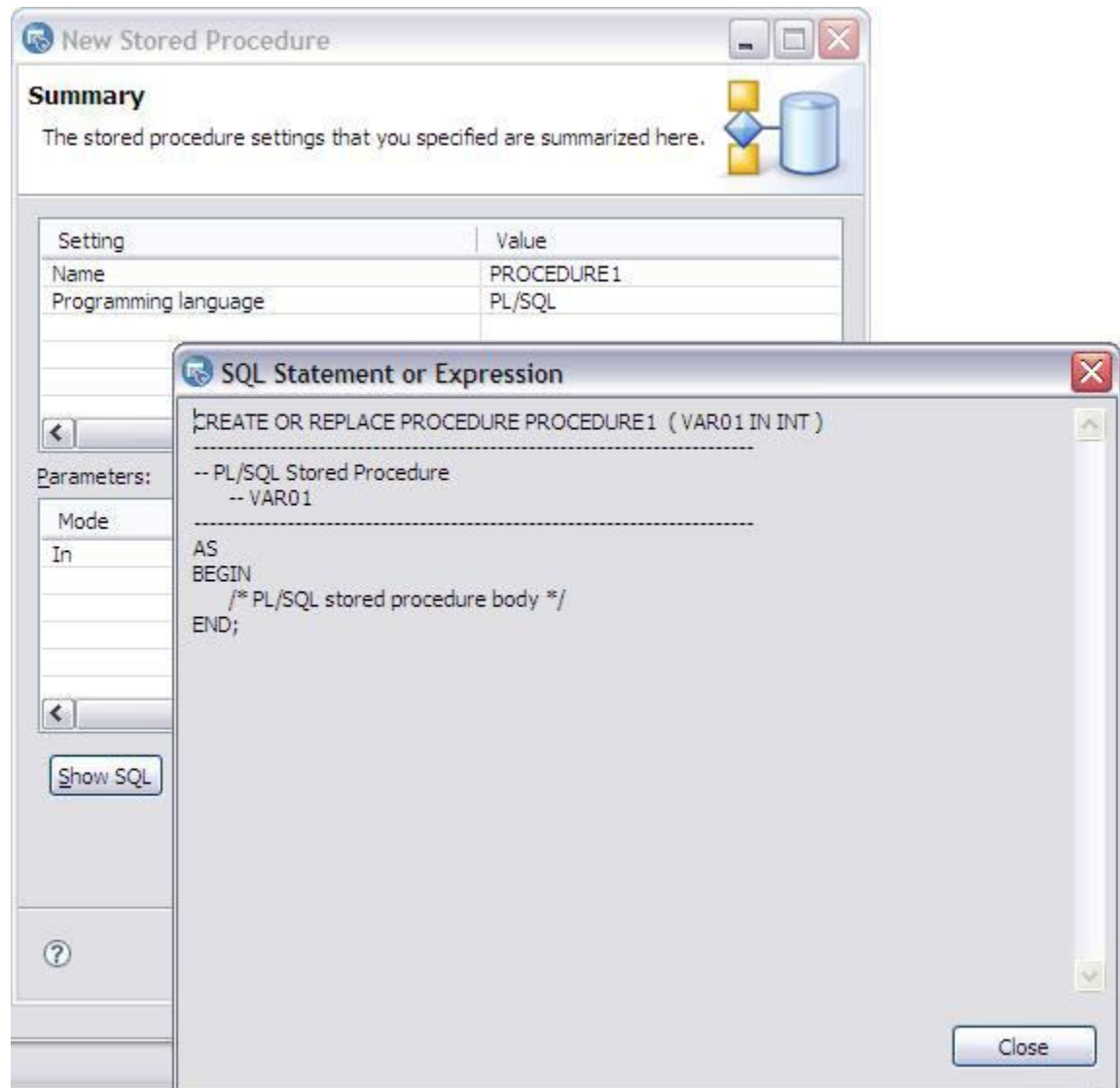
Đối với Oracle, bạn không thể xác định các độ dài, các độ chính xác hoặc các thang đo cho một kiểu dữ liệu được tham số hóa. Oracle không cho

phép các giá trị này trong các khai báo tham số (hoặc trong khai báo kiểu trả về cho một hàm). Thay vào đó, Oracle cho phép các độ dài, các độ chính xác hoặc các thang đo tối đa cho các kiểu dữ liệu này.

Các biến được khai báo trong một thường trình PL/SQL có thể gồm các độ dài, các độ chính xác hoặc các thang đo. Bạn có thể thêm chúng trong trình soạn thảo thường trình.

5. Khi bạn hoàn tất việc thêm các tham số, nhấn **Next >** để đến trang Tóm tắt (Summary).
6. Nhấn **Show SQL** để xem mã PL/SQL sẽ được tạo ra, như trong Hình 19.

Hình 19. Kiểm tra mã PL/SQL đã tạo ra cho một thủ tục của Oracle



7. Nhấn **Finish** để tạo thủ tục, thêm nó vào dự án của bạn và mở nó trong trình soạn thảo thường trình.
8. Kiểm tra thủ tục mới trong trình soạn thảo thường trình. Trước khi có thể triển khai thủ tục này, bạn cần phải thêm mã vào phần thân của nó. Việc tăng cường khả năng thử nghiệm và đơn giản là để thay đổi VAR01 thành một tham số đầu ra và để gán một giá trị cho nó, như sau:

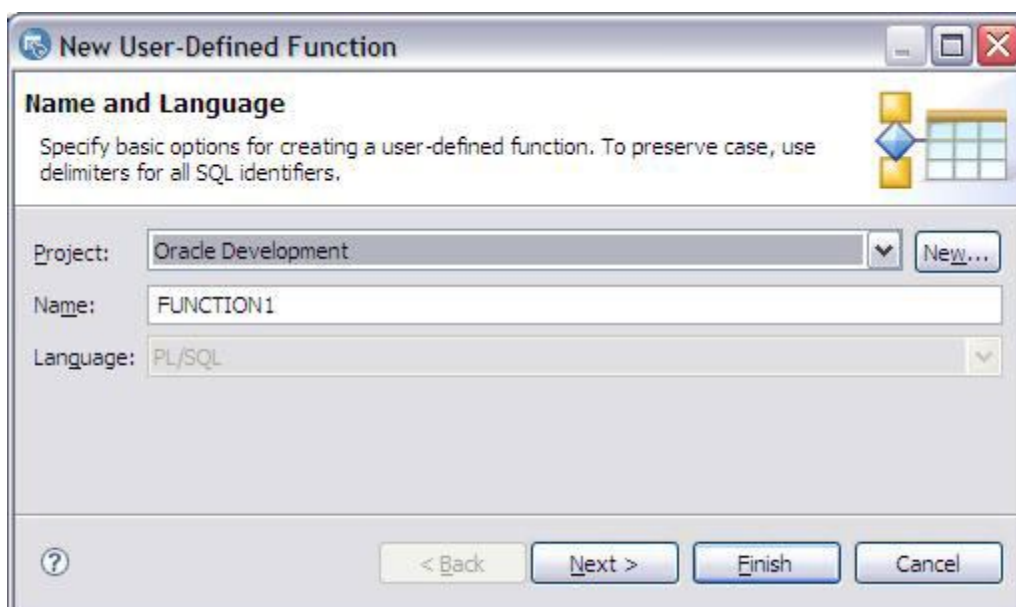
```
CREATE OR REPLACE PROCEDURE PROCEDURE1 ( VAR01 OUT INT )  
  
AS  
  
BEGIN  
  
    VAR01 := 2020;  
  
END;
```

Tạo một hàm PL/SQL với Oracle

Thực hiện theo các bước sau để sử dụng trình thủ thuật Hàm mới do người dùng định nghĩa (New User-Defined Function) để tạo một hàm PL/SQL:

1. Nhấn chuột phải vào thư mục Các hàm do người dùng định nghĩa (**User-Defined Functions**) trong dự án phát triển dữ liệu Oracle của bạn. Từ trình đơn hiện lên, chọn **New -> User-Defined Function**. Trình thủ thuật New User-Defined Function xuất hiện, như trong Hình 20.

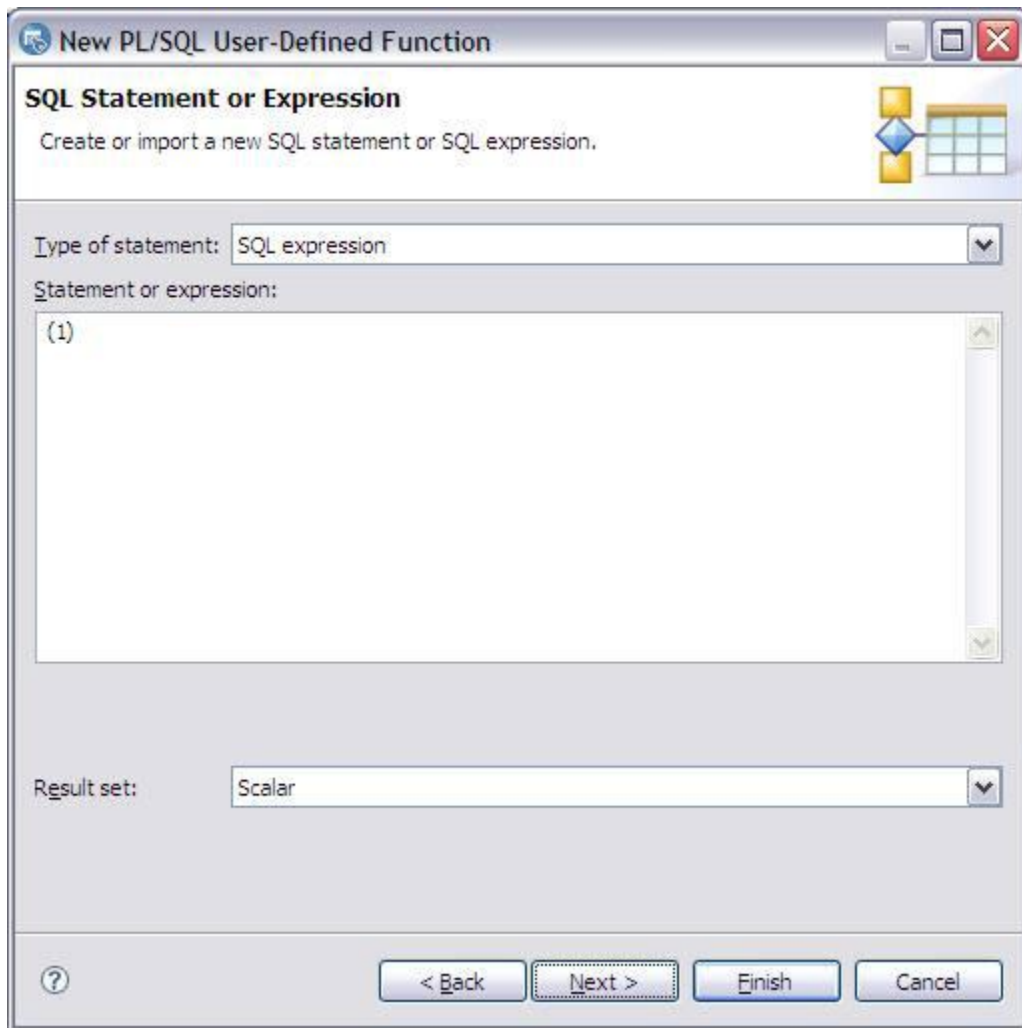
Hình 20. Đặt tên hàm PL/SQL mới với Oracle



Đối với Oracle, PL/SQL là ngôn ngữ hỗ trợ duy nhất, vì thế trường Language bị vô hiệu hóa.

2. Nhấn **Next >** để hiển thị trang câu lệnh hoặc biểu thức SQL (SQL Statement or Expression), như trong Hình 21.

Hình 21. Chỉ định một câu lệnh hay biểu thức cho một hàm Oracle

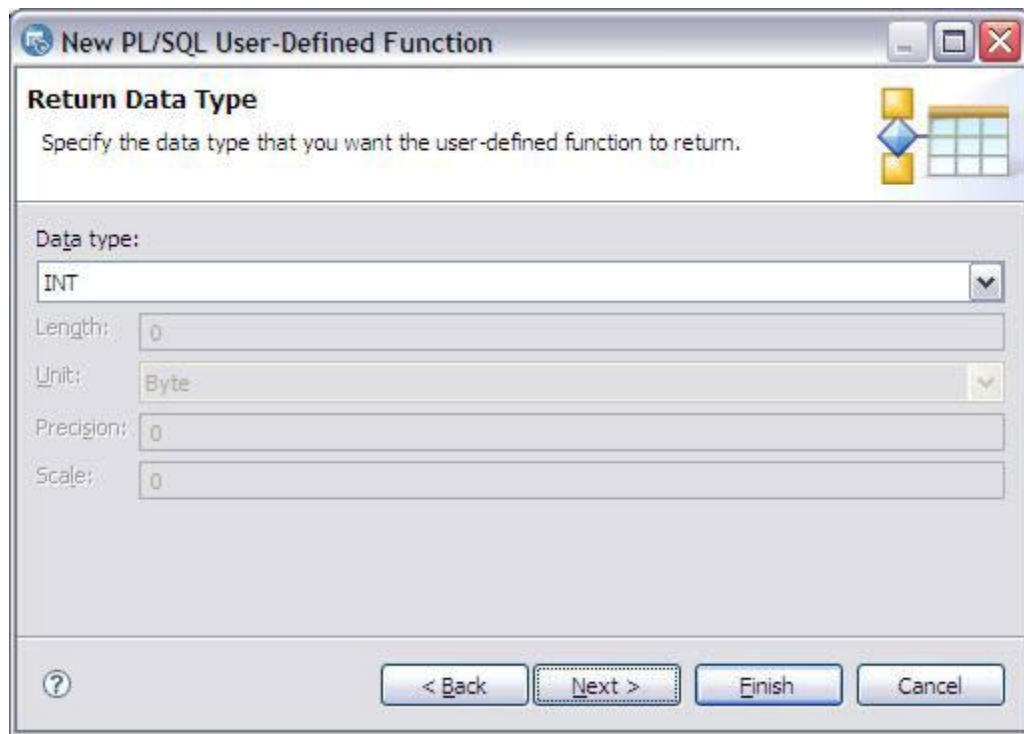


Từ trang này, bạn có thể nhập một biểu thức được hàm này trả về. Theo tùy chọn, bạn có thể không nhập vào một biểu thức ở đây và thay đổi nó sau này trong trình soạn thảo thường trình.

Đối với PL/SQL, ODS 2.2 không hỗ trợ các hàm bảng. Một hàm PL/SQL phải có một kết quả vô hướng.

3. Nhấn **Next >** để hiển thị trang Kiểu dữ liệu trả về (Return Data Type), như trong Hình 22.

Hình 22. Chỉ định kiểu trả về của hàm với Oracle

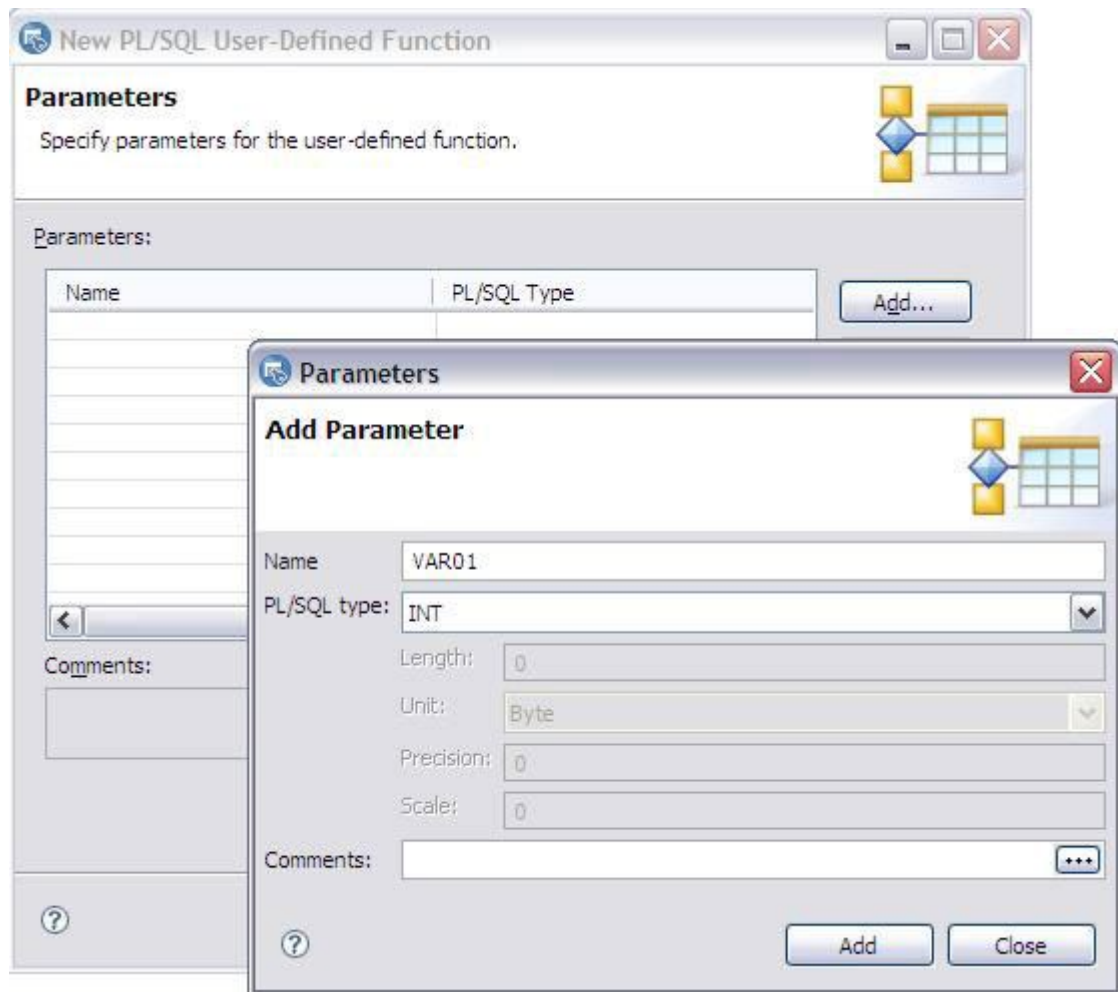


The image shows a dialog box titled "New PL/SQL User-Defined Function" with a sub-header "Return Data Type". The instruction reads: "Specify the data type that you want the user-defined function to return." The dialog contains several input fields: "Data type:" with a dropdown menu showing "INT"; "Length:" with a text box containing "0"; "Unit:" with a dropdown menu showing "Byte"; "Precision:" with a text box containing "0"; and "Scale:" with a text box containing "0". At the bottom, there are four buttons: a help icon (?), "< Back", "Next >" (which is highlighted with a dashed border), "Finish", and "Cancel".

Đối với Oracle, bạn không thể xác định độ dài, độ chính xác hoặc thang đo cho một kiểu dữ liệu trả về. Oracle không cho phép các giá trị này trong các khai báo kiểu trả về. Thay vào đó, Oracle cho phép độ dài, độ chính xác hoặc thang đo tối đa cho kiểu dữ liệu được trả về.

4. Nhấn **Next >** để hiển thị trang Parameters, như trong Hình 23.

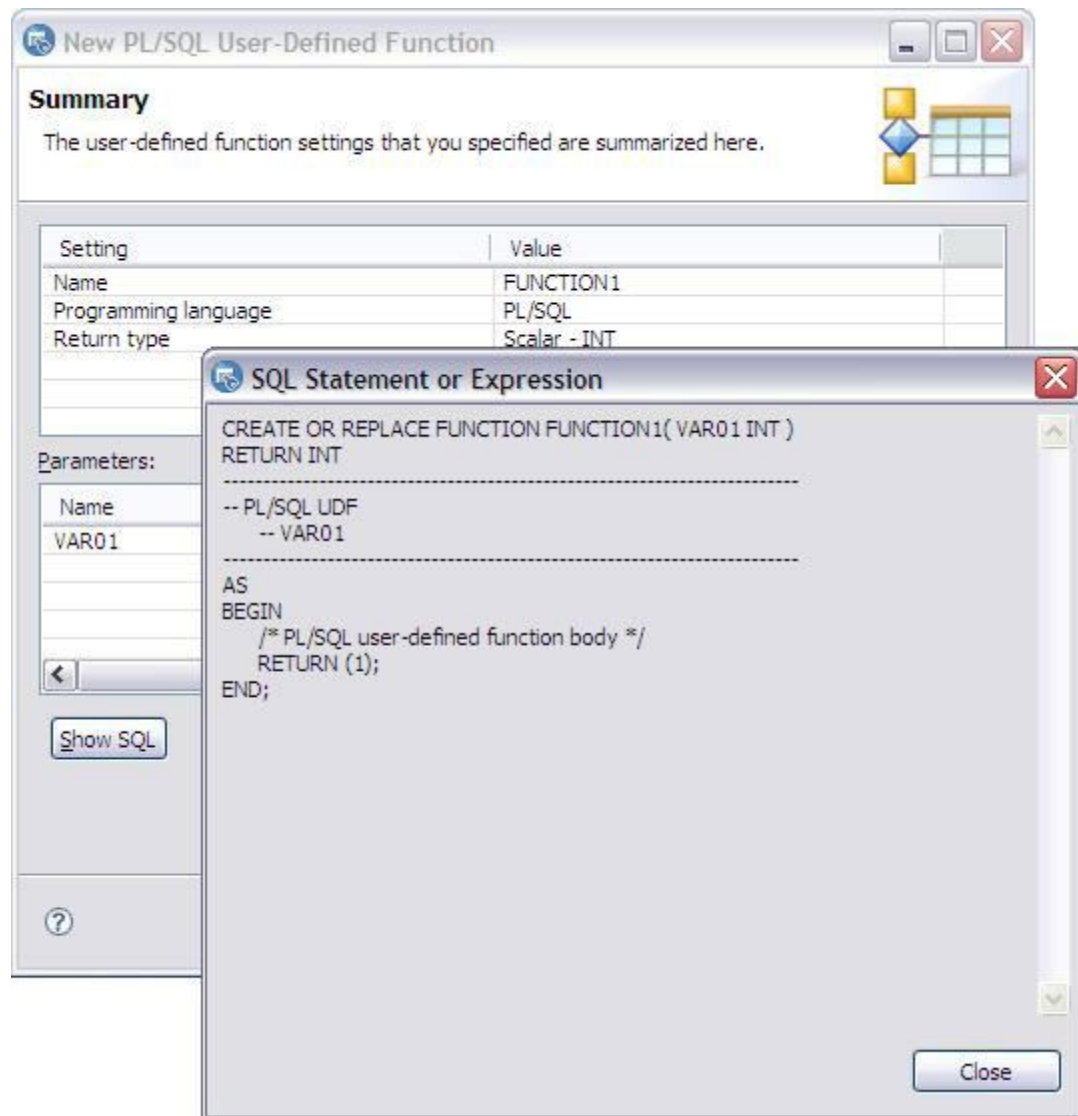
Hình 23. Các tham số cho một hàm PL/SQL với Oracle



Sử dụng **Add...** và hộp thoại Parameters để thêm các tham số của bất kì kiểu dữ liệu được hỗ trợ nào.

5. Khi bạn hoàn tất việc thêm các thông số, nhấn **Next >** để đến trang Summary.
6. Nhấn **Show SQL** để xem mã PL/SQL sẽ được tạo, như trong Hình 24.

Hình 24. Kiểm tra mã PL/SQL được tạo cho một hàm Oracle



ODS 2.2 không xác định kiểu dữ liệu của biểu thức và nó có thể không tự động xác định kiểu trả về. Mã được tạo ra sẽ không hoạt động nếu không có sửa đổi, nếu bạn thay đổi biểu thức SQL hoặc kiểu dữ liệu trả về sao cho chúng không tương ứng.

7. Nhấn **Finish** để tạo ra hàm này, thêm nó vào dự án của bạn và mở nó trong trình soạn thảo thường trình.

8. Kiểm tra hàm mới trong trình soạn thảo thường trình. Để thử nghiệm, bạn có thể trả về giá trị của tham số đầu vào VAR01, như sau:

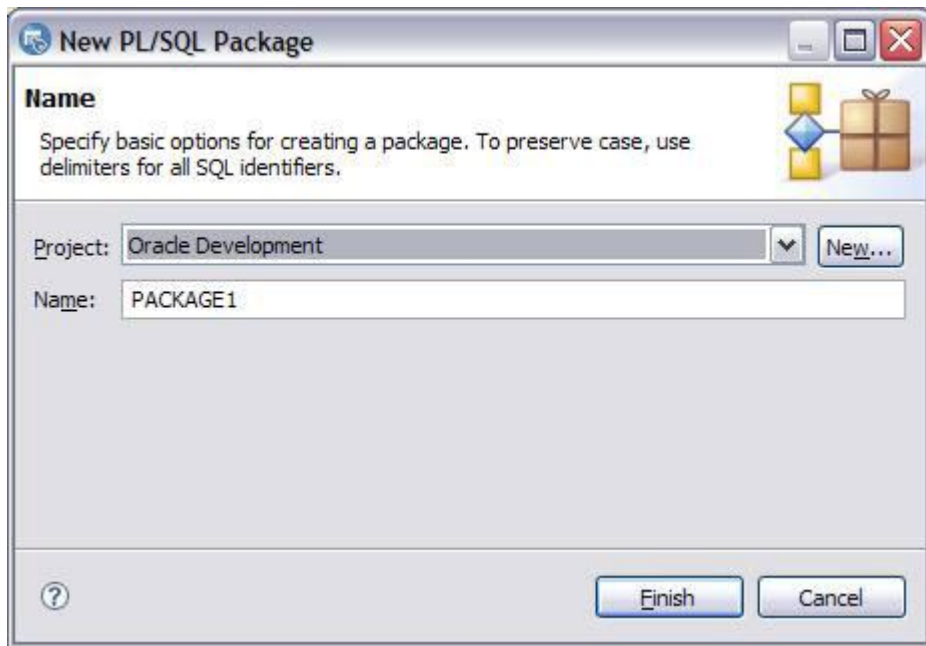
```
CREATE OR REPLACE FUNCTION FUNCTION1 ( VAR01 INT )  
  
RETURN INT  
  
AS  
  
BEGIN  
  
    RETURN VAR01;  
  
END;
```

Tạo một gói PL/SQL với Oracle

Thực hiện theo các bước sau để sử dụng trình thủ thuật các gói PL/SQL mới (New PL/SQL Package) để tạo một gói PL/SQL:

1. Nhấn chuột phải trên thư mục **PL/SQL Packages** dự án phát triển dữ liệu của Oracle. Từ trình đơn, chọn **New -> PL/SQL Package**. Trình thủ thuật New PL/SQL Package xuất hiện, như trong Hình 25.

Hình 25. Đặt tên một gói PL/SQL mới với Oracle



Trình thủ thuật này chỉ có một trang. Xác định các nền tảng đích và tên của gói PL/SQL.

2. Nhấn **Finish** để tạo gói PL/SQL, thêm nó vào dự án của bạn và mở nó trong trình soạn thảo gói PL/SQL.
3. Kiểm tra gói PL/SQL mới trong trình soạn thảo. Nó có hai thường trình, ExampleProcedure và ExampleFunction, mà bạn có thể triển khai và thử nghiệm như chúng có hoặc sửa đổi hoặc thay thế trong trình soạn thảo với các thường trình của gói PL/SQL riêng của bạn.

Đặc tả gói PL/SQL cung cấp chữ ký cho thủ tục và hàm:

```
CREATE OR REPLACE PACKAGE PACKAGE1
```

```
AS
```

```
/* PL/SQL package specification */  
  
PROCEDURE ExampleProcedure;  
  
FUNCTION ExampleFunction RETURN INT;  
  
END PACKAGE1;
```

Phần thân gói PL/SQL cài đặt thủ tục và hàm này:

```
CREATE OR REPLACE PACKAGE BODY PACKAGE1
```

```
AS
```

```
/* PL/SQL package body */
```

```
PROCEDURE ExampleProcedure
```

```
AS
```

```
X INT := 1;
```

```
BEGIN
```

```
  X := X + 1;
```

```
END ExampleProcedure;
```

```
FUNCTION ExampleFunction
```

```
RETURN INT
```

```
AS
```

```
X INT;
```

```
BEGIN
```

```
    X := 1;
```

```
    RETURN (X + 1);
```

```
END ExampleFunction;
```

```
END PACKAGE1;
```

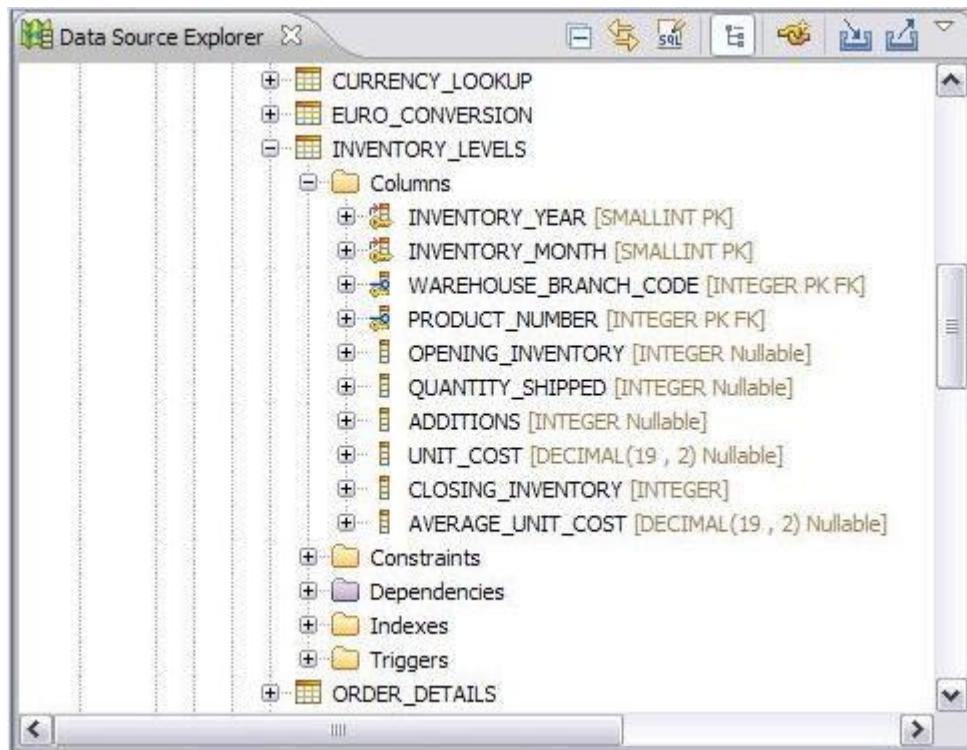
Di chuyển một bảng DB2 sang Oracle

Phần này cho bạn thấy cách soạn thảo gói PL/SQL mặc định mà bạn đã tạo ra để cho nó thực hiện một số công việc có ích. Sau đó, bạn sẽ tìm hiểu cách sao chép gói PL/SQL Oracle sang DB2. Vì vậy, bạn muốn có cùng một bảng trên cả hai DB2 và Oracle.

Bài viết này giả thiết rằng bạn đã cài đặt cơ sở dữ liệu mẫu GSDB trên máy chủ DB2 LUW 9.7 của bạn, nhưng nó không giả thiết rằng cơ sở dữ liệu Oracle của bạn có cùng bảng mẫu. Vì vậy, hãy làm theo các bước sau để tạo lại một bảng DB2 và dữ liệu của nó trên Oracle.

1. Kết nối đến cơ sở dữ liệu GSDB của bạn đối với DB2 và mở rộng các bảng trong lược đồ GOSALES của nó để tìm thấy bảng INVENTORY_LEVELS.
2. Mở rộng các cột của bảng để bạn có thể xem các tên và các kiểu dữ liệu, như trong Hình 26.

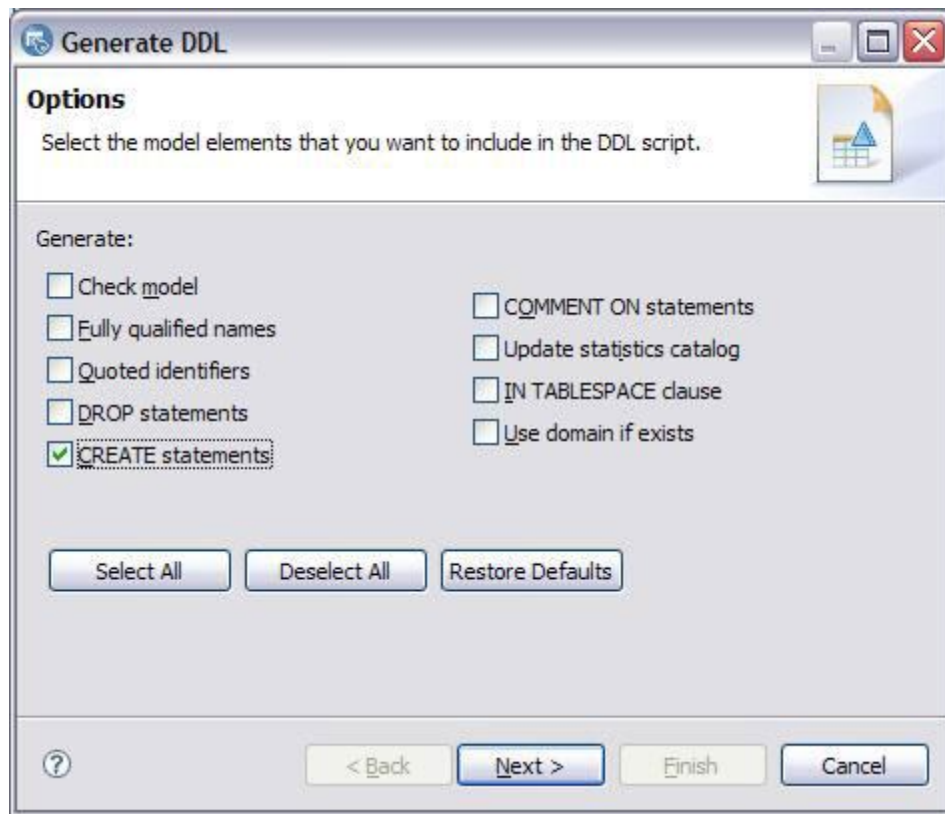
Hình 26. Xem các cột của bảng GOSALES.INVENTORY_LEVELS



3. Nhấn chuột phải vào bảng INVENTORY_LEVELS. Từ trình đơn hiện lên, chọn **Generate DDL....** Trình đơn Generate DDL xuất hiện
4. Xóa dấu chọn của hộp chọn Các câu lệnh COMMENT ON (**COMMENT ON statements**) và mệnh đề Trong vùng bảng (**IN TABLESPACE clause**). Để lại dấu chọn **CREATE statements**, như trong Hình 27.

Hình 27. Thiết lập các tùy chọn cho việc tạo DDL, để tạo bảng

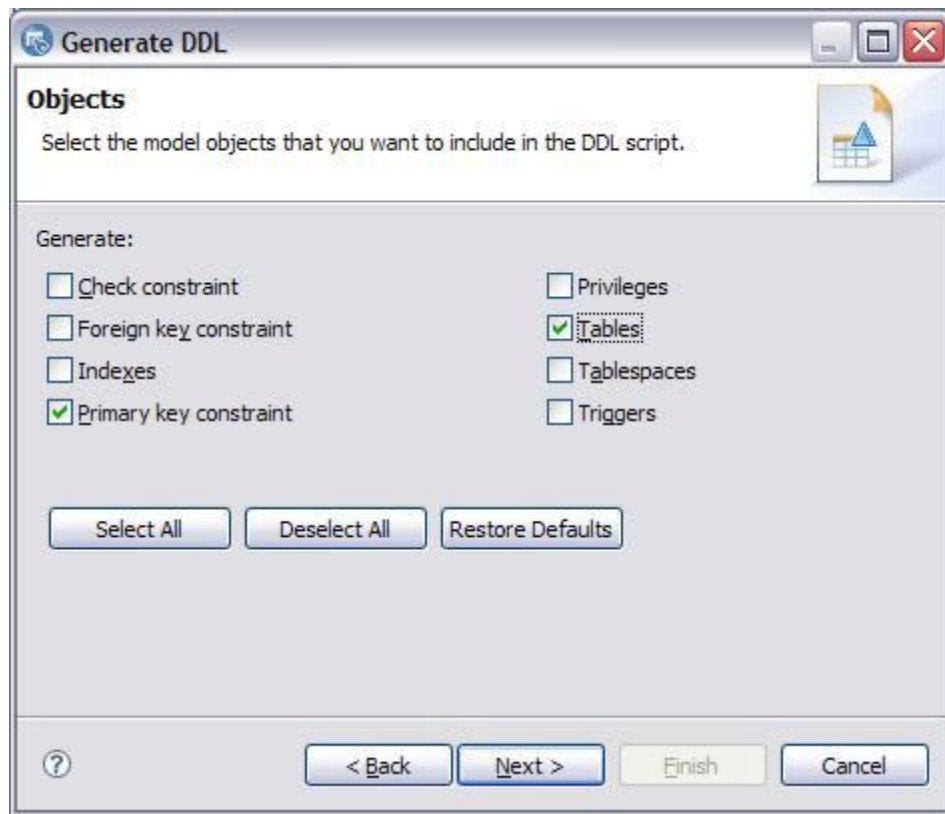
INVENTORY_LEVELS



5. Nhấn **Next >**. Trang Objects xuất hiện.
6. Nhấn Bỏ chọn Tất cả (**Deselect All**), sau đó đánh dấu chọn hộp Ràng buộc khóa chính (**Primary key constraint**) và Các bảng (**Tables**), như trong Hình 28.

Hình 28. Chọn các đối tượng cho việc tạo DDL, để tạo bảng

INVENTORY_LEVELS



7. Nhấn **Next >**. Trang Lưu và chạy DDL (Save and Run DDL) xuất hiện.
8. Xem lại các câu lệnh đã tạo ra trong vùng Xem trước DDL (Preview DDL):

```
--<ScriptOptions statementTerminator="!">
```

```
CREATE TABLE INVENTORY_LEVELS (  
  
    INVENTORY_YEAR SMALLINT NOT NULL,  
  
    INVENTORY_MONTH SMALLINT NOT NULL,
```


WAREHOUSE_BRANCH_CODE INTEGER NOT NULL,

PRODUCT_NUMBER INTEGER NOT NULL,

OPENING_INVENTORY INTEGER,

QUANTITY_SHIPPED INTEGER,

ADDITIONS INTEGER,

UNIT_COST DECIMAL(19 , 2),

CLOSING_INVENTORY INTEGER NOT NULL,

AVERAGE_UNIT_COST DECIMAL(19 , 2)

)

DATA CAPTURE NONE!

ALTER TABLE INVENTORY_LEVELS ADD CONSTRAINT
SQL090729130124500 PRIMARY KEY

(INVENTORY_YEAR,

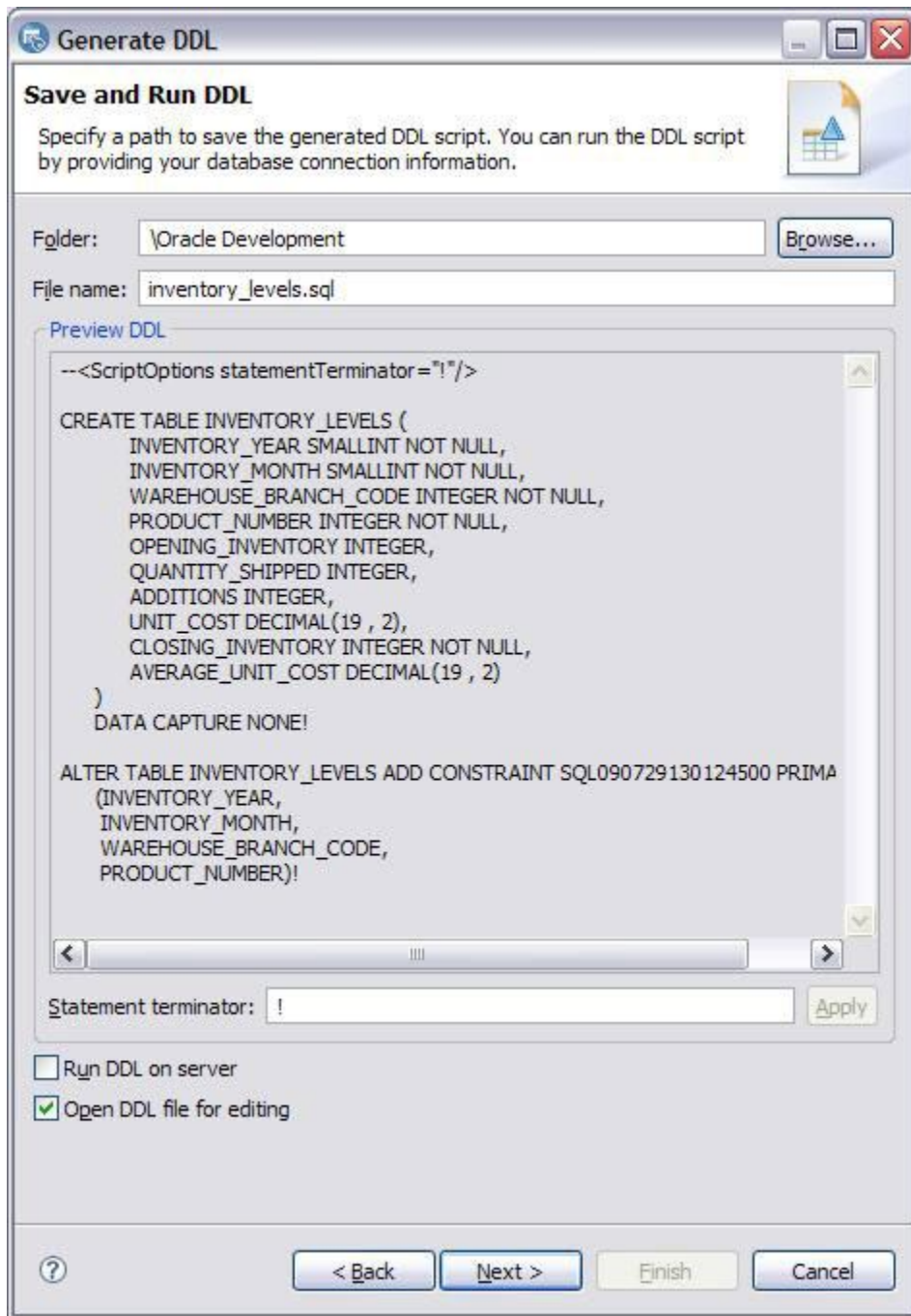
INVENTORY_MONTH,

WAREHOUSE_BRANCH_CODE,

PRODUCT_NUMBER)!

9. Nhấn **Browse...** để hiển thị hộp thoại Lựa chọn thư mục (Folder Selection).
10. Chọn Phát triển Oracle (**Oracle Development**) và nhấn **OK**.
11. Thay đổi giá trị của trường tên tệp thành `inventory_levels.sql`, như trong Hình 29.

Hình 29. Xác định tên cho tệp với DDL được tạo.



12. Nhấn **Next >** để hiển thị các trang Summary

13. Nhấn **Finish**. Kịch bản lệnh inventory_levels sẽ xuất hiện trong thư mục Các kịch bản lệnh SQL (SQL Scripts) của dự án phát triển Oracle của bạn.

14. Kịch bản lệnh này được tạo cho DB2, nhưng vì muốn chạy nó với Oracle, bạn cần phải vào trình soạn thảo SQL và loại bỏ mệnh đề DATA CAPTURE NONE. Rồi kịch bản lệnh này sẽ giống như sau:

```
--<ScriptOptions statementTerminator="!">
```

```
CREATE TABLE INVENTORY_LEVELS (  
  
    INVENTORY_YEAR SMALLINT NOT NULL,  
  
    INVENTORY_MONTH SMALLINT NOT NULL,  
  
    WAREHOUSE_BRANCH_CODE INTEGER NOT NULL,  
  
    PRODUCT_NUMBER INTEGER NOT NULL,  
  
    OPENING_INVENTORY INTEGER,  
  
    QUANTITY_SHIPPED INTEGER,  
  
    ADDITIONS INTEGER,  
  
    UNIT_COST DECIMAL(19 , 2),  
  
    CLOSING_INVENTORY INTEGER NOT NULL,  
  
    AVERAGE_UNIT_COST DECIMAL(19 , 2)  
  
    )!
```

```
ALTER TABLE INVENTORY_LEVELS ADD CONSTRAINT  
SQL090729130124500 PRIMARY KEY
```

```
(INVENTORY_YEAR,  
  
INVENTORY_MONTH,  
  
WAREHOUSE_BRANCH_CODE,  
  
PRODUCT_NUMBER)!
```

15. Nhấn chuột phải vào trong trình soạn thảo SQL. Từ trình đơn hiện lên của trình soạn thảo, chọn **Run SQL**.

Khi kích bản lệnh hoàn thành, bạn sẽ thấy một trạng thái Đã thành công ("Succeeded") cho phép toán này trong khung nhìn SQL Results. Vì tên bảng trong câu lệnh CREATE không đủ tiêu chuẩn, bảng này được tạo trong cơ sở dữ liệu Oracle của bạn theo mã nhận dạng người dùng của bạn.

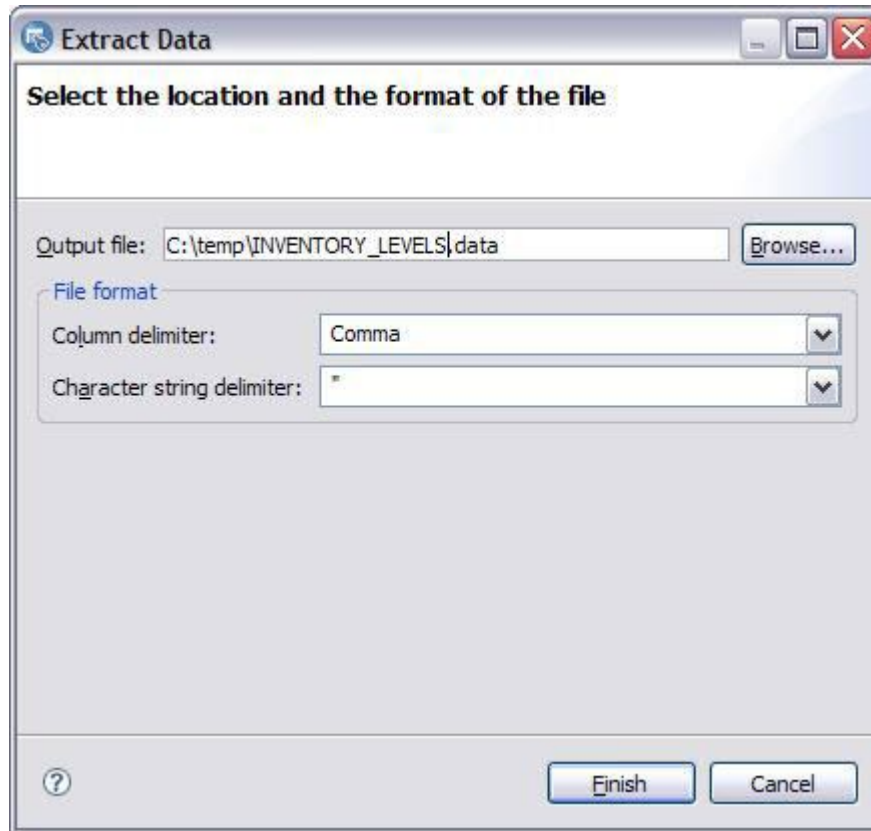
Trong các bước còn lại của phần này, bạn sẽ chuyển dữ liệu từ bảng INVENTORY_LEVELS trong DB2 vào bảng tương ứng trong Oracle.

16. Trong kết nối DB2 của bạn tới GSDB, nhấn chuột phải trên bảng INVENTORY_LEVELS. Từ trình đơn hiện lên, chọn **Data -> Extract...**
Trình thủ thuật Trích ra dữ liệu (Extract Data) xuất hiện.

17. Sử dụng trường tệp đầu ra để xác định đường dẫn và tên của tệp mà bạn muốn lấy dữ liệu từ bảng tới tệp đó. Theo tùy chọn, bạn có thể sử dụng

Browse... để hỗ trợ cho việc này. Ví dụ, bạn có thể nhập vào C:\temp\INVENTORY_LEVELS.data, như trong Hình 30.

Hình 30. Xuất dữ liệu từ GOSALES.INVENTORY_LEVELS



18. Nhấn **Finish**. Hoạt động này bắt đầu trong khung nhìn SQL Results. Khi nó hoàn thành, tab message1 cho thấy như sau:

```
Extracting "GOSALES"."INVENTORY_LEVELS"...
```

```
Data extraction was successful.
```

```
53730 row(s) extracted.
```

19. Trong kết nối Oracle của bạn, nhấn chuột phải trên bảng INVENTORY_LEVELS. Từ trình đơn hiện lên, chọn **Data** -> **Load...**
Trình thủ thuật Nạp dữ liệu (Load Data) sẽ xuất hiện.
20. Trong trường tệp đầu vào, nhập vào đường dẫn và tên của tệp đầu ra, mà đã trích xuất dữ liệu đến trình thủ thuật Trích xuất dữ liệu (Extract Data) trong bước 17.
21. Nhấn **Finish**. Một lần nữa, một hoạt động bắt đầu trong khung nhìn SQL Results. Khi nó hoàn thành, bảng INVENTORY_LEVELS trong Oracle chứa dữ liệu tương tự như bảng trong DB2.
22. Để xem một ví dụ về dữ liệu đó, nhấn chuột phải vào bảng INVENTORY_LEVELS trong kết nối Oracle của bạn và chọn **Data** -> **Sample Contents** từ trình đơn hiện lên.

Hoạt động Các nội dung mẫu (Sample Contents) hiển thị dữ liệu trong tab Result1 của khung nhìn SQL Results.

Sửa đổi một gói PL/SQL với Oracle

Khi bạn kết thúc việc tạo một đối tượng mới trong một trình thủ thuật, ODS 2.2 mở đối tượng đó trong trình soạn thảo thích hợp.

Nếu bạn đã đóng một thể hiện của trình soạn thảo cho một đối tượng, bạn có thể mở lại nó bằng cách nhấn đúp vào đối tượng đó trong dự án phát triển dữ liệu của bạn hoặc bằng cách nhấn chuột phải vào nó và chọn **Open** từ trình đơn hiện lên.

1. Mở trình soạn thảo gói PL/SQL cho gói PL/SQL có tên là package1 mà bạn đã tạo ra trong dự án Phát triển Oracle của mình.

Trình soạn thảo gói PL/SQL có hai tab, một cho đặc tả và một cho phần thân. Đặc tả này là khai báo của API công cộng cho gói PL/SQL của bạn. Việc thực hiện là trong phần thân.

Cả hai tab này hiển thị mã nguồn của bạn bằng trình soạn thảo SQL; tuy nhiên, tùy chọn Xác nhận hợp lệ cú pháp câu lệnh (**Validate Statement Syntax**) (hiển thị trong trình đơn hiện lên với trình soạn thảo) không được đánh dấu chọn theo mặc định do ODS 2.2 không có khả năng phân tích cú pháp PL/SQL trực tiếp để xác nhận tính hợp lệ của cú pháp. Nếu bạn đánh dấu chọn tùy chọn này trong trình đơn hiện lên, thì trình phân tích cú pháp SQL sẽ sinh ra các lỗi về cú pháp PL/SQL.

Nội dung của cả hai tab hoàn toàn có thể soạn thảo được. Bạn có thể thay đổi tên của gói PL/SQL, thêm, sửa đổi hoặc loại bỏ các thường trình và (trong tab Body) cài đặt các phần thân thường trình.

Bạn có thể thay đổi tên của gói PL/SQL trong trình soạn thảo, nhưng bạn phải thay đổi nó tại bốn vị trí, ở đầu và ở cuối của đặc tả và phần thân. Khi bạn lưu một sự thay đổi tên, tên đó được cập nhật trong dự án của bạn.

- Đi đến tab Specification và thay đổi tên của gói thành INVENTORY và thay các mẫu hiện có bằng chữ ký cho một hàm QUERY_CLOSING_INVENTORY, như sau:

```
CREATE OR REPLACE PACKAGE INVENTORY
```

```
AS
```

```
    FUNCTION QUERY_CLOSING_INVENTORY(
```


Y NUMBER,

M NUMBER,

W NUMBER,

P NUMBER

)

RETURN NUMBER;

END INVENTORY;

3. Đi đến tab Body và thay đổi tên của gói thành INVENTORY và thay các mẫu hiện có bằng việc thực hiện với hàm QUERY_CLOSING_INVENTORY, như sau:

CREATE OR REPLACE PACKAGE BODY INVENTORY

AS

FUNCTION QUERY_CLOSING_INVENTORY(

Y NUMBER,

M NUMBER,

W NUMBER,

P NUMBER

)

RETURN NUMBER

AS

X NUMBER(38,0);

BEGIN

SELECT CLOSING_INVENTORY INTO X

FROM INVENTORY_LEVELS

WHERE INVENTORY_YEAR = Y

AND INVENTORY_MONTH = M

AND WAREHOUSE_BRANCH_CODE = W

AND PRODUCT_NUMBER = P;

RETURN X;

END;

END INVENTORY;

4. Lưu các thay đổi của bạn bằng cách nhấn **Ctrl+S** hoặc bằng cách chọn **File** -> **Save** trên trình đơn chính hoặc từ trình đơn hiện lên của trình soạn thảo. Cả hai đặc tả và phần thân được lưu trữ và tên mới xuất hiện trong thư mục PL/SQL Packages của dự án của bạn.

Sau đó, bạn sẽ thấy cách triển khai và thử nghiệm gói PL/SQL này.

Soạn thảo một thường trình PL/SQL

Sau khi tạo một thủ tục hoặc hàm PL/SQL, bạn thường muốn soạn thảo nó để thêm logic nghiệp vụ riêng của bạn. Các trình soạn thảo cho một thủ tục hoặc hàm PL/SQL cung cấp chức năng tương tự cho trình soạn thảo cho một gói PL/SQL.

Trình soạn thảo thường trình PL/SQL có hai tab. Tab Cấu hình (Configuration) chứa các thuộc tính của mô hình. Tab Source chứa mã nguồn PL/SQL.

Tab Configuration đối với một thủ tục PL/SQL cho Oracle được hiển thị trong Hình 31:

Hình 31. Xem lại tab Configuration của trình soạn thảo thủ tục PL/SQL cho

TEST_TIMESTAMP



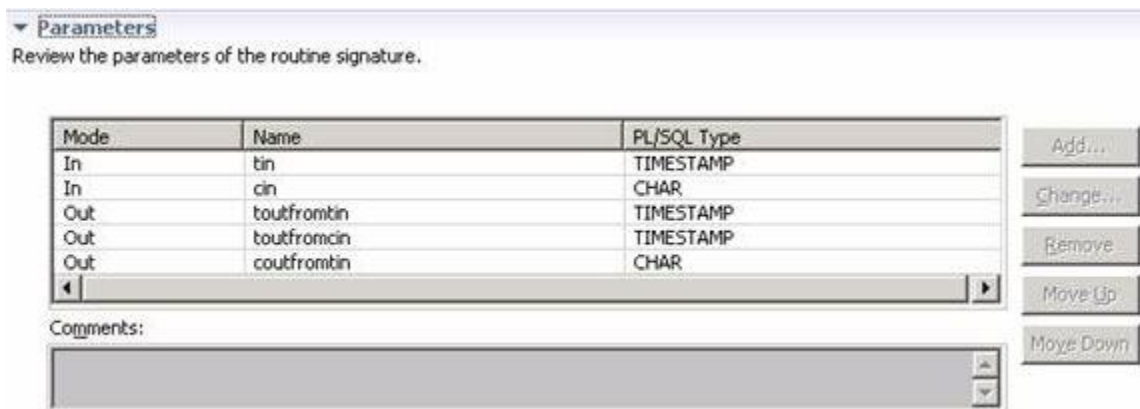
Thuộc tính có ích nhất trên tab Configuration là cái cho biết liệu thường trình này có được kích hoạt để gỡ rối không. Bạn phải đánh dấu chọn hộp này và triển khai thay đổi này trước khi bạn có thể gỡ rối thủ tục.

ODS 2.2 không hỗ trợ gỡ rối một hàm độc lập (PL/SQL, SQL hoặc OLE DB).

Các liên kết Triển khai (**Deploy**), Chạy (**Run**) và Gỡ rối (**Debug**) ở bên phải của phần Cấu hình triển khai (Deployment Configuration) hoạt động như các nút bấm hay các mục của trình đơn, để gọi các hành động bình thường đối với triển khai và thử nghiệm công việc của bạn.

Các phần Các tham số và Các tùy chọn (Parameters and Options) có thể được mở rộng. Các tham số trong bảng cho một thường trình PL/SQL là không soạn thảo được, như trong Hình 32.

Hình 32. Xem lại các tham số cho thủ tục PL/SQL TEST_TIMESTAMP



Bạn có thể soạn thảo nội dung của tab Source. Bạn có thể thay đổi tên của thường trình, thêm, sửa đổi hoặc xoá các tên tham số, các chế độ và các kiểu dữ liệu, sửa đổi kiểu trả về của một hàm và cài đặt phần thân của thường trình đó. Trình soạn thảo không xác nhận hợp lệ cú pháp PL/SQL khi bạn nhập nó vào, nhưng nó nhấn mạnh cú pháp bằng màu như trong Hình 33.

Hình 33. Soạn thảo mã nguồn của thủ tục PL/SQL TEST_TIMESTAMP

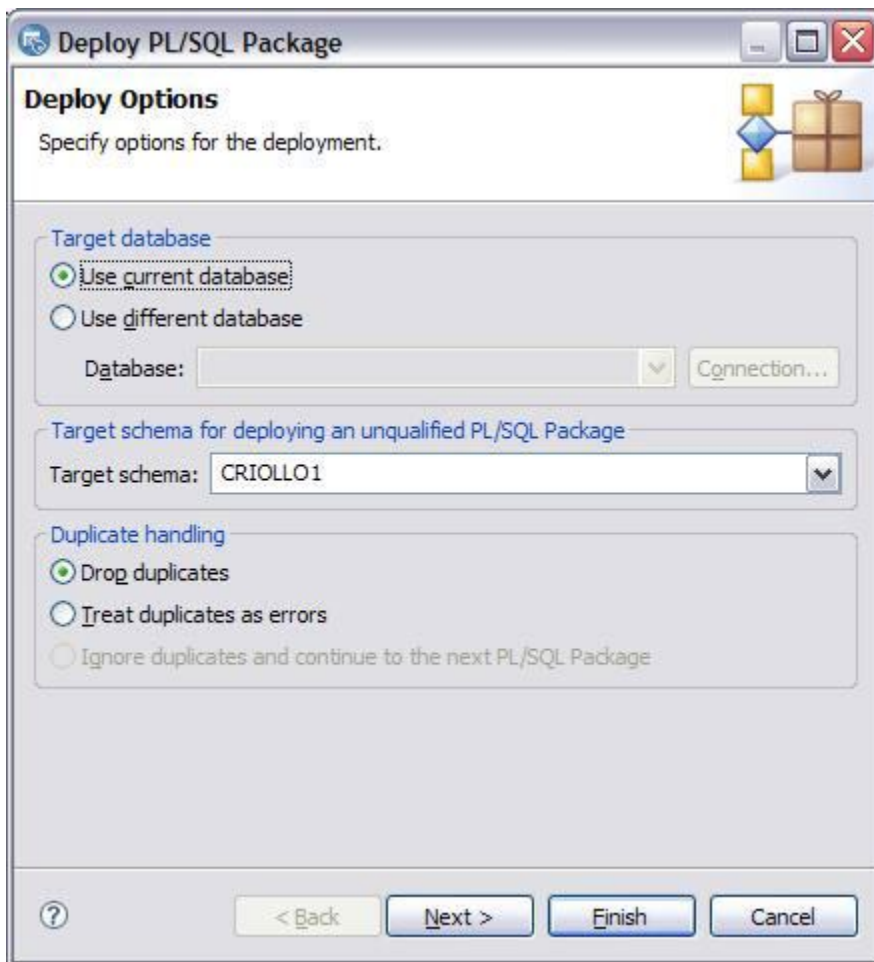


Triển khai gói PL/SQL cho máy chủ Oracle của bạn

Không giống như một ứng dụng Java, mà có thể biên dịch và chạy cục bộ, bạn phải triển khai các gói, thủ tục và hàm PL/SQL cho máy chủ cơ sở dữ liệu của bạn trước khi bạn có thể chạy chúng. Bạn không thể chạy chúng trong dự án của bạn, nhưng có thể triển khai chúng và chạy chúng từ dự án của bạn.

Khi bạn chọn một gói, thủ tục hoặc hàm PL/SQL trong một dự án phát triển dữ liệu và sau đó chọn hành động **Deploy...** từ trình đơn hiện lên của đối tượng, và nếu bạn kết nối với cơ sở dữ liệu liên kết với dự án, thì trình thủ thuật Triển khai gói PL/SQL (Deploy PL/SQL Package) xuất hiện. Trong trường hợp này, trình thủ thuật bắt đầu với trang Triển khai tùy chọn (Deploy Options), như trong Hình 34.

Hình 34. Thiết lập các tùy chọn triển khai cho một gói PL/SQL



Khi bạn chọn thư mục của đối tượng này hoặc nhiều đối tượng trong một thư mục và sau đó chọn hành động **Deploy...** từ trình đơn hiện lên, trình thủ thuật Triển khai gói PL/SQL (Deploy PL/SQL Package) xuất hiện. Trong trường hợp này, trình thủ thuật bắt đầu với trang Chọn lựa (Selection) trước trang Deploy Options như trong Hình 35. Từ trang này bạn có thể kiểm tra các đối tượng riêng hoặc nhấn vào Chọn tất cả (**Select All**).

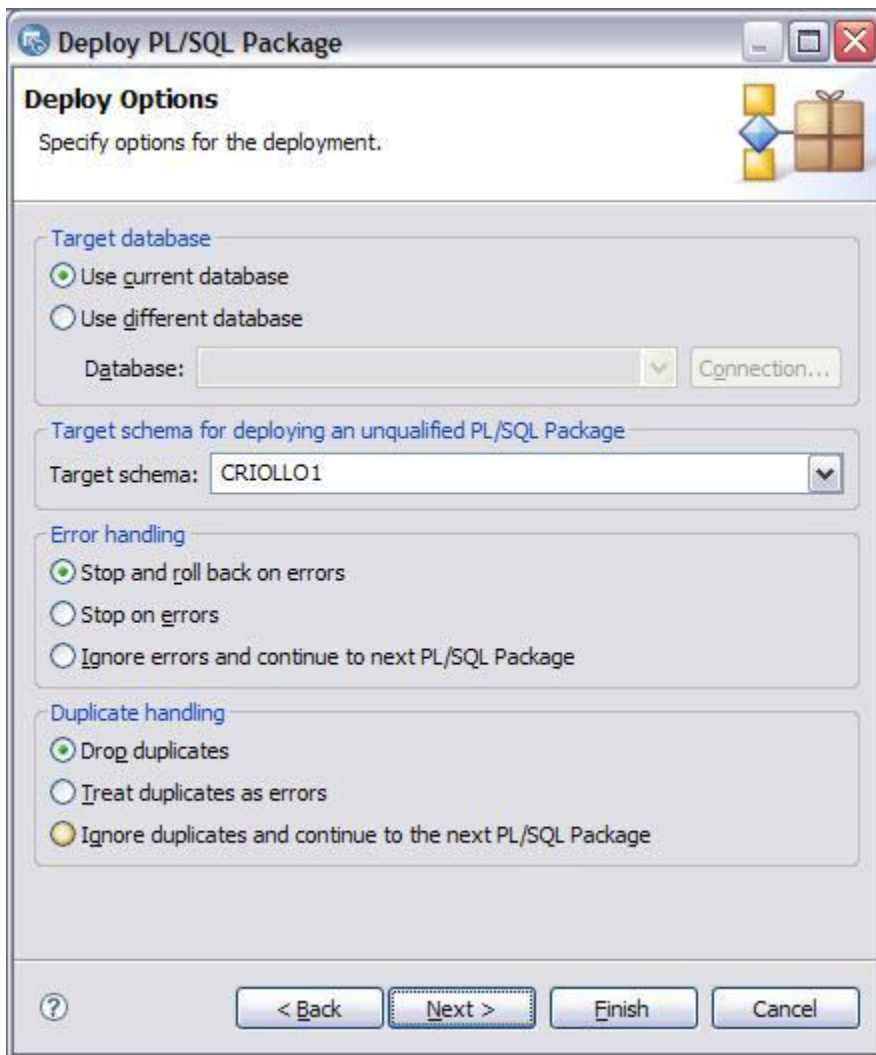
Hình 35. Chọn các gói PL/SQL để triển khai



Trình thủ thuật Triển khai (Deploy) cho phép bạn triển khai một hoặc nhiều đối tượng cùng loại.

Khi bạn triển khai nhiều thủ tục PL/SQL, bạn có các tùy chọn phụ trên trang Deploy Options, như trong Hình 36.

Hình 36. Thiết lập các tùy chọn cho việc triển khai nhiều gói PL/SQL



Bạn có thể triển khai các đối tượng được chọn cho cơ sở dữ liệu hiện tại của mình, đây là cơ sở dữ liệu có liên kết với dự án hoặc có thể chọn một cơ sở dữ liệu tương thích khác làm đích triển khai của bạn.

Bạn có thể xác định một lược đồ đích (hoặc chủ sở hữu, với Oracle). Đối với DB2, lược đồ đích là SQLID sẽ được sử dụng cho các đối tượng không đủ tiêu chuẩn.

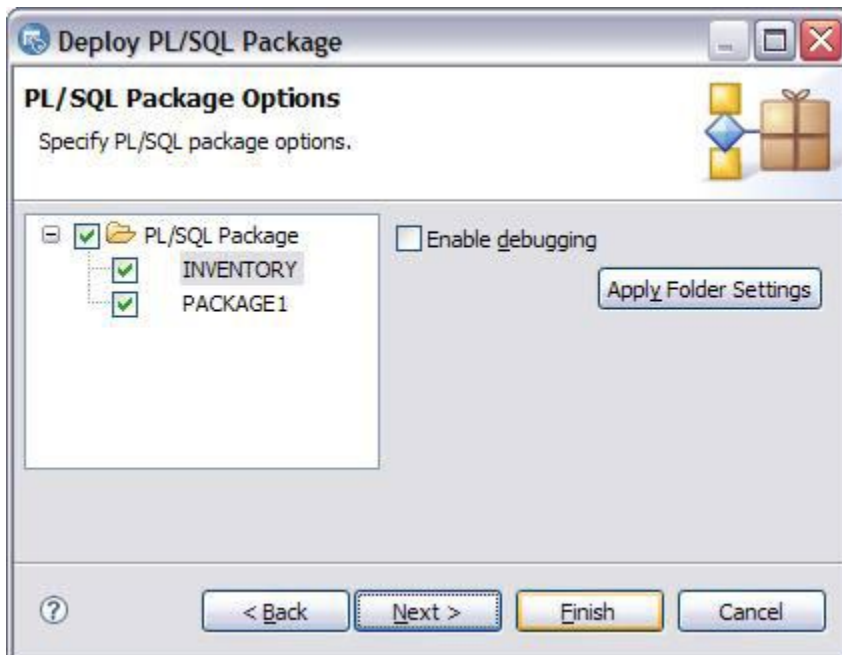
Đối với DB2, bạn có thể thay đổi đường dẫn mặc định bằng cách thêm hoặc tách các lược đồ ra khỏi danh sách được phân cách bằng dấu phẩy này.

Bạn cũng có thể xác định cách xử lý các đối tượng trùng lặp. Theo mặc định, ODS 2.2 loại bỏ các đối tượng trùng lặp trước khi tạo lại chúng, nhưng nó cũng có thể dừng và khôi phục lại các thay đổi nếu nó gặp một lỗi do đối tượng đã tồn tại trên cơ sở dữ liệu.

Khi mã nguồn của gói PL/SQL của bạn chứa câu lệnh `CREATE OR REPLACE`, các tùy chọn xử lý trùng lặp sẽ được bỏ qua trừ khi bạn loại bỏ hoặc thay thế `OR REPLACE`. Đối với DB2 LUW 9.7, bạn có thể thêm `OR REPLACE` cho bất kỳ câu lệnh `CREATE PROCEDURE` hoặc `CREATE FUNCTION` nào, nhưng với các thường trình SQL và Java, các tùy chọn xử lý trùng lặp không được bỏ qua.

Khi bạn nhấn **Next >**, trang Deploy Options hoặc trang PL/SQL Package Options xuất hiện, như trong Hình 37.

Hình 37. Thiết lập các tùy chọn triển khai cho các gói PL/SQL



Trang này cho phép bạn thay đổi các tùy chọn cho mỗi đối tượng mà bạn đang triển khai. Đối với gói PL/SQL, lựa chọn duy nhất của bạn là có cho phép gỡ rối

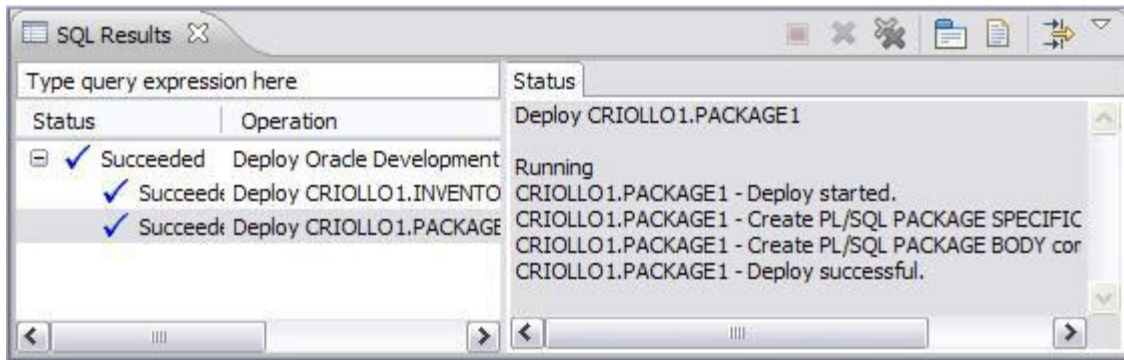
không. Có thể có nhiều tùy chọn hơn cho các kiểu đối tượng khác và các đích cơ sở dữ liệu.

Ngoại trừ lựa chọn gỡ rối, các tùy chọn mà bạn thay đổi ở đây không được lưu trong dự án của bạn. Nếu việc triển khai thành công, chúng sẽ được lưu chỉ trong cơ sở dữ liệu đích của bạn.

Khi bạn nhấn vào **Finish**, các đối tượng được chọn sẽ được triển khai tuần tự trong một đoạn nền. Các kết quả được báo cáo trong khung nhìn SQL Results.

Khi có nhiều hơn một đối tượng được triển khai, các hoạt động cho từng đối tượng được nhóm lại với nhau trong khung nhìn SQL Results. Để xem trạng thái của một đối tượng riêng, mở rộng các nhóm và chọn đối tượng đó, như trong Hình 38.

Hình 38. Mở rộng hoạt động triển khai trong khung nhìn SQL Results



Chạy một gói PL/SQL trên Oracle

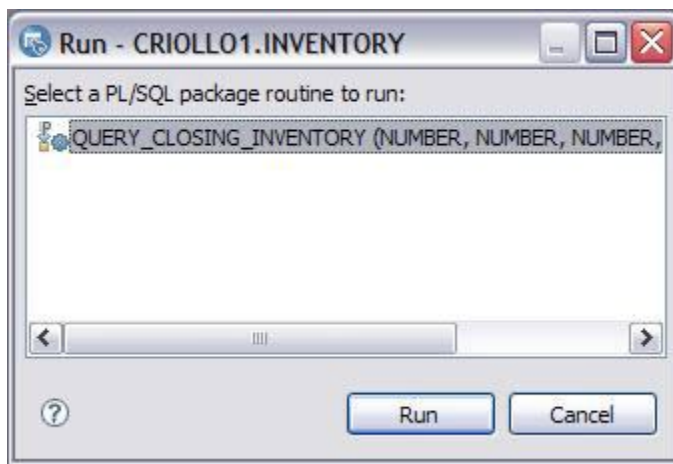
Bạn có thể chạy một thường trình độc lập (bất kể ngôn ngữ thực hiện của nó) trong Data Source Explorer. Trong Data Project Explorer, nếu có thể triển khai nó, thì bạn cũng có thể chạy nó. Khi bạn chọn một gói PL/SQL để chạy, đầu tiên bạn được nhắc chọn một thủ tục hoặc hàm trong gói PL/SQL. Nếu thường trình khai báo các tham số đầu vào, thì bạn cũng được nhắc để cung cấp giá trị cho chúng.

Thực hiện theo các bước sau để chạy hàm

CRIOLLO1.INVENTORY.QUERY_CLOSING_INVENTORY:

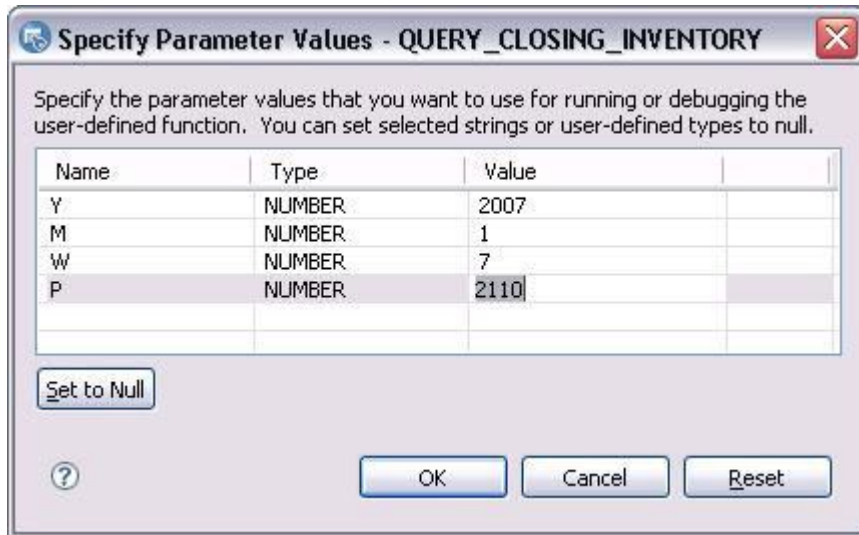
1. Nhấn chuột phải trên gói PL/SQL có tên INVENTORY. Từ trình đơn hiện lên, chọn **Run...** Hộp thoại Run - CRIOLLO1.INVENTORY xuất hiện, như trong Hình 39.

Hình 39. Chạy hàm QUERY_CLOSING_INVENTORY trong gói PL/SQL



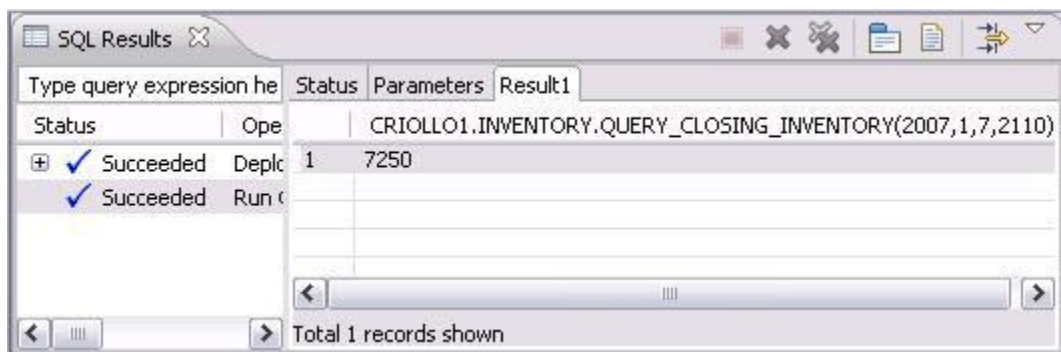
2. Nhấn **Run**. Hộp thoại Chỉ định các giá trị tham số (Specify Parameter Values) xuất hiện.
3. Như trong hình 40, nhập các giá trị tham số sau đây:
 - o Y = 2007
 - o M = 1
 - o W = 7
 - o P = 2110

Hình 40. Nhập các giá trị cho các tham số đầu vào để chạy hàm QUERY_CLOSING_INVENTORY



4. Nhấn **OK**. Hoạt động chạy xuất hiện trong khung nhìn SQL Results. Tab Result1 với hoạt động này cho thấy giá trị trả về của hàm, như trong Hình 41.

Hình 41. Kiểm tra kết quả sau khi chạy hàm QUERY_CLOSING_INVENTORY



Bạn cũng có thể sử dụng Data Source Explorer hoặc Data Project Explorer để chạy một thường trình trong một gói PL/SQL.

Gỡ rối một gói PL/SQL trên Oracle

ODS 2.2 cung cấp các trình gỡ rối thường trình PL/SQL cho cả hai DB2 LUW 9.7 và (khi sử dụng trình điều khiển JDBC mở Oracle) Oracle 10g hoặc 11g.

Thực hiện theo các bước sau để gỡ rối một gói PL/SQL trên Oracle:

1. Khi bạn tạo, soạn thảo hoặc triển khai một thường trình có thể được gỡ rối, hãy đánh dấu chọn hộp Cho phép gỡ rối (**Enable debugging**):

Nếu bạn triển khai một gói hoặc thủ tục PL/SQL mà không đánh dấu chọn hộp này, máy chủ không tạo ra các thông tin gỡ rối cho nó. Thiết lập và lưu tùy chọn này trong trình soạn thảo thường trình là không đủ cho phép gỡ rối. Bạn cũng phải triển khai thường trình đó với tùy chọn này đã chọn.

Triển khai gói PL/SQL có tên INVENTORY. Hãy chắc chắn đánh dấu chọn hộp kiểm tra **Enable debugging** trên trang PL/SQL Package Options.

2. Nhấn chuột phải vào gói INVENTORY. Từ trình đơn hiện lên, chọn **Debug...** Tương tự như khi bạn chạy gói đó, điều này làm cho hộp thoại Run - CRIOLLO1.INVENTORY xuất hiện với hàm được chọn sau đây:
QUERY_CLOSING_INVENTORY
(NUMBER,NUMBER,NUMBER,NUMBER) RETURN NUMBER

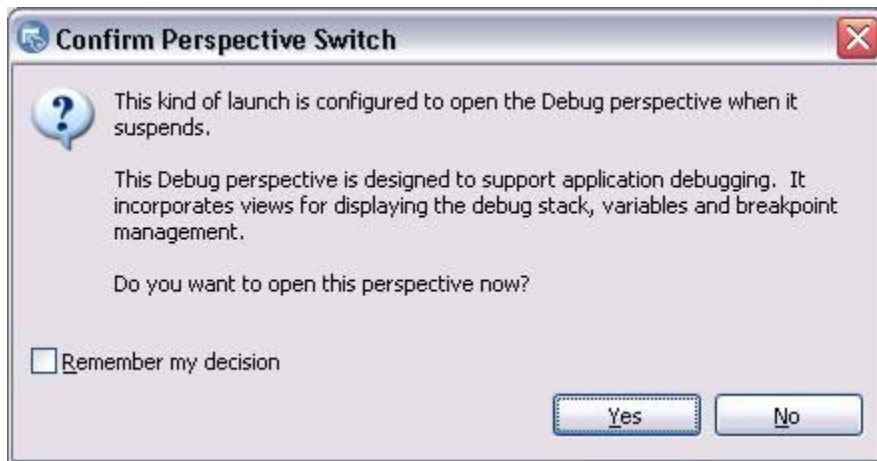
3. Như các bạn đã làm khi chạy thường trình này, hãy nhập các giá trị tham số sau đây:
 - o Y = 2007

- $M = 1$
- $W = 7$
- $P = 2110$

4. Nhấn **OK**.

Sau đó bạn sẽ được yêu cầu xác nhận rằng bạn muốn chuyển sang phối cảnh Debug, như trong Hình 42.






Hình 42. Xác nhận rằng bạn muốn chuyển sang phối cảnh Debug



5. Nhấn **Yes**.

6. Phối cảnh Debug mang đến cho bạn các khung nhìn mà bạn cần để gỡ rối một thường trình. Bạn có thể sắp đặt và bố trí theo kích thước các khung nhìn này cho phù hợp với sở thích riêng của bạn:

- Theo dõi ngăn xếp xuất hiện trong khung nhìn Debug. Thanh công cụ cho khung nhìn này chứa các nút có ích sau đây:

Biểu tượng	Tên nút (phím tắt)	Chức năng
	Resume (F8)	Chạy thường trình của bạn tới điểm dừng kế tiếp hoặc hoàn thành.
	Terminate (Ctrl+F2)	Kết thúc thường trình của bạn và phiên làm việc gỡ rối mà không cần trả về kết quả.
	Step into (F5)	Cho phép bạn gỡ rối hàm đã tham chiếu hoặc thủ tục đã gọi câu lệnh hiện tại.
	Step over (F6)	Bước qua hàm đã tham chiếu hoặc thủ tục đã gọi câu lệnh hiện tại.
	Step return (F7)	Trả về từ một hàm hay thủ tục mà bạn đã bước vào.

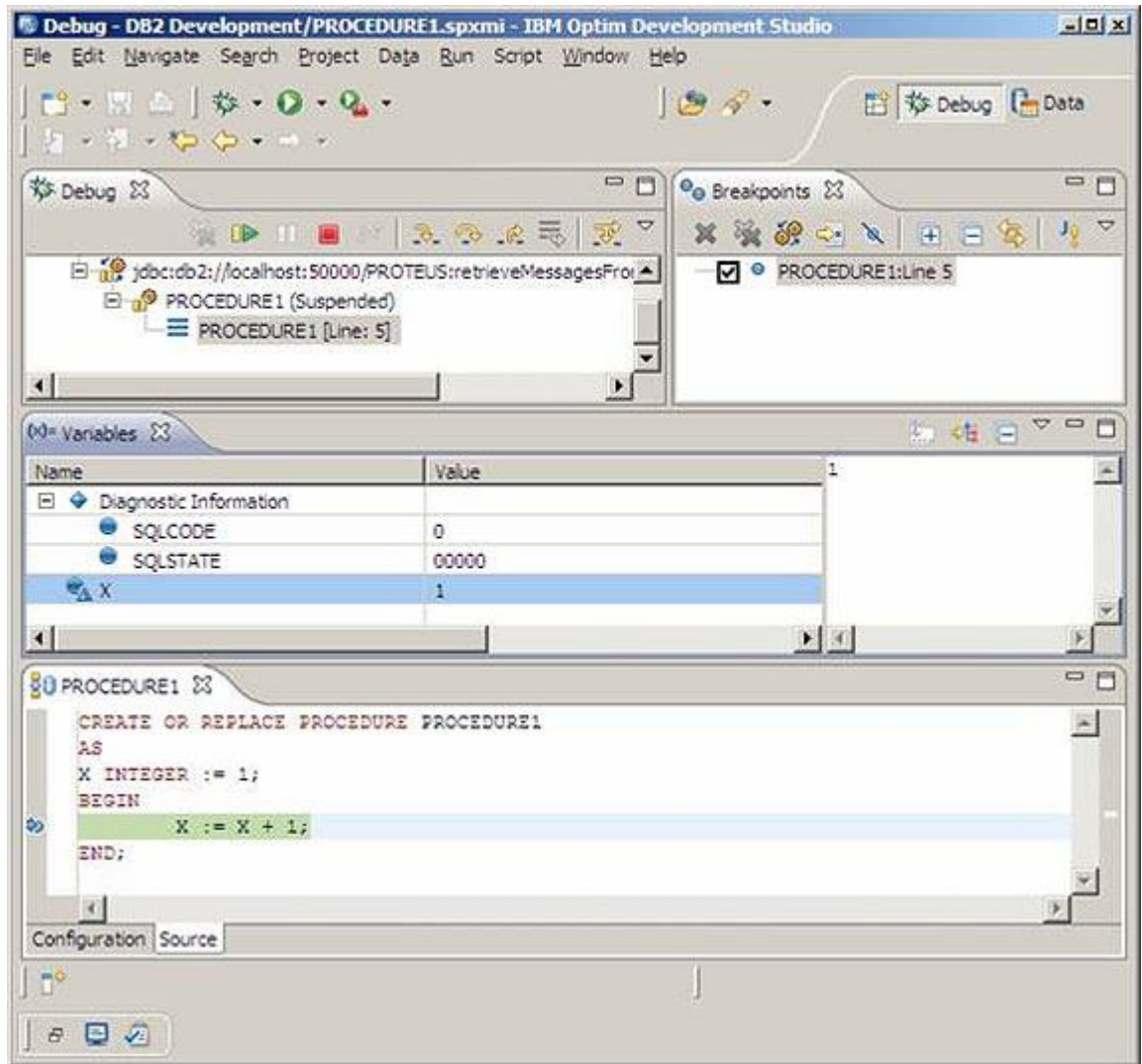
-
- Các điểm dừng của bạn xuất hiện trong khung nhìn Breakpoints.
- Các biến thường trình với các giá trị hiện tại của chúng xuất hiện trong khung nhìn Các biến (Variables).
- Mã nguồn của bạn sẽ xuất hiện trong trình soạn thảo thường trình.

7. Trình gỡ rối luôn dừng lại ở dòng đầu tiên của một thường trình, ngay cả khi bạn chưa đặt một điểm dừng (Breakpoint).

Để thiết lập một điểm dừng, nhấn đúp vào lề trái của trình soạn thảo thường trình hoặc nhấn chuột phải vào trong lề bên trái của câu lệnh, nơi bạn muốn trình gỡ rối dừng lại và chọn **Add Breakpoint** từ trình đơn hiện lên.

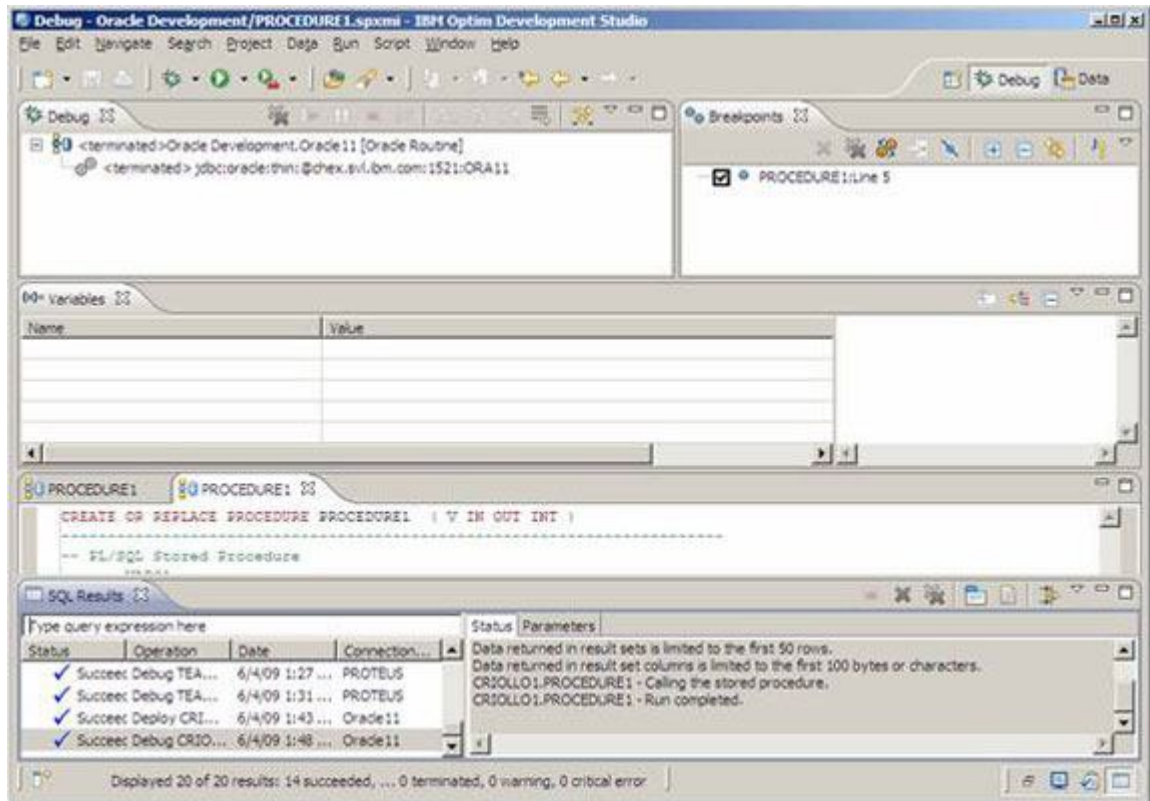
Hình 43 cho thấy khung nhìn của phối cảnh gỡ rối sau khi đã bắt gặp một điểm dừng trên dòng 5 của PROCEDURE1.

Hình 43. Thiết lập một điểm dừng trong phối cảnh Debug



8. Nếu bạn chạy cho đến khi hoàn thành, thì khung nhìn SQL Results sẽ xuất hiện để hiển thị cho bạn trạng thái và các kết quả bất kỳ, như trong Hình 44.

Hình 44. Kiểm tra trạng thái và kết quả sau khi gỡ rối



Lời khuyên: Khi bạn soạn thảo một gói hoặc thủ tục Oracle PL/SQL mà bạn định gỡ rối, hãy giữ CREATE và từ khóa đối tượng trên cùng dòng. Máy chủ của Oracle sẽ loại bỏ các ngắt dòng cho đến sau từ khóa đối tượng (PACKAGE, PACKAGE BODY, PROCEDURE hoặc FUNCTION). Vì vậy, nếu bạn không làm điều này, các số dòng trong mã nguồn của bạn sẽ không khớp với các số dòng mà máy chủ biết.

Một thủ tục có thể gọi thủ tục khác hoặc thực hiện một hàm khi đánh giá một biểu thức. Trình gỡ rối cho phép bạn bước vào các cuộc gọi lồng nhau này và các tham chiếu hàm.

Bạn có thể gỡ rối một hàm PL/SQL nếu nó là một gói PL/SQL và bạn có thể bước từ một thủ tục PL/SQL đến một hàm PL/SQL nếu nó đang ở trong cùng một gói PL/SQL. Bạn có thể gỡ rối một thủ tục PL/SQL nếu nó ở trong một gói PL/SQL

hoặc nếu nó độc lập và bạn có thể bước từ một cuộc gọi trong phần thân của một thủ tục vào trong thủ tục PL/SQL mà nó được gọi.

Không giống như trình gỡ rối thường trình SQL và Java trong ODS 2.2, trình gỡ rối PL/SQL cho DB2 không yêu cầu bạn cài đặt hoặc bắt đầu một trình quản lý phiên gỡ rối. Trình quản lý phiên gỡ rối được tích hợp với máy khách của trình gỡ rối và bắt đầu tự động.

Phát triển các đối tượng PL/SQL cho DB2 LUW 9.7

Để phát triển một gói, thủ tục hoặc hàm PL/SQL cho DB2, bạn thực hiện cùng các nhiệm vụ như làm với Oracle. Các sự khác nhau giữa sự hỗ trợ của DB2 và của Oracle cho PL/SQL không quan trọng:

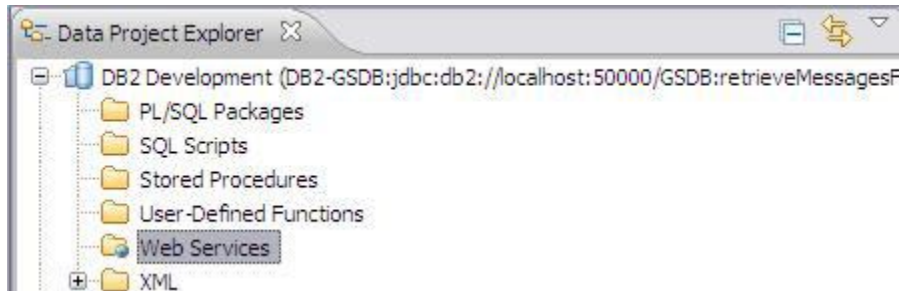
- Khi bạn đặt tên một gói, thủ tục hoặc hàm PL/SQL cho Oracle, tên (không có chủ sở hữu) được dài đến 30 ký tự. Đối với DB2, tên có thể dài đến 128 ký tự.
- Khi bạn tạo một thủ tục hoặc hàm độc lập cho DB2, bạn có một sự lựa chọn về các ngôn ngữ khác PL/SQL (SQL và Java). Do đó, trường Language để chọn ngôn ngữ thực hiện trong trình thủ thuật này là được phép, và là SQL theo mặc định.
- Khi bạn đang sử dụng một trình thủ thuật để thêm một tham số cho một thủ tục hoặc hàm PL/SQL hoặc thay đổi kiểu trả về của một hàm, danh sách các kiểu dữ liệu cho DB2 bao gồm các kiểu dữ liệu SQL cũng như các kiểu dữ liệu PL/SQL. Bạn có thể sử dụng, ví dụ, CHAR(254) FOR BIT DATA đối với tham số cho DB2. Ở đây DB2 hỗ trợ các từ đồng nghĩa cho các kiểu

dữ liệu, như là INTEGER và INT, trình thủ thuật hiển thị từ đồng nghĩa DB2 ưa thích (INTEGER).

Để có một danh sách các kiểu dữ liệu được hỗ trợ, xem Phụ lục A.

- Đối với DB2, bạn xác định độ dài, các độ chính xác hoặc các thang đo đối với các kiểu dữ liệu được tham số hóa như thường và những điều này hạn chế các giá trị của các tham số.
- Khi bạn tạo một dự án DB2 trong Data Project Explorer, ngoài các thư mục cho các gói PL/SQL, các kịch bản lệnh SQL, các thủ tục lưu sẵn, các hàm do người dùng định nghĩa, các đối tượng XML, bạn thấy một thư mục Các dịch vụ Web (Web Services), như trong Hình 45.

Hình 45. Xem thư mục Các dịch vụ Web trong một dự án phát triển dữ liệu cho DB2



Các hành động liên kết với thư mục Web Services giúp bạn tạo ra các dịch vụ Web chạy các câu lệnh theo kịch bản lệnh SQL hoặc gọi các thủ tục được lưu sẵn và các hàm do người dùng định nghĩa.

Để tạo ra một dịch vụ web, nhấn chuột phải vào thư mục Web Services và chọn **New Web Service...** từ trình đơn hiện lên, để chạy trình thủ thuật. Sau đó bạn có thể kéo các kịch bản lệnh SQL và các thủ tục được lưu sẵn vào dịch vụ mới.

Bạn không thể sao chép hoặc kéo một gói PL/SQL vào trong dịch vụ Web để tạo ra một hoạt động gọi thủ tục trong gói PL/SQL. Để tạo một hoạt động dịch vụ Web gọi một thủ tục trong một gói PL/SQL, hãy tạo một dịch vụ bù nhìn (dummy) cho thủ tục khác, sau đó soạn thảo hoạt động đó để thêm tên gói PL/SQL như là một từ hạn định ở giữa, ví dụ:

```
TEAM.INVENTORY.TOTAL_INVENTORY_LEVELS
```

Sau khi bạn tạo dự án phát triển dữ liệu DB2 của mình, bạn có thể bắt đầu phát triển, nhờ sử dụng PL/SQL trong một dự án DB2 bằng cách kéo hoặc sao chép các đối tượng PL/SQL từ dự án Oracle sang dự án DB2 của bạn. Hoặc, bạn có thể sao chép hoặc kéo các đối tượng từ kết nối cơ sở dữ liệu Oracle của bạn vào Data Source Explorer.

Kéo một đối tượng PL/SQL từ dự án Oracle sang dự án DB2

Sau khi bạn đã tạo một gói PL/SQL hoặc thủ tục hoặc hàm độc lập trong một dự án phát triển dữ liệu với Oracle, bạn có thể sao chép hoặc kéo nó sang dự án Oracle khác hoặc vào trong một dự án phát triển dữ liệu với DB2 LUW 9.7.

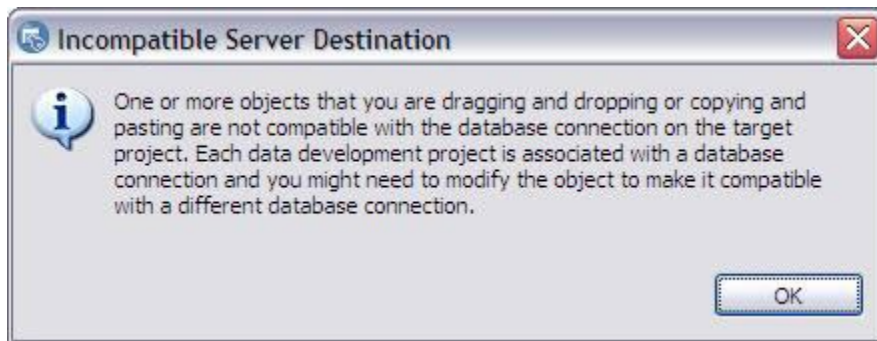
Sau khi bạn đã tạo một gói PL/SQL hoặc thủ tục hoặc hàm độc lập trong một dự án phát triển dữ liệu với Oracle, bạn có thể tìm thấy nó trong Data Source explorer và sao chép hoặc kéo nó vào một dự án phát triển dữ liệu với Oracle hoặc DB2 LUW 9.7.

Thực hiện theo các bước sau để kéo một đối tượng PL/SQL từ Oracle vào một dự án DB2:

1. Từ Data Source Explorer, mở rộng kết nối Oracle của bạn và tìm gói PL/SQL có tên là INVENTORY.

2. Điều chỉnh bàn làm việc của bạn sao cho nhìn thấy được cả Data Source Explorer và cả Data Project Explorer.
3. Kéo gói PL/SQL từ Data Source Explorer vào các thư mục PL/SQL Packages của dự án DB2 của bạn trong Data Project Explorer. (Nếu cả Data Source Explorer và cả Data Project Explorer không thể nhìn thấy được, bạn có thể sao chép từ nguồn và dán vào dự án đích).
4. Khi hoạt động kéo và thả hoàn thành, bạn sẽ được thông báo rằng có thể tồn tại một sự không tương thích, như trong Hình 46.

Hình 46. Chấp nhận một sự không tương thích có thể tồn tại giữa các dự án



Nhấn **OK**.

5. Mở gói PL/SQL có tên là INVENTORY mà bạn đã chuyển sang dự án DB2 của bạn bằng cách nhấn đúp vào nó, hoặc bằng cách nhấn chuột phải và chọn **Open** từ trình đơn thả xuống.
6. DB2 không giống như độ chính xác và thang đo trên khai báo biến X NUMBER(38,0). Thay đổi tất cả các kiểu dữ liệu để khớp với các kiểu

trong bảng INVENTORY_LEVELS trên DB2, trong tab Đặc tả của trình soạn thảo:

```
CREATE OR REPLACE PACKAGE INVENTORY
```

```
AS
```

```
FUNCTION QUERY_CLOSING_INVENTORY(  
    Y SMALLINT,  
    M SMALLINT,  
    W INTEGER,  
    P INTEGER  
    )  
RETURN INTEGER;
```

```
END INVENTORY;
```

7. Hãy thực hiện các thay đổi tương ứng trong tab Body:

```
CREATE OR REPLACE PACKAGE BODY INVENTORY
```

AS

FUNCTION QUERY_CLOSING_INVENTORY(

Y SMALLINT,

M SMALLINT,

W INTEGER,

P INTEGER

)

RETURN INTEGER

AS

X INTEGER;

BEGIN

SELECT CLOSING_INVENTORY INTO X

FROM INVENTORY_LEVELS

WHERE INVENTORY_YEAR = Y

AND INVENTORY_MONTH = M

AND WAREHOUSE_BRANCH_CODE = W

AND PRODUCT_NUMBER = P;

RETURN X;

END;

END INVENTORY;

8. Nếu bạn đã kéo một đối tượng PL/SQL bao gồm một kiểu dữ liệu ký tự, thì bạn cần phải chắc chắn đặt các độ dài chuỗi tối đa khi bạn soạn thảo các kiểu dữ liệu. Đối với Oracle, chiều dài mặc định cho CHAR là 2000. Đối với DB2, chiều dài mặc định cho CHAR là 1.
9. Lưu các thay đổi của bạn bằng cách nhấn **Ctrl+S** hoặc bằng cách chọn **File** -> **Save** trên trình đơn chính hoặc từ trình đơn hiện lên của trình soạn thảo trình. Cả hai đặc tả và phần thân được lưu.

Triển khai, chạy và gỡ rỗi trên DB2

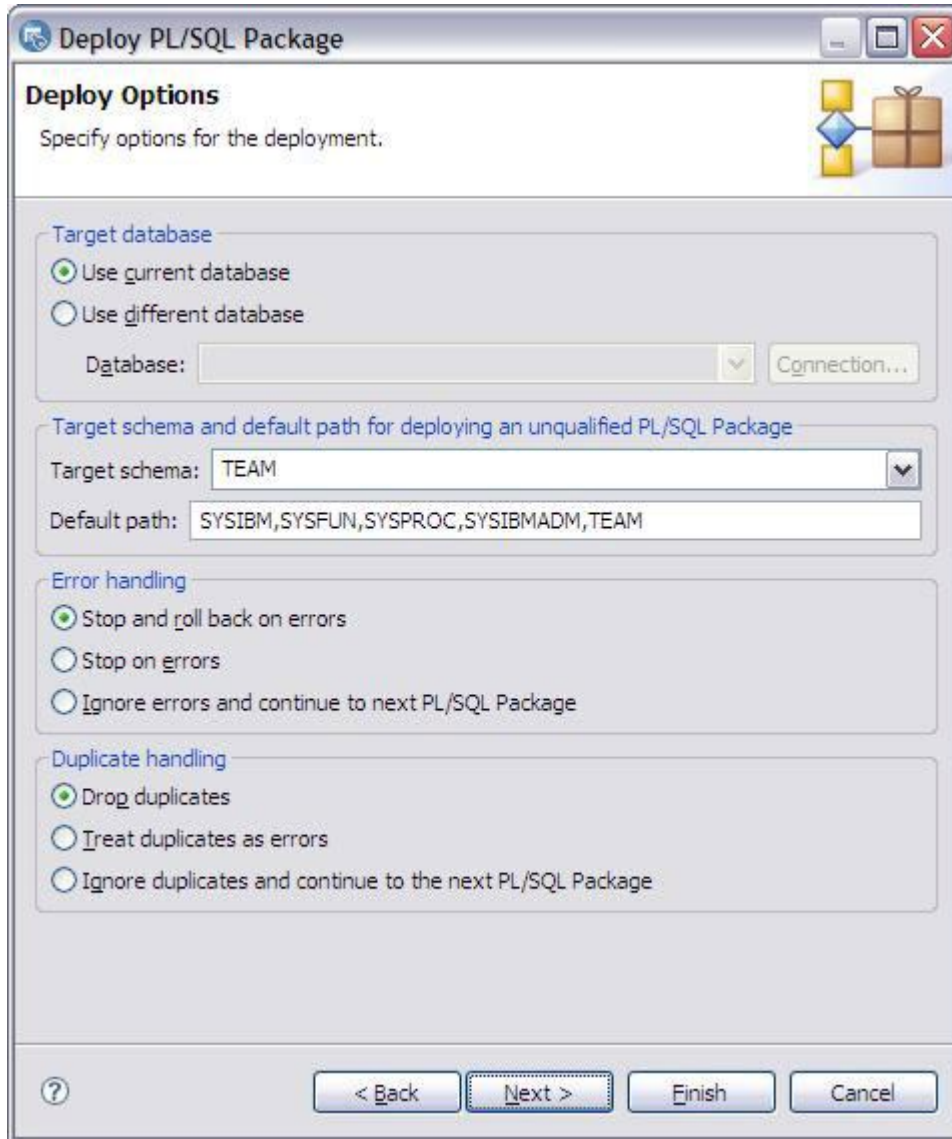
Triển khai, chạy và gỡ rỗi các gói và thường trình PL/SQL trên DB2 tương tự như thực hiện các nhiệm vụ này trên Oracle hoặc thực hiện các nhiệm vụ này cho các thường trình SQL và Java trên DB2.

Thực hiện theo các bước sau để triển khai, chạy và gỡ rỗi một trình thủ thuật PL/SQL Package để tạo một gói PL/SQL:

1. Khi bạn chọn một gói, thủ tục hoặc hàm PL/SQL trong một dự án phát triển dữ liệu và sau đó chọn hành động **Deploy...** từ trình đơn hiện lên của đối tượng và nếu bạn được kết nối với cơ sở dữ liệu liên kết với dự án đó, thì sau đó trình thủ thuật Deploy PL/SQL Package xuất hiện, như trong Hình

47.

Hình 47. Chọn các tùy chọn cho các gói PL/SQL để triển khai cho DB2



Trường đường dẫn mặc định sẽ xuất hiện trong nhóm thứ hai (Lược đồ đích...) chỉ cho DB2. Trường này không xuất hiện với Oracle.

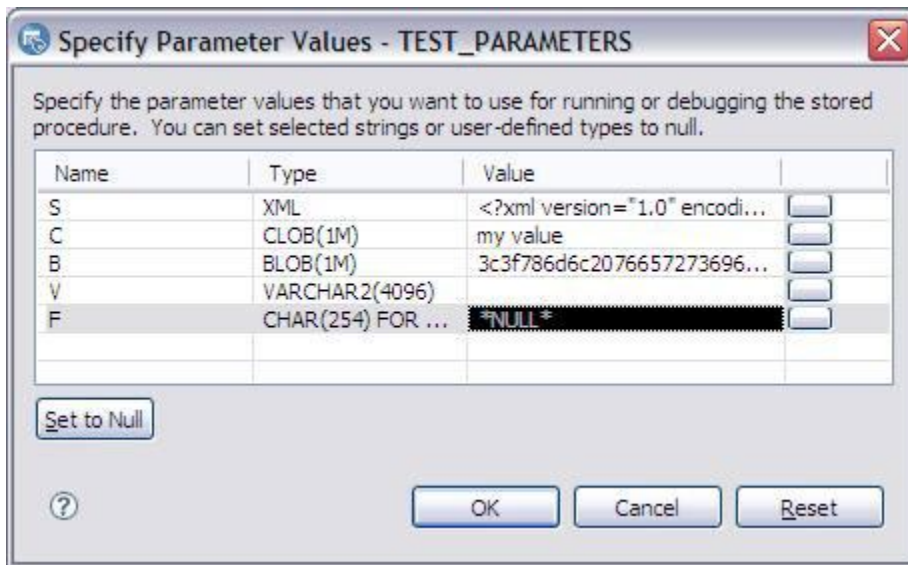
Nhóm thứ ba (Xử lý lỗi) xuất hiện trên trang này chỉ khi bạn đã chọn nhiều hơn một gói PL/SQL để triển khai với hành động này.

2. Khi phát triển, một thói quen tốt là đánh dấu chọn hộp kiểm tra **Enable debugging** trên trang Các tùy chọn gói PL/SQL (PL/SQL Package Options) của trình thủ thuật Triển khai (Deploy) để cho DB2 biên dịch đối tượng với các thông tin gỡ rối
3. Nhấn **Finish**. Các đối tượng được chọn sẽ được triển khai tuần tự trong một đoạn nền. Các kết quả được báo cáo trong khung nhìn SQL Results.
4. Bạn có thể chạy một thường trình độc lập (bất kể ngôn ngữ thực hiện của nó) trong Data Source Explorer. Trong Data Project Explorer, nếu bạn có thể triển khai nó, thì sau đó bạn cũng có thể chạy nó. Nếu thường trình khai báo các tham số đầu vào, thì sau đó bạn được nhắc cung cấp giá trị cho chúng.
5. Các kiểu dữ liệu tham số được hiển thị trong hộp thoại chạy được các máy chủ cơ sở dữ liệu chuẩn hóa. Các máy chủ DB2 chuẩn hóa các kiểu dữ liệu PL/SQL - cụ thể cho các kiểu SQL tương đương của chúng. Ví dụ, `BINARY_INTEGER` được lưu trữ trên một cơ sở dữ liệu DB2 như là `INTEGER`.
6. Bạn cũng có thể chạy một thường trình trong một gói PL/SQL trong Data Source Explorer hoặc trong Data Project Explorer.
7. ODS 2.2 cung cấp một trình gỡ rối thường trình PL/SQL cho DB2 LUW 9.7. Không giống như các trình gỡ rối thường trình SQL và Java trong ODS 2.2, trình gỡ rối PL/SQL cho DB2 không yêu cầu bạn thiết lập hoặc bắt đầu một trình quản lý của phiên gỡ rối. Trình quản lý của phiên gỡ rối được tích hợp với máy khách của trình gỡ rối và khởi động tự động.

Xác định các giá trị đầu vào là dài hoặc nhị phân

Hộp thoại Chỉ định các giá trị tham số (Specify Parameter Values) có thêm chức năng xác định và nhập khẩu chuỗi ký tự lớn, nhị phân và các giá trị XML. Bạn có thể soạn thảo một ô trong một hộp thoại thứ cấp bằng cách nhấn vào nút chấm lửng trong cột gần bên phải cho tham số đó, như trong Hình 48.

Hình 48. Nhấn vào nút dấu chấm lửng để xác định giá trị dài



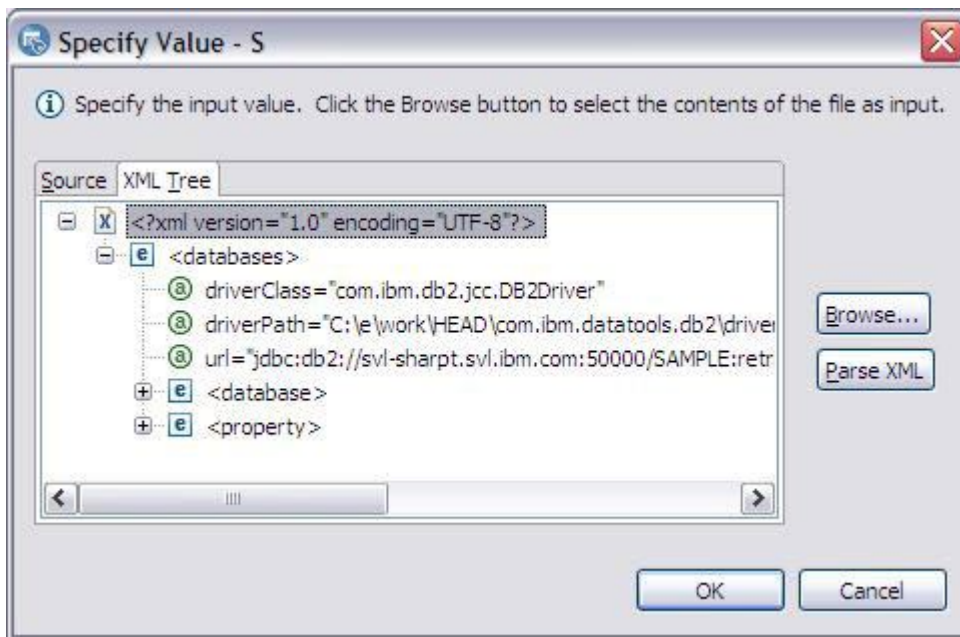
Bạn có thể gõ một giá trị số hoặc chuỗi ký tự trong ô bảng. Đối với một giá trị chuỗi-ký tự, bạn không nhập các dấu phân cách chuỗi. (Ví dụ, bạn sẽ nhập my value, không phải 'my value').

Để đặt NULL cho một giá trị chuỗi-ký tự (phân biệt NULL với một chuỗi rỗng), nhấn thiết lập rỗng (**Set to NULL**). Giá trị giả *NULL* được hiển thị và một NULL được chuyển tới thường trình đó.

Nếu nút dấu chấm lửng xuất hiện, bạn có thể nhấn vào nó để có được một hộp thoại lớn hơn để soạn thảo giá trị này.

Đối với một kiểu dữ liệu XML, nhấn **Browse...** để chọn một tệp XML và xem nó trong tab XML Tree, như trong Hình 49.

Hình 49. Chọn một tài liệu XML cho một tham số đầu vào



Đối với một kiểu dữ liệu nhị phân, bạn phải nhập một chuỗi hệ đếm mười sáu không có các dấu phân cách. Bất kỳ tệp nào mà bạn chọn với **Browse...** sẽ được biến đổi sang hệ đếm mười sáu, như trong Hình 50.

Hình 50. Chọn một tệp nhị phân cho một tham số đầu vào



Các kiểu dữ liệu tham số được hiển thị trong hộp thoại chạy được máy chủ cơ sở dữ liệu chuẩn hóa. Các máy chủ DB2 chuẩn hóa các kiểu dữ liệu PL/SQL-cụ thể theo các kiểu SQL tương đương của chúng. Các máy chủ Oracle chuẩn hóa các kiểu con (Subtypes) theo các kiểu cơ bản của chúng. Ví dụ, INT được lưu trữ trên một cơ sở dữ liệu Oracle như NUMBER, sao cho hộp thoại chạy cho thấy NUMBER thay vì INT.

Chi tiết lập trình

Sử dụng các kiểu CURSOR trong PL/SQL

Một thủ tục PL/SQL không hoàn toàn trả về tất cả các con trỏ mở, khi thực hiện các thủ tục SQL riêng trong DB2. Thay vào đó, bạn cần phải khai báo một tham số OUT của kiểu CURSOR hoặc REF CURSOR (như SYS_REFCURSOR) và mở nó bằng một truy vấn cơ sở dữ liệu. Giá trị đầu ra này được một ứng dụng JDBC thu nhận như là một tập kết quả.

Ví dụ, với một thủ tục PL/SQL trên Oracle:

```
CREATE OR REPLACE PROCEDURE TEST_SYSREFCURSOR (
```

```
    cursor_1 OUT SYS_REFCURSOR )
```

```
AS
```

```
BEGIN
```

```
OPEN cursor_1 FOR SELECT OBJECT_NAME  
  
FROM USER_PROCEDURES  
  
WHERE OBJECT_NAME LIKE 'SP0%';  
  
END;
```

Để cho phép chương trình gọi xử lý con trỏ, bạn phải để nó mở.

Kiểm soát luồng

Phần thân của một thường trình PL/SQL được chứa trong một khối PL/SQL, khối này bắt đầu bằng BEGIN và kết thúc bằng END tiếp theo là tên của thường trình theo tùy chọn.

Trong phần thân của một thường trình PL/SQL, bạn có thể sử dụng các cấu trúc điều khiển tuần tự, lặp và rẽ nhánh.

```
IF (condition) THEN...[ELSEIF (condition) THEN...][ELSE...] END IF;
```

```
CASE condition WHEN value THEN...[WHEN value THEN...] ELSE...END  
CASE;
```

```
FOR counter IN range LOOP...END LOOP;
```

WHILE *condition* LOOP...END LOOP;

WHILE...EXIT WHEN *condition*;

Ngoài ra, PL/SQL cho phép bạn mã hóa một trình xử lý ngoại lệ ở phần cuối của một khối:

EXCEPTION

WHEN *exception-name* THEN ...

[WHEN *exception-name* THEN ...]

Xem "Xử lý các lỗi và các ngoại lệ" trong hướng dẫn của nhà phát triển hai ngày cơ sở dữ liệu Oracle (*Oracle Database 2 Day Developer's Guide*) với một danh sách các tên ngoại lệ phổ biến. Có một liên kết tới hướng dẫn này trong phần Tài nguyên.

Ngoài ra, bạn có thể khai báo các ngoại lệ tùy chỉnh và nâng cao chúng theo cách lập trình:

exception-name EXCEPTION;

IF *condition* THEN RAISE *exception-name*;

Các quy ước đặt tên

Đối với PL/SQL, về cơ bản bạn có thể thực hiện theo chuẩn SQL liên quan đến việc đặt tên. Tất cả các tên cho các đối tượng Oracle có chiều dài tối đa là 30, ngoại trừ với tên cơ sở dữ liệu, có chiều dài tối đa là 8.

Trong một câu lệnh SQL hoặc PL/SQL, một trình nhận dạng SQL có thể không có khoảng trống hay các ký tự đặc biệt. Các tên này được máy chủ cơ sở dữ liệu xếp theo chữ hoa, về cơ bản làm cho các tên này không phân biệt theo dạng chữ.

Các tên trong SQL hoặc PL/SQL với các khoảng trống, các ký tự đặc biệt hoặc chữ thường phải được phân cách bằng các dấu ngoặc kép. Các dấu phân cách này không được lưu trữ trong danh mục cơ sở dữ liệu.

Lược đồ này cho chúng ta hai định dạng tên:

Định dạng SQL

myName

Định dạng danh mục

MYNAME

MYNAME MYNAME

"myName" myName

"my.name" my.name

Trong ODS 2.2, bạn thấy các tên SQL theo định dạng danh mục trong Data Source Explorer và bạn nhìn thấy các tên SQL theo định dạng SQL trong Data Project Explorer.

Các tên của các gói và các thường trình độc lập PL/SQL có thể được đánh giá. Trong DB2, từ hạn định là một tên lược đồ, tên này cũng là một từ định danh SQL. Trong Oracle, từ hạn định này là một tên chủ sở hữu. Từ hạn định này đứng trước tên đơn giản của đối tượng và được tách ra khỏi nó bằng một dấu chấm:

"mySchema"."myName"

Các gói PL/SQL giới thiệu thành phần khác, vì gói này là phần tử của lược đồ cơ sở dữ liệu, không là các thường trình trong gói đó. Để thử nghiệm một thường trình trong một gói, bạn có thể mô tả tên thường trình với tên gói trong câu lệnh CALL:

```
CALL "mySchema"."myPackage"."myName"();
```

Oracle giới thiệu định dạng đặt tên khác, mà ODS 2.2 không hỗ trợ nó: các tên từ xa. Một tên từ xa có tên của cơ sở dữ liệu khác tiếp theo tên đơn giản được phân cách bằng một dấu @:

```
CALL "mySchema"."myPackage"."myName"@otherDB();
```

Các hạn chế

- Không phải tất cả chức năng PL/SQL được ODS 2.2 hỗ trợ:
 - Xem bảng các kiểu dữ liệu trong Phụ lục A đối với các kiểu dữ liệu không được hỗ trợ.
 - Đối với DB2, bạn phải khai báo các độ dài, các độ chính xác và các thang đo của các kiểu dữ liệu dùng như tham số trong thường trình.
 - IBM không hỗ trợ biên dịch và thực hiện câu lệnh PL/SQL cho các ấn bản sản phẩm sau:
 - DB2 Express
 - DB2 Express-C
 - DB2 Personal Edition (Phiên bản cá nhân DB2)

- DB2 không cho phép bạn tạo ra các hàm và các bộ kích hoạt PL/SQL trong một môi trường cơ sở dữ liệu được phân vùng.
- Đối với DB2, một khai báo TYPE không được hỗ trợ trong một hàm, thủ tục, bộ kích hoạt hoặc khối ẩn danh.
- Đối với DB2, tùy chọn FOR EACH STATEMENT không được hỗ trợ cho các bộ kích hoạt PL/SQL.
- Không phải tất cả các chức năng đã hỗ trợ cho các thường trình SQL và Java cho DB2 được hỗ trợ cho PL/SQL:
 - Không có chức năng xuất khẩu hoặc triển khai dòng lệnh cho các đối tượng PL/SQL.
 - Không có mã xử lý lỗi được tạo ra như cho các thủ tục DB2 SQL.
 - Bạn không thể thêm một câu lệnh SQL trong trình thủ thuật khi tạo một thủ tục lưu sẵn PL/SQL. Mã được trình thủ thuật tạo ra không dựa vào việc truy cập cơ sở dữ liệu cụ thể của bạn, nhưng bị hạn chế với các khai báo về các tham số.
 - Không có sự trợ giúp nội dung nào trong trình soạn thảo SQL hoặc trình soạn thảo thường trình cho PL/SQL.
 - Không có sự hỗ trợ cho Các giá trị cài đặt hoạt động (Run Settings).
 - Không có trình thủ thuật nhập khẩu vào được cung cấp cho các gói, các thủ tục hoặc hàm PL/SQL
 - Bạn không thể sao chép hoặc kéo một gói PL/SQL vào trong một dịch vụ Web để tạo ra một hoạt động gọi thủ tục trong gói PL/SQL.

- Không phải tất cả chức năng DB2 hoặc Oracle được ODS 2.2 hỗ trợ:
 - Không có sự phát triển của các hàm Java nào trong một dự án phát triển dữ liệu với Oracle.
 - Không có sự phát triển của các bộ kích hoạt trong một dự án phát triển dữ liệu, trừ trường hợp mà bạn có thể chạy một câu lệnh CREATE TRIGGER trong một kịch bản lệnh SQL trong trình soạn thảo SQL.
 - Không hỗ trợ các dự án phát triển dữ liệu cho các mô-đun DB2.
 - Không hỗ trợ cho các tên từ xa trên Oracle.
- ODS 2.2 không sử dụng khung nhìn Outline để hiển thị các thường trình trong một gói PL/SQL.
- Bạn không thể sao chép hoặc kéo một thường trình độc lập từ một dự án hoặc cơ sở dữ liệu vào trong một gói PL/SQL trong một dự án. Bạn không thể kéo một thường trình trong một gói PL/SQL (hoặc từ một dự án hoặc từ một cơ sở dữ liệu) vào trong một thư mục Các thủ tục lưu sẵn (Stored Procedures) hoặc thư mục Hàm do người dùng định nghĩa (User-Defined Function). .
- Bạn có thể không nhất thiết phải sao chép các đối tượng giữa các cơ sở dữ liệu Oracle và DB2 mà không cần soạn thảo chúng do các sự khác nhau về chức năng. Nếu một ràng buộc cụ thể, ví dụ, không được máy hỗ trợ, nhưng ràng buộc ấy vẫn có thể được tạo ra, nó sẽ có. Một cảnh báo sẽ hiện trên khung nhìn SQL Results. Nếu không thể tạo được ràng buộc này, một lỗi sẽ xuất hiện.

- ODS 2.2 sao chép các đối tượng PL/SQL mà không dịch chúng cho dự án đích. Nếu một gói, thủ tục hoặc hàm Oracle PL/SQL chứa cú pháp hoặc chức năng không được hỗ trợ trên DB2, ODS 2.2 không thể thực hiện đối tượng đã được sao thành công. Bạn sẽ phải sửa vấn đề này trước khi bạn có thể triển khai một đối tượng PL/SQL thành công.
- Bạn không thể sử dụng trình soạn thảo Data Object để tạo ra các vùng bảng BIGFILE tuy nhiên, bạn có thể sử dụng hành động của trình đơn Mở trình soạn thảo SQL (Open with SQL editor) để soạn thảo DDL đã tạo ra và thêm từ khóa BIGFILE.
- Các trình phân tích cú pháp SQL hiện có không hỗ trợ PL/SQL. Trong ODS 2.2, các trình phân tích cú pháp được sử dụng để đánh giá trực tiếp mã nguồn của bạn trong các trình soạn thảo, để nhập khẩu và để giải thích các câu lệnh SQL trong trình thủ thuật. Một trình phân tích cú pháp PL/SQL không được bao gồm trong bản phát hành này; do đó, các trình soạn thảo có những hạn chế sau đây:
 - Các trình soạn thảo PL/SQL không cung cấp sự hỗ trợ nội dung hoặc đánh dấu các lỗi. Đối với một trình soạn thảo thường trình, việc xác nhận hợp lệ cú pháp là tắt theo mặc định; tuy nhiên, đối với trình soạn thảo kịch bản lệnh SQL, bạn nên xóa dấu chọn trên hộp chọn Xác nhận hợp lệ cú pháp của câu lệnh (Validate Statement Syntax) trong trình đơn của trình soạn thảo khi bạn soạn thảo một kịch bản có chứa PL/SQL.
 - Trình soạn thảo gói PL/SQL không đồng bộ các thay đổi giữa đặc tả và phần thân.

- Để xóa một thường trình trong một gói PL/SQL, bạn cần phải soạn thảo gói PL/SQL và loại bỏ thường trình khỏi cả đặc tả và cả phần thân.
- Bạn không thể cập nhật các tham số của thường trình PL/SQL độc lập trong tab Configuration của trình soạn thảo.
- Trình thủ thuật thủ tục PL/SQL này không bao gồm trang các câu lệnh SQL (SQL statements).
- Mặc dù có một trang các câu lệnh SQL trong trình thủ thuật hàm PL/SQL, nó không xác định kiểu trả về từ câu lệnh hoặc biểu thức SQL của bạn và nó không cung cấp các nút **Import**, **Validate** hoặc **Create SQL**.
- Việc gỡ rối các thường trình PL/SQL có những hạn chế sau:
 - Việc gỡ rối các thường trình PL/SQL yêu cầu một kết nối khi sử dụng một trình điều khiển JDBC mởng Oracle.
 - Bạn có thể gỡ rối một hàm PL/SQL chỉ khi nó ở trong một gói PL/SQL. Bạn không thể gỡ rối bất kỳ hàm độc lập nào, bất kể ngôn ngữ thực hiện của nó.
 - Bạn không thể cập nhật một phần tử LOB (biến LOB, trường LOB trong một biến hàng hoặc phần tử LOB trong một biến mảng) trong khi gỡ rối.
 - Trình gỡ rối sẽ hiển thị chỉ 32 K byte đầu tiên của một giá trị LOB.
 - Trình gỡ rối không hỗ trợ một mảng các hàng.

Phụ lục A. Các kiểu dữ liệu được hỗ trợ

Bảng sau đây cho thấy các kiểu dữ liệu mà ODS 2.2 hỗ trợ cho DB2 LUW 9.7 và Oracle 10g và 11g. Nếu kiểu dữ liệu *được hỗ trợ*, có nghĩa là bạn có thể chọn từ đồng nghĩa đầu tiên trong một trình thủ thuật cho kiểu dữ liệu của một kiểu tham số hoặc kiểu trả về và bạn có thể gỡ bất cứ từ đồng nghĩa nào của nó trong trình soạn thảo thường trình.

Các kiểu dữ liệu PL/SQL và DB2 từ định nghĩa cơ sở dữ liệu DB2 9.7 được ODS 2.2 hỗ trợ (chữ in đậm chỉ ra kiểu đó là duy nhất với DB2)	Các kiểu dữ liệu PL/SQL từ định nghĩa cơ sở dữ liệu Oracle được ODS 2.2 hỗ trợ (chữ in đậm chỉ ra kiểu đó là duy nhất với PL/SQL)
---	--

- | | |
|--|--|
| <ul style="list-style-type: none">• BIGINT• BINARY_INTEGER• BLOB• CHAR, CHARACTER• CHAR FOR BIT DATA, CHARACTER FOR BIT DATA• CLOB, CHARACTER LARGE OBJECT, CHAR LARGE | <ul style="list-style-type: none">• BINARY_DOUBLE• BINARY_FLOAT• BINARY_INTEGER• CHAR, CHARACTER• DATE• DECIMAL, DEC• DOUBLE PRECISION |
|--|--|

- **OBJECT**
- **CURSOR** ¹
- **DATE** ²
- **DBCLOB**
- **DECFLOAT**
- **DECIMAL, DEC**
- **DOUBLE, DOUBLE PRECISION**
- **FLOAT**
- **GRAPHIC**
- **INTEGER, INT**
- **LONG RAW**
- **NCHAR, NATIONAL CHARACTER, NATIONAL CHAR**
- **NCLOB** ³
- **NUMBER** ⁴
- **NUMERIC, NUM**
- **NVARCHAR2, NATIONAL**
- **FLOAT**
- **INT, INTEGER**
- **LONG RAW**
- **NATURAL**
- **NUMBER**
- **NUMERIC**
- **PLS_INTEGER**
- **POSITIVE**
- **RAW**
- **REAL**
- **ROWID**
- **SIGNTYPE**
- **SMALLINT**
- **STRING**
- **SYS_REFCURSOR, REFCURSOR, REF CURSOR** ¹
- **TIMESTAMP**
- **TIMESTAMP WITH LOCAL**

- CHARACTER VARYING,
NATIONAL CHAR VARYING,
NCHAR VARYING
- PLS_INTEGER
 - RAW
 - REAL
 - SMALLINT
 - SYS_REFCURSOR,
REFCURSOR, REF CURSOR ¹
 - **TIME**
 - **TIMEZONE** ⁶
 - **TIMESTAMP WITH TIME ZONE**
 - VARCHAR2, CHARACTER VARYING, CHAR VARYING, VARCHAR
 - **VARCHAR FOR BIT DATA, CHARACTER VARYING FOR BIT DATA, CHAR VARYING FOR BIT DATA**
 - VARCHAR2 ⁵
 - **VARGRAPHIC**

Đây là những kiểu dữ liệu mà ODS 2.2 không hỗ trợ.

Các kiểu dữ liệu PL/SQL không được hỗ trợ cho DB2 **Các kiểu dữ liệu PL/SQL không được hỗ trợ cho Oracle**

- | | |
|---|--|
| <ul style="list-style-type: none">• BINARY_DOUBLE | <ul style="list-style-type: none">• BFILE⁷ |
| <ul style="list-style-type: none">• BINARY_FLOAT | <ul style="list-style-type: none">• BLOB⁷ |
| <ul style="list-style-type: none">• BOOLEAN (JCC does not support BOOLEAN for PL/SQL) | <ul style="list-style-type: none">• BOOLEAN, PL/SQL
BOOLEAN |
| <ul style="list-style-type: none">• BFILE | <ul style="list-style-type: none">• CLOB⁷ |
| <ul style="list-style-type: none">• INTERVAL | <ul style="list-style-type: none">• INTERVAL |
| <ul style="list-style-type: none">• INTERVAL YEAR TO MONTH | <ul style="list-style-type: none">• INTERVAL YEAR TO
MONTH |
| <ul style="list-style-type: none">• INTERVAL DAY TO SECOND | <ul style="list-style-type: none">• INTERVAL DAY TO SECOND |
| <ul style="list-style-type: none">• LONG | <ul style="list-style-type: none">• LONG |
| <ul style="list-style-type: none">• NATURALN | <ul style="list-style-type: none">• NATURALN⁸ |
| <ul style="list-style-type: none">• POSITIVE | <ul style="list-style-type: none">• NCHAR,
NATIONAL CHARACTER,
NATIONAL CHAR |
| <ul style="list-style-type: none">• POSITIVEN | |
| <ul style="list-style-type: none">• ROW | |
| <ul style="list-style-type: none">• SIGNTYPE | <ul style="list-style-type: none">• NCLOB⁷ |
| <ul style="list-style-type: none">• SIMPLE_INTEGER | <ul style="list-style-type: none">• NVARCHAR2,
NATIONAL CHARACTER |

- STRING
 - TABLE
 - TIMESTAMP WITH LOCAL TIME ZONE
 - TIMESTAMP WITH TIME ZONE
 - VARRAY
 - XMLTYPE
- VARYING,
 - NATIONAL CHAR VARYING,
 - NCHAR VARYING
 - POSITIVEN⁸
 - ROW
 - SIMPLE_INTEGER⁸
 - TABLE
 - UROWID
 - VARRAY
 - XMLTYPE⁷

Ghi chú:

1. SYS_REFCURSOR và CURSOR được hỗ trợ chỉ cho các tham số OUT (trừ nơi mà chúng đã chuyển một giá trị từ một thường trình khác). Các hàm PL/SQL do người dùng định nghĩa phải có các trả về vô hướng; do đó, các kiểu dữ liệu này không được hỗ trợ cho các hàm PL/SQL do người dùng định nghĩa.
2. Khi biến đăng ký DB2_COMPATIBILITY_VECTOR được đặt cho kiểu dữ liệu DATE , DATE tương đương với TIMESTAMP (0).
3. Kiểu dữ liệu NCLOB không được hỗ trợ để sử dụng trong các câu lệnh PL/SQL hoặc trong các ngữ cảnh PL/SQL khi cơ sở dữ liệu không được

- định nghĩa như là một cơ sở dữ liệu Unicode. Trong cơ sở dữ liệu Unicode, kiểu dữ liệu NCLOB được ánh xạ tới một kiểu dữ liệu DBCLOB.
4. NUMBER được hỗ trợ cho SQL trên DB2 khi tham số cấu hình cơ sở dữ liệu number_compat được đặt là ON.
 5. VARCHAR2 được hỗ trợ cho SQL trên DB2 khi tham số cấu hình cơ sở dữ liệu varchar2_compat được đặt là ON.
 6. DẤU THỜI GIAN VỚI VÙNG THỜI GIAN ĐỊA PHƯƠNG (TIMESTAMP WITH LOCAL TIME ZONE) không được hỗ trợ như là một kiểu trả về.
 7. BFILE, SIMPLE_INTEGER, BLOB, CLOB, NCLOB và XMLTYPE không được hỗ trợ cho Oracle bởi vì ODS không có quyền truy cập vào các lớp của trình bao (wrapper) mà các trình điều khiển Oracle yêu cầu nó.
 8. NATURALN, POSITIVEN và SIMPLE_INTEGER là các kiểu con của PLS_INTEGER với một ràng buộc NOT NULL. Điều này ngăn không cho bạn sử dụng chúng như là các tham số OUT không liệt kê chúng để tạo các tham số của thường trình, nhưng bạn có thể sử dụng chúng như là IN hoặc IN OUT trong trình soạn thảo.

Ngoài những hạn chế nêu trên, Data Studio không hỗ trợ các kiểu dữ liệu Oracle sau đây:

- Các kiểu do người dùng định nghĩa
 - Các kiểu đối tượng
 - Các bảng lồng nhau REF

- Các kiểu bất kỳ
 - ANYTYPE
 - ANYDATA
 - ANYDATASET

- Các kiểu XML
 - Các kiểu dữ liệu URI
 - Gói URIFactory

- Các kiểu khoảng trống
 - SDO_GEOMETRY
 - SDO_TOPO_GEOMETRY
 - SDO_GEORASTER
 - SI_Texture
 - SI_FeatureList

- Các kiểu phương tiện
 - ORDAudio
 - ORDImage
 - ORDImageSignature
 - ORDVideo

- ORDDoc
- SI_StillImage
- SI_Color
- SI_AverageColor
- SI_ColorHistogram
- SI_PositionalColor

Các hạn chế của kiểu dữ liệu:

- ODS 2.2 không có sự hỗ trợ cho các kiểu dữ liệu sưu tập PL/SQL (ROW và VARRAY) ngoài CURSOR. Bạn chỉ có thể khai báo về các kiểu trả về vô hướng cho các hàm PL/SQL.
- Khi sao chép dữ liệu từ Oracle vào DB2, có một số hạn chế dựa trên các kiểu dữ liệu. Dữ liệu của các kiểu Oracle sau đây không thể được sao chép trong phiên bản này:
 - TIMESTAMP WITH TIME ZONE
 - TIMESTAMP WITH LOCAL TIME ZONE
 - TIMESTAMP (*fractional-seconds-precision*)
 - RAW
 - BFILE
 - XML

- INTERVAL (không được hỗ trợ trên máy chủ DB2)
- NUMBER (4, -1) [ở đây thang đo là âm]
- NUMBER (5,10) [ở đây độ chính xác thấp hơn thang đo]
- Các kiểu do người sử dụng định nghĩa
- ROW và ARRAY (các kiểu ghép)
- Chiều dài tối đa của một BLOB hoặc CLOB trên Oracle có thể là 4 GB, nhưng ODS 2.2 và DB2 có thể chỉ xử lý lên đến 2 GB. Các LOB dài hơn sẽ được cắt ngắn.
- Khi sao chép dữ liệu từ DB2 vào Oracle, có thể có một số cắt ngắn:
 - DB2 DECFLOAT hỗ trợ một phạm vi lớn hơn FLOAT nhị phân của Oracle.
 - DB2 TIMESTAMP hỗ trợ độ chính xác lớn hơn (mặc dù nó không thể được sử dụng vì java.sql.Timestamp bị giới hạn ở một độ chính xác của phân đoạn thứ hai của 9).