

Szoftverfejlesztés az NI (Emerson) nagyvállalati környezetében – SQL, PL/SQL, ERP

SQL

- Structured Query Language
- Standard language to access relational databases
- The most basic syntax of SQL starts with a SELECT clause and a FROM clause. A SELECT clause tells the database WHAT you want to select. A FROM clause tells the database where to find that data.

```
SELECT table_name  
FROM all_tables  
ORDER BY table_name;
```

SQL - DML

- DML are the keywords you use to access and manipulate data
- SELECT, UPDATE, INSERT, MERGE, DELETE, etc

```
INSERT INTO whatever (column1, column2, column3)  
VALUES (1, 'Text Data', sysdate );
```

```
SELECT column1, column2, column3  
FROM whatever;
```

```
COMMIT;
```

SQL - DDL

- DDL are the keywords you use to create objects such as views, tables and procedures etc.
- CREATE TABLE, ALTER VIEW, CREATE OR REPLACE PROCEDURE, etc.

```
CREATE TABLE whatever (  
    column1 NUMBER NOT NULL,  
    column2 VARCHAR2(10),  
    column3 DATE );
```

PL/SQL

- PL/SQL is the Oracle native programming language that provides database-centric application development. It can natively call static SQL and provides multiple methods of calling dynamic SQL
- PL/SQL is a procedural language like C++, Java, ADA, etc. It has variables and variable declarations, conditional controls like IF and CASE. It has looping structures such as LOOP, FOR LOOP and the WHILE LOOP. PL/SQL uses SQL to use, manipulate and save data to the database.

```
BEGIN
  -- A PL/SQL cursor
  FOR c1 IN (
    SELECT * FROM whatever ) -- This is SQL called by PL/SQL
  LOOP

    DBMS_OUTPUT.PUT_LINE( 'Column1 is: ' || TO_CHAR(c1.column1) ||
                          ', Column2 is: ' || c1.column2 ||
                          ', Column3 is: ' || TO_CHAR(c1.column3) );

  END LOOP;

END;
```

Differences between SQL and PL/SQL

- SQL is a data-oriented language for selecting and manipulating sets of data. PL/SQL is a procedural language to create applications. You don't normally have a "SQL application". You normally have an application that uses SQL and a relational database on the back-end. PL/SQL can be the application language just like Java or PHP can. SQL may be the source of data for your screens, web pages and reports. PL/SQL might be the language you use to build, format and display those screens, web pages and reports.

PL/SQL - Declarations

- Variables are placeholders that store the values that can change through the PL/SQL Block.
- Local variables - These are declared in a inner block and cannot be referenced by outside Blocks.
- Global variables - These are declared in a outer block and can be referenced by its itself and by its inner blocks.

```
DECLARE
    var_num1 number;
    var_num2 number;
BEGIN
    var_num1 := 100;
    var_num2 := 200;
    DECLARE
        var_mult number;
    BEGIN
        var_mult := var_num1
*   var_num2;
    END;
END;
/
```

PL/SQL - Statements

IF THEN ELSE STATEMENT

1)

```
IF condition  
THEN  
    statement 1;  
ELSE  
    statement 2;  
END IF;
```

2)

```
IF condition 1  
THEN  
    statement 1;  
    statement 2;  
ELSIF condition2 THEN  
    statement 3;  
ELSE  
    statement 4;  
END IF;
```


PL/SQL - Statements

- Iterative Statements are used when we want to repeat the execution of one or more statements for specified number of times.
- There are three types of loops in PL/SQL:
 - Simple Loop
 - LOOP
 - statements;
 - EXIT;
 - {or EXIT WHEN condition;}
 - END LOOP;

PL/SQL - Statements

- Iterative Statements are used when we want to repeat the execution of one or more statements for specified number of times.
- There are three types of loops in PL/SQL:

– While Loop

- WHILE <condition> LOOP
- statements;
- END LOOP;

PL/SQL - Statements

- Iterative Statements are used when we want to repeat the execution of one or more statements for specified number of times.
- There are three types of loops in PL/SQL:

– For Loop

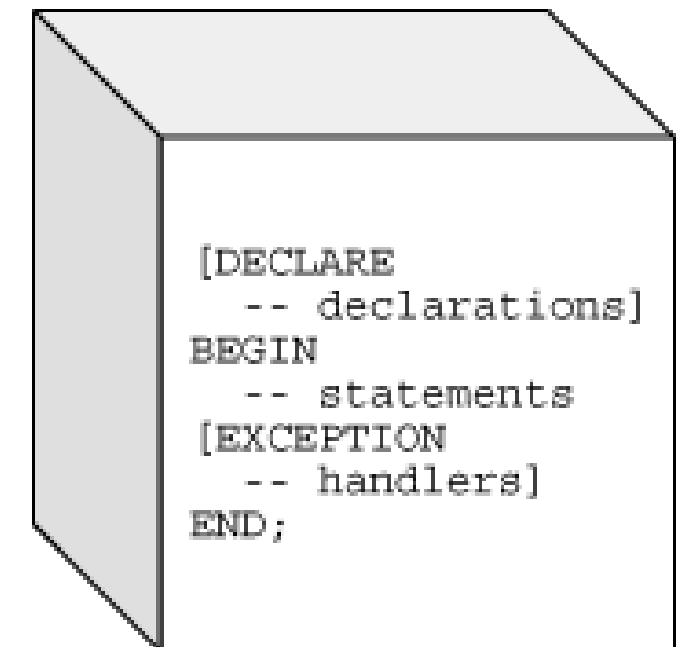
- FOR counter IN val1..val2
- LOOP statements;
- END LOOP;
- val1 - Start integer value.
- val2 - End integer value.

PL/SQL – Dynamic SQL

- **Dynamic SQL** is a programming methodology for generating and running SQL statements at run time. It is useful when writing general-purpose and flexible programs like ad hoc query systems, when writing programs that must run database definition language (DDL) statements, or when you do not know at compilation time the full text of a SQL statement or the number or data types of its input and output variables.
 - For example, a **SELECT** statement that includes an identifier that is unknown at compile time (such as a table name) or a **WHERE** clause in which the number of subclauses is unknown at compile time
- The **EXECUTE IMMEDIATE** statement builds and runs a dynamic SQL statement in a single operation.

PL/SQL – Blocks

- A PL/SQL block has three basic parts:
 - a declarative part (**DECLARE**),
 - an executable part (**BEGIN .. END**),
 - and an exception-handling (**EXCEPTION**) part that handles error conditions.
- Only the executable part is required. The optional declarative part is written first, where you define types, variables, and similar items. These items are manipulated in the executable part. Exceptions raised during execution can be dealt with in the exception-handling part.



PL/SQL – Procedure

- A **subprogram** is a program unit/module that performs a particular task. These subprograms are combined to form larger programs. This is basically called the 'Modular design'. A subprogram can be invoked by another subprogram or program which is called the calling program.
- A subprogram can be created:

- At schema level

- Inside a package

- Inside a PL/SQL block

```
CREATE OR REPLACE PROCEDURE greetings AS
BEGIN
    dbms_output.put_line('Hello World!');
END;
/
```

PL/SQL – Function

- A PL/SQL function is same as a procedure except that it returns a value.

```
CREATE OR REPLACE FUNCTION totalCustomers RETURN number
IS
    total number(2) := 0;
BEGIN
    SELECT count(*) into total
    FROM customers;

    RETURN total;
END;
/
```


PL/SQL – Package

- PL/SQL lets you bundle logically related types, variables, cursors, and subprograms into a package, a database object that is a step above regular stored procedures. The package defines a simple, clear, interface to a set of related procedures and types that can be accessed by SQL statements.
- Packages usually have two parts: a specification and a body. The specification defines the application programming interface; it declares the types, constants, variables, exceptions, cursors, and subprograms. The body fills in the SQL queries for cursors and the code for subprograms.

PL/SQL – Exception Handling

- In PL/SQL, a warning or error condition is called an exception.
- Exceptions can be internally defined (by the run-time system) or user defined. Examples of internally defined exceptions include division by zero and out of memory.
- When an error occurs, an exception is raised. That is, normal execution stops and control transfers to the exception-handling part of your PL/SQL block or subprogram.
 - Internal exceptions are raised implicitly (automatically) by the run-time system.
 - User-defined exceptions must be raised explicitly by RAISE statements, which can also raise predefined exceptions.
- To handle raised exceptions, you write separate routines called exception handlers.
- After an exception handler runs, the current block stops executing and the enclosing block resumes with the next statement.
- If there is no enclosing block, control returns to the host environment.



Oracle E-Business Suite



Integrált vállalatirányítási rendszer



ERP systems

- ERP = Enterprise Resource Planning
- Data of the business processes are collected in one system
- Automates the tasks generated by business processes
- Based on database
- Unified user interface (GUI)
- Built of modules in most cases (e.g. financial, customer, supplier, manufacturing data), generally these can be bought and installed independently from each other
- Customizable
- Appropriate for making integrated analysis, executive summary
- Can be useful for small companies
- Expected for big companies

ERP systems

- Well-known ERP systems:
 - Oracle Applications (Oracle)
 - SAP / Small Business One (SAP SBO)
 - AXAPTA (Microsoft)
 - Navision (Microsoft)
 - IFS Applications (Swedish)
 - LIBRA (Hungarian: Volán Elektronika)
 - Microsoft Dynamics
 - etc.

Modules (applications) in Oracle Applications

- Oracle Applications is an ERP system made up of modules that can be purchased and installed independently from each other
- These applications cover the business processes of the given business areas
- Modules can be divided into 2 groups:
 - **ERP:** Financials, Manufacturing, Order Management, Purchasing, Supply Chain Planning, HRMS (human resource management)
 - **CRM:** Business Intelligence, Marketing, Sales, Call Center

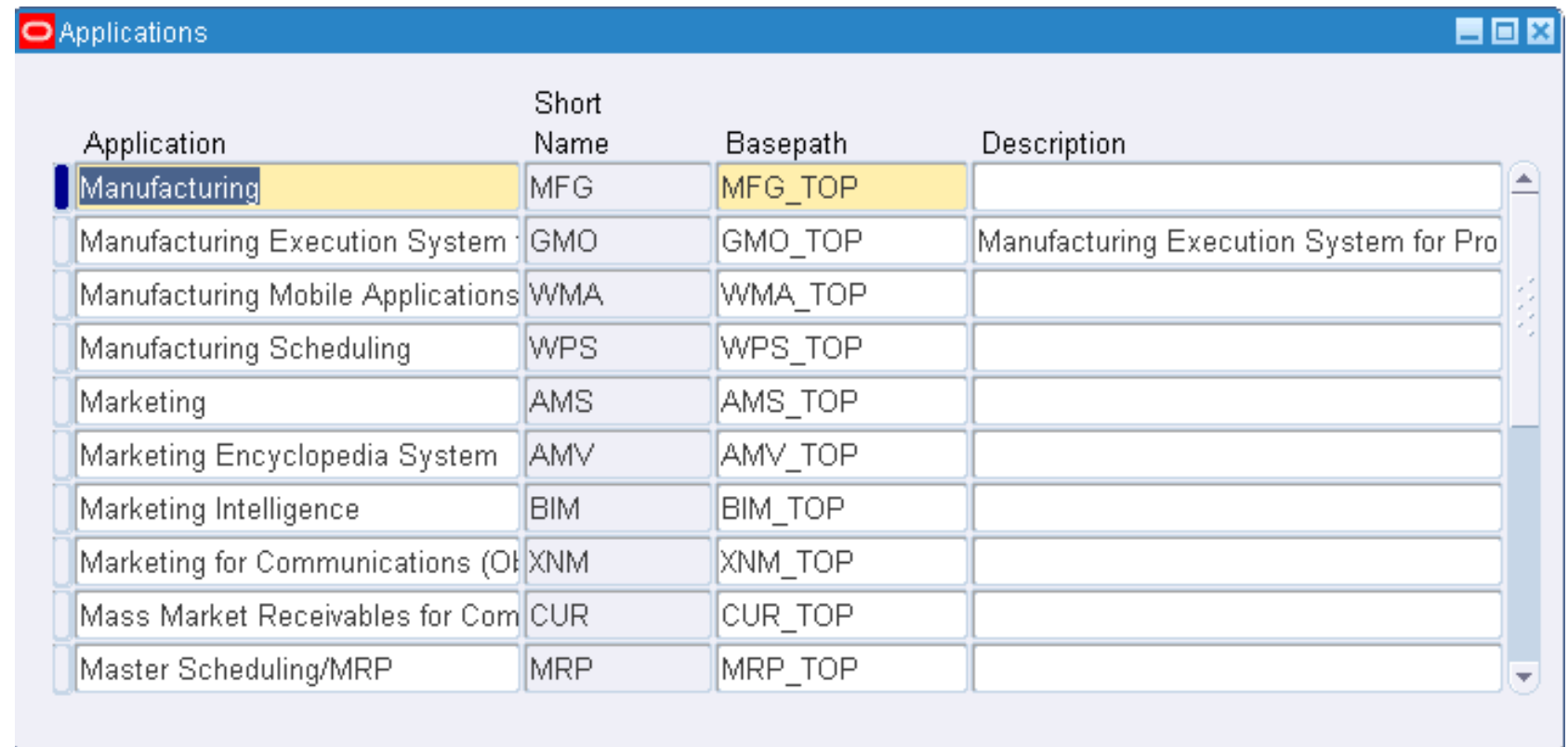
Modules (applications) in Oracle Applications

- Oracle applications:

- Inventory (INV)
- Bill of Material (BOM)
- Manufacturing (MFG)
- Order Entry (OE)
- Purchasing (PO)
- Work in Process (WIP)
- etc.

- Custom applications:

- NIINV, NIBOM, NIMFG, NIOE, NIPO, NIWIP, NIMRP, NIMSC etc.



The screenshot shows the 'Applications' window in Oracle. It contains a table with the following columns: Application, Short Name, Basepath, and Description. The 'Manufacturing' application is highlighted in yellow.

Application	Short Name	Basepath	Description
Manufacturing	MFG	MFG_TOP	
Manufacturing Execution System	GMO	GMO_TOP	Manufacturing Execution System for Pro
Manufacturing Mobile Applications	WMA	WMA_TOP	
Manufacturing Scheduling	WPS	WPS_TOP	
Marketing	AMS	AMS_TOP	
Marketing Encyclopedia System	AMV	AMV_TOP	
Marketing Intelligence	BIM	BIM_TOP	
Marketing for Communications (O	XNM	XNM_TOP	
Mass Market Receivables for Com	CUR	CUR_TOP	
Master Scheduling/MRP	MRP	MRP_TOP	

Bevezetés az Oracle EBS világába

- ERP systems
- Oracle Applications
 - Logging in, menu, responsibility
 - GUI (LOV, flexfields, querying data etc.)
 - Modules (applications)
 - Concurrent managers
 - AOL objects
Value set, Oracle Alert, Quick code, Flexfield, Concurrent program, Request set
 - How to add a new form to a menu (form, function, menu)
 - Accessible programs and forms for a user
(user, responsibility, request group, menu)

Menu, Responsibilities

- Once you open a form a new menu window opens as well...

Navigator - NI Application Developer

Functions Documents

NI Metadata:

NI Metadata

- + NI Metadata
- + Security
- + Flexfields
- + Concurrent Programs
- + Concurrent Managers
- + Applications
- + Printers
- + Profile Options
- + Alerts
- + Concurrent Requests
- NI ID Lookup Inquiry GUI
- + XML Publisher Administrator
- NI Data Migration Entity Setup

Top Ten List

1. Concurrent Programs Query Only
2. View All Concurrent Requests - Query

Open

Users

User Name: HUBUJDOA

Password:

Description:

Status: Active

Password Expiration

☒ Days 90

☐ Accesses

☐ None

Person: Bujdosó, Angelika

Customer:

Supplier:

E-Mail: angelika.bujdoso@ni.com

Fax:

Effective Dates

From: 07-NOV-2005

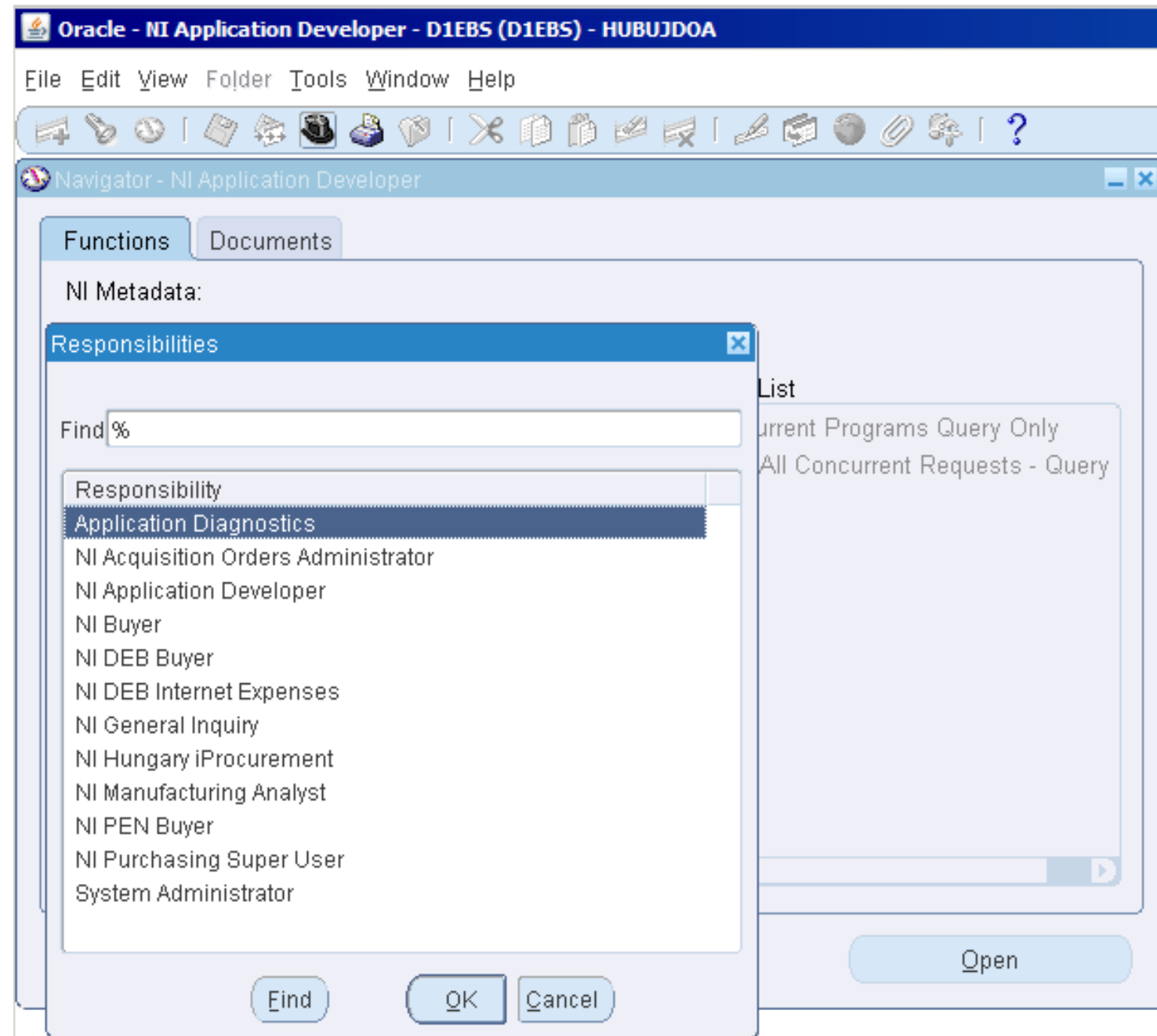
To:

Direct Responsibilities Indirect Responsibilities Securing Attributes

Responsibility	Application	Description	Security Group	Effective Dates
				From To
System Administrator	System Administration		Standard	15-NOV-2013
NI DEB Internet Expenses	Payables		Standard	14-AUG-2011
NI General Inquiry	Inventory		Standard	14-AUG-2011
Order Management Super Us	Order Management		Standard	05-MAY-2012 07-MAY-2012
NI Application Developer	NI Custom Application		Standard	14-AUG-2011

Changing responsibility

- Click on the top hat icon and choose another responsibility assigned to the current user



Oracle Applications GUI

- Meaning of colors:
- Mandatory: yellow
- Optional: white
- Disabled: grey

The screenshot shows the 'Concurrent Programs' window with the following fields and options:

- Program**: [Yellow field] ☒ Enabled
- Short Name**: [Yellow field]
- Application**: [Yellow field]
- Description**: [White field]
- Executable** section:
 - Name**: [Yellow field]
 - Method**: [White field]
 - Options**: [White field]
 - Priority**: [White field]
- Request** section:
 - Type**: [White field]
 - Incrementor**: [White field]
 - MLS Function**: [White field]
 - ☒ Use in SRS
 - ☐ Run Alone
 - ☐ Enable Trace
 - ☐ Allow Disabled Values
 - ☒ Restart on System Failure
 - ☒ NLS Compliant
- Output** section:
 - Format**: [Yellow dropdown menu with 'Text' selected]
 - ☒ Save (C)
 - ☒ Print
 - Columns**: [White field]
 - Rows**: [White field]
 - Style**: [White field]
 - ☐ Style Required
 - Printer**: [White field]
- Business Events** section:
 - ☐ Request Submitted (Y)
 - ☐ Request On Hold
 - ☐ Request Resumed
 - ☐ Request Running
 - ☐ Program Completed
 - ☐ Post Processing Started
 - ☐ Post Processing Ended
 - ☐ Request Completed (Z)
- Buttons** at the bottom: Copy to..., Session Control, Incompatibilities, Parameters

Oracle Applications GUI

- User interface elements:

- text boxes
- buttons
- checkboxes
- radio buttons
- tabs
- scrollbars
- list of values
- combo boxes

etc.

The screenshot displays the 'Organization Item (DEB)' window in the Oracle Applications GUI. The window has a blue title bar and a sidebar with icons for various functions. The main content area is divided into several sections:

- Organization Information:** Organization (DEB), NI Debrecen Mfg, Item (778004-04), and Description (MID-7652, INTEGRATED 2 AXIS SERVO DRIVE V [85]).
- Display Attributes:** Radio buttons for Master, Org (selected), and All.
- Tabs:** Main, Inventory (selected), Bills of Material, Asset Management, Costing, Purchasing, Receiving, and Physical Attributes.
- Inventory Item Section:** Includes checkboxes for Inventory Item (checked), Stockable (checked), Transactable (checked), Revision Control (D) (unchecked), Reservable (G) (checked), and Check Material Shortage (unchecked).
- Lot Expiration (Shelf Life) Section:** Includes a dropdown for Control (No Control), and text boxes for Shelf Life Days, Retest Interval, Expiration Action Interval, and Expiration Action.
- Lot Section:** Includes a dropdown for Control (No Control), and text boxes for Starting Prefix, Starting Number, Maturity Days, and Hold Days.
- Serial Section:** Includes a dropdown for Generation (Predefined), and text boxes for Starting Prefix (S) and Starting Number (00000001).
- Grade Controlled Section:** Includes a checkbox for Grade Controlled (unchecked) and a text box for Default Grade.
- Locator Control Section:** Includes a dropdown for Locator Control (No Control), and checkboxes for Restrict Subinventories and Restrict Locators (J) (unchecked).

Querying Data

- Entering „Enter Query” mode:
 - F11
- Executing the query:
 - CTRL + F11
- Leaving „Enter Query” mode:
 - F4

The screenshot displays the Oracle - NI Manufacturing Analyst - D1EBS (D1EBS) - HUBUJDOA application window. The window has a menu bar (File, Edit, View, Folder, Tools, Bills, Components, Window, Help) and a toolbar with various icons. The main form is titled "Bills of Material (DEB)". It contains several input fields: "Item", "Alternate", "Revision", "UOM", "Date", and a "Display" dropdown menu set to "Future and Current". There is also a checkbox for "Implemented Only". Below these fields is a tabbed interface with tabs: "Main", "Date Effectivity", "Unit Effectivity", "ECO", "Component Details", "Material Control", and "Order Management". The "Main" tab is active, showing a table with columns: "Item Seq", "Operation Seq", "Component", "Item Description", "Revision", "UOM", and "Basis". The table has several rows, with the first row highlighted. At the bottom of the window, there are buttons for "Substitutes", "Designators", "Operations", "Bill Details", and "Revision".

List Of Values (LOV)

- Opening a list of values (LOV): clicking on the „...” button next to the field

The screenshot displays the 'Bills of Material (DEB)' application window. The main window has a header with fields for Item, Alternate, Revision, Date (03-DEC-2013 16:16:34), and UOM. Below the header are tabs for Main, Date Effectivity, Unit Effectivity, ECO, Component Details, Material Control, and Order Management. The 'Main' tab is active, showing a table with columns for Item Seq, Operation Seq, Component, Item Description, Revision, UOM, and Basis. A dialog box titled 'Enter Reduction Criteria for Long-List' is open over the table, with a 'Part Number' field set to '778%'. Another dialog box titled 'Find Bills to Compare' is open on the right, showing a list of organizations for Bill 1 and Bill 2. The list includes ADC (NI Asia Distribution Center), ASV (NI Austin Service Virtual), CSC (NI China Service Center), DEB (NI Debrecen Mfg), DSC (NI Debrecen Service Center), DUB (NI Ireland INV), EU (NI Europe INV), EUL (NI Europe Literature), JPL (NI Japan Literature), NI (NI Item Master), ODC (NI Debrecen Olite Distribution Center), PEN (NI Penang Mfg), and PSC (NI Penang Service Center). The 'Compare' section has checkboxes for Item Seq, Quantity (checked), and Disable Date. The 'Order By' section has radio buttons for Operation Seq, Item Seq (selected) and Item Seq, Operation Seq.

Bills of Material (DEB)

Item: [] UOM: []
Alternate: []
Revision: [] Date: 03-DEC-2013 16:16:34 []
Display: Future and Current [v] Implemented Only [x]

Main | Date Effectivity | Unit Effectivity | ECO | Component Details | Material Control | Order Management

Item Seq	Operation Seq	Component	Item Description	Revision	UOM	Basis
[]	[]	[]	[]	[]	[]	[]
[]	[]	[]	[]	[]	[]	[]
[]	[]	[]	[]	[]	[]	[]
[]	[]	[]	[]	[]	[]	[]
[]	[]	[]	[]	[]	[]	[]
[]	[]	[]	[]	[]	[]	[]
[]	[]	[]	[]	[]	[]	[]
[]	[]	[]	[]	[]	[]	[]
[]	[]	[]	[]	[]	[]	[]
[]	[]	[]	[]	[]	[]	[]

Enter Reduction Criteria for Long-List

Part Number: 778%

OK Cancel Clear

Find Bills to Compare

Bill 1 Organization: DEB
Item: []
Alternate: []
Revision: []
Effective Date: []
Display Option: []

Bill 2 Organization: DEB
Item: []
Alternate: []
Revision: []
Effective Date: 03-DEC-2013 16:22:16
Display Option: Current [v]
Implemented Only [x]
Engineering Bill []

Find %

Code	Name
ADC	NI Asia Distribution Center
ASV	NI Austin Service Virtual
CSC	NI China Service Center
DEB	NI Debrecen Mfg
DSC	NI Debrecen Service Center
DUB	NI Ireland INV
EU	NI Europe INV
EUL	NI Europe Literature
JPL	NI Japan Literature
NI	NI Item Master
ODC	NI Debrecen Olite Distribution Center
PEN	NI Penang Mfg
PSC	NI Penang Service Center

Compare
☐ Item Seq
☒ Quantity
☐ Disable Date

Order By
☒ Operation Seq, Item Seq
☐ Item Seq, Operation Seq

Find OK Cancel Compare

Value Sets

Value Sets

Value Set Name

Description

List Type

Security Type

Format Validation

Format Type

☐ Numbers Only (0-9)

☐ Uppercase Only (A-Z)

☐ Right-justify and Zero-fill Numbers (0001)

Min Value

Max Value

Value Validation

Validation Type

Usages

Edit Information

Lookups

Type

NIMFG_DR_SERVICE_TYPES

Meaning

Service types for the Matrix

Application

NI Manufacturing

Description

Service types fro the Matrix

Access Level

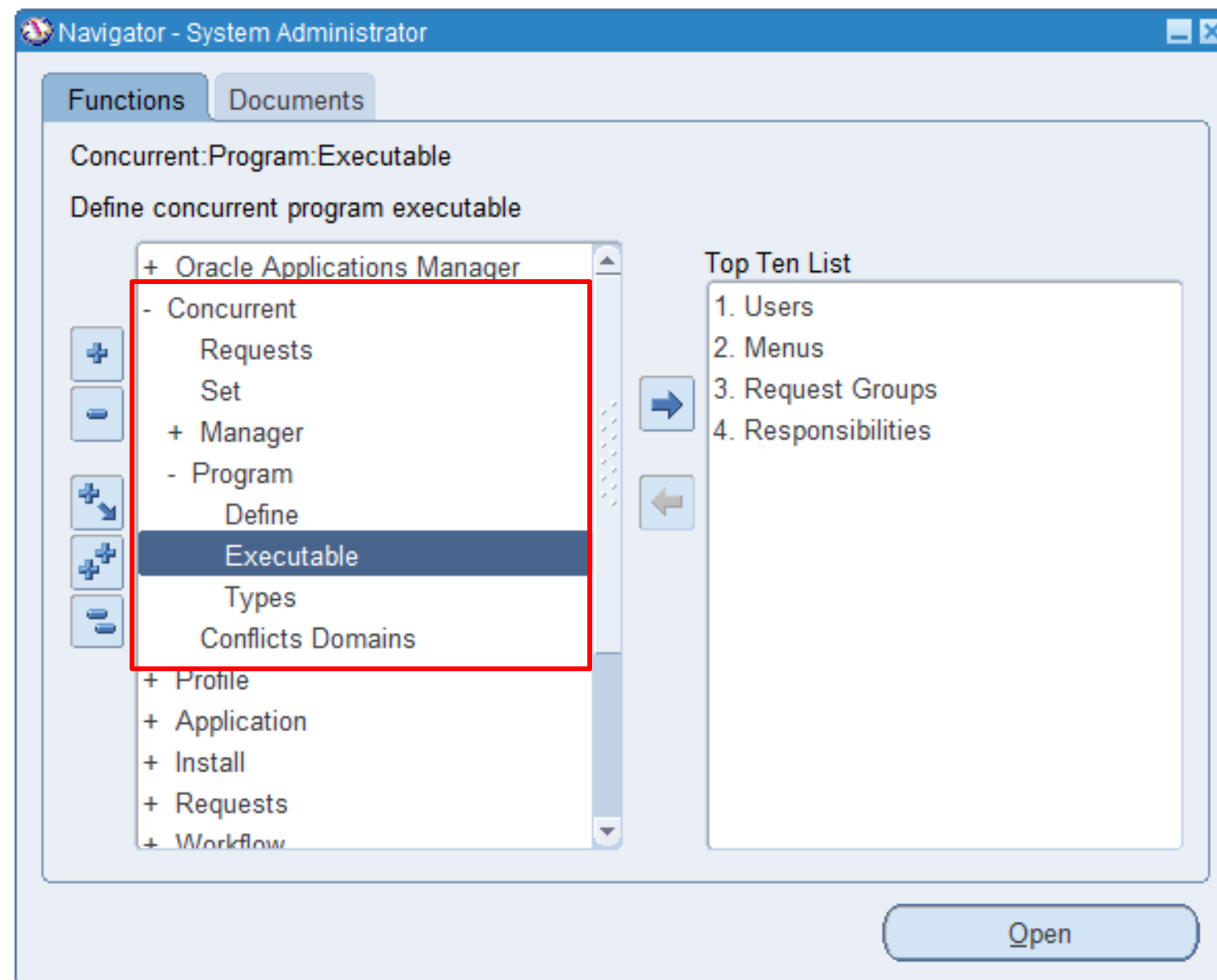
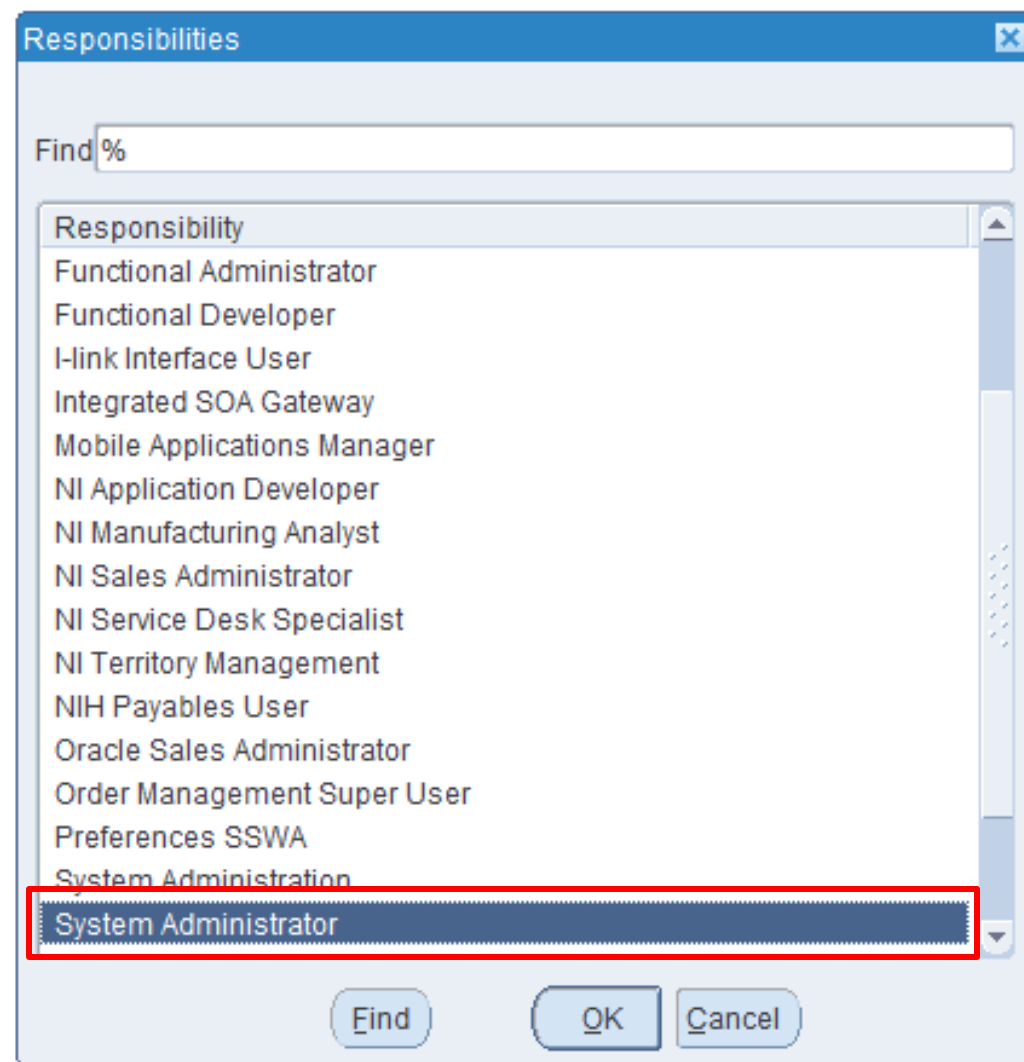
☒ User

☐ Extensible

☐ System

				Effective Dates		Enabled	
Code	Meaning	Description	Tag	From	To		[]
ACCREDITED	Accredited Calibration	Accredited Calibration		25-MAR-2011		<input checked="" type="checkbox"/>	<input type="checkbox"/>
COMPLIANT	Compliant Calibration	Compliant Calibration		25-MAR-2011		<input checked="" type="checkbox"/>	<input type="checkbox"/>
EXCHANGE	Exchange Only	Exchange Only		25-NOV-2015		<input checked="" type="checkbox"/>	<input type="checkbox"/>
EXPEDITED_1	Expedited Traceable C	Expedited Traceable Ca		19-OCT-2015		<input checked="" type="checkbox"/>	<input type="checkbox"/>
FACTORY_CA	Factory Calibration	Factory Calibration		05-NOV-2011		<input checked="" type="checkbox"/>	<input type="checkbox"/>
FACTORY_TE	Factory Test	Factory Test		25-MAR-2011		<input checked="" type="checkbox"/>	<input type="checkbox"/>
REPAIR	Repair	Repair		25-MAR-2011		<input checked="" type="checkbox"/>	<input type="checkbox"/>
TRACEABLE	Traceable Calibration	Traceable Calibration		25-MAR-2011		<input checked="" type="checkbox"/>	<input type="checkbox"/>
						<input type="checkbox"/>	<input type="checkbox"/>
						<input type="checkbox"/>	<input type="checkbox"/>

Executables



Concurrent Programs - Executable

Program Executable

Executable	NI Transfer Price Analysis Report
Short Name	NI_TRANSFER_PRICE_RPT
Application	NI E-Business Tax
Description	
Execution Method	PL/SQL Stored Procedure
Execution File Name	ni_transfer_price_pkg.tp_report
Subroutine Name	
Execution File Path	



Concurrent Programs

Program	NI Tfr Price Analysis Report	<input checked="" type="checkbox"/> Enabled
Short Name	NI_TRANSFER_PRICE_REPORT	
Application	NI E-Business Tax	
Description		

Executable	
Name	NI_TRANSFER_PRICE_RPT
Method	PL/SQL Stored Procedure

Options	
Priority	

Request	
Type	NIJOBS
Incrementor	
MLS Function	

<input checked="" type="checkbox"/> Use in SRS	<input type="checkbox"/> Allow Disabled Values
<input type="checkbox"/> Run Alone	<input checked="" type="checkbox"/> Restart on System Failure
<input type="checkbox"/> Enable Trace	<input checked="" type="checkbox"/> NLS Compliant
<input type="checkbox"/> Recalculate Default Parameters	

Output	
Format	HTML
<input checked="" type="checkbox"/> Save (C)	
<input checked="" type="checkbox"/> Print	
Columns	
Rows	
Style	Compress_and_share_X
<input type="checkbox"/> Style Required	
Printer	NI_FIN_COMPRESS_AI

Business Events	
<input type="checkbox"/> Request Submitted (Y)	<input type="checkbox"/> Request Running
<input type="checkbox"/> Request On Hold	<input type="checkbox"/> Program Completed
<input type="checkbox"/> Request Resumed	<input type="checkbox"/> Post Processing Started
	<input type="checkbox"/> Post Processing Ended
	<input type="checkbox"/> Request Completed (Z)

Copy to... Session Control Incompatibilities Parameters (G)