


Introduction

Prerequisites


- Training accounts in the training database
- Under each account grant: create public synonym, and execute any procedure
- Under each account run the CT.sql script (provided by instructor) to create the training tables

 **SUNGARD** SCT HIGHER EDUCATION www.sungardct.com 4

Introduction

Goals


- Utilize HTML and PL/SQL programming languages in web programming for Self Service Banner.
- Define Oracle 9iAS architecture, security settings, advanced features and development and standards in web programming for Self Service Banner.

 **SUNGARD** SCT HIGHER EDUCATION www.sungardct.com 5

Introduction

Expectations


- What do you expect to get out of this training?

 **SUNGARD** SCT HIGHER EDUCATION www.sungardct.com 6

Introduction

Agenda

- **Day 1**
 - Introduction
 - Oracle 9iAS Architecture
 - HTML Review
 - PL/SQL Review
 - PL/SQL Web Toolkit
- **Day 2**
 - Security
 - Development and Standards
 - Style Sheets




SUNGARD SCT HIGHER EDUCATION www.sungardct.com 7

Introduction

What is Banner Self Service?

- **Add-on to Banner baseline products**
- **Advantages of Banner Web Design**
 - reduced network processing of data servers
 - move from fat client to thin client on desktop
 - no middle tier forms, logic maintained on database server
- **Uses Oracle PL/SQL Packages on Banner database**
 - static HTML pages, CSS and image files on Web Server




SUNGARD SCT HIGHER EDUCATION www.sungardct.com 8

Introduction

Banner Web Documentation

- **Sources**
 - Banner Bookshelf
 - Action Line
 - CD distribution
- **Self Service Implementation Guide**
- **Web Tailor User Guide**
- **Self Service Methodology Handbook**
- **Web for General User Guide**



SUNGARD SCT HIGHER EDUCATION www.sungardct.com 9

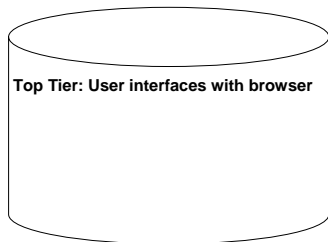
Oracle 9iAS Architecture

- After completing this section, you will be able to:
 - Identify self service banner architecture tiers
 - Describe configuration files and configuration file paths
 - Define web listener, data access descriptor and browser requests



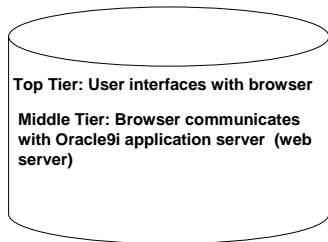
Oracle 9iAS Architecture - Tiers

Self Service Banner Architecture Tiers



Oracle 9iAS Architecture - Tiers

Self Service Banner Architecture Tiers



Oracle 9iAS Architecture - Tiers

Self Service Banner Architecture Tiers

Top Tier: User interfaces with browser

Middle Tier: Browser communicates with Oracle9i application server (web server)

Database Tier: Web requests are processed by stored procedures

SUNGARD SCT HIGHER EDUCATION

www.sungardict.com 13

Oracle 9iAS Architecture - Tiers

Oracle Application Server

- Oracle Application Server puts into place an architecture which will allow you to build applications that tap into the Oracle database, all the while using the deployable and versatile medium of the World Wide Web.
 - Transaction management
 - Multithreaded processing of requests
 - Thin client, server-side processing
 - Platform and browser independent
 - Pre-disposed to growth factors

SUNGARD SCT HIGHER EDUCATION

www.sungardict.com 14

Oracle 9iAS Architecture - Tiers

Oracle 9i Application Server

Oracle9i Application Server

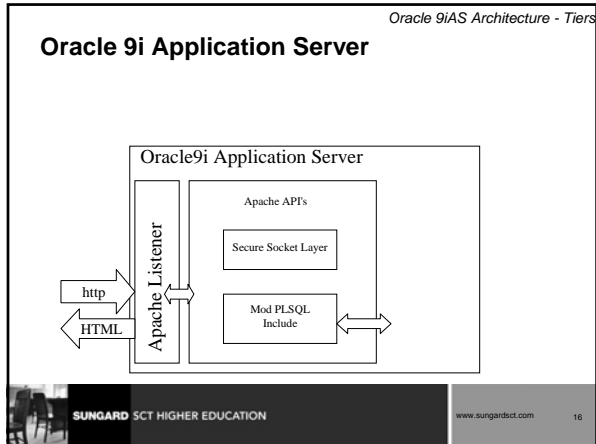
Apache Listener

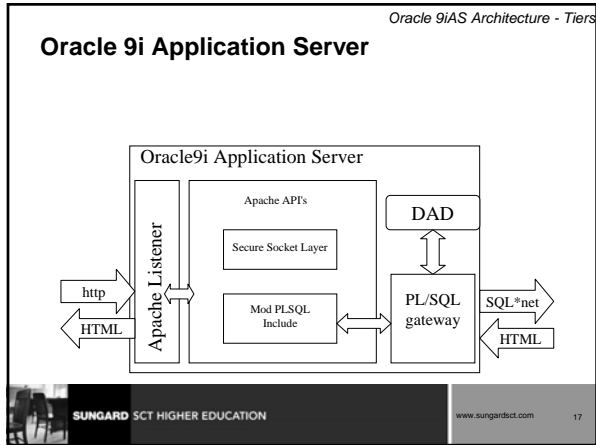
http →

← HTML

SUNGARD SCT HIGHER EDUCATION

www.sungardict.com 15





Oracle 9iAS Architecture - Details

Configuration Files


PATH	DESCRIPTION
\$ORACLE_HOME/Apache/Apache/conf/httpd.conf	Master Apache Listener file
\$ORACLE_HOME/Apache/Apache/conf/oracle_apache.conf	Oracle specific information
\$ORACLE_HOME/Apache/modplsql/cfg/plsql.conf	PLSQL gateway configuration file
\$ORACLE_HOME/Apache/modplsql/cfg/wdbsvr.app	DAD configuration file

www.sungardict.com 18

Oracle 9iAS Architecture - Details

Web Listener

- Responsible for opening a port on the web server to receive, secure, and interpret HTTP requests
- **Software daemon**
 - listens for requests on pre-defined ports (use any higher than 1000)
- **For maintenance, best to have separate web listeners for each database instance**
 - Allows systems to be brought up and down independently
 - Allows SSL to be used for web
- **Defined by one file configuration file on 9iAS**
`$ORACLE_HOME/Apache/Apache/conf/httpd.conf`



SUNGARD SCT HIGHER EDUCATION

www.sungardict.com 19


Oracle 9iAS Architecture - Details

Data Access Descriptor

- Locate the remote database
- Establish a SQL*Net connection
- Login to the remote database as oracle user
- Execute requested package.procedure
- Define in flat file:
`ORACLE_HOME/Apache/modplsql/cfg/wdbsvr.app`
- DADs can be modified, removed and added via web page
http://server.domain:port/pls/admin_/gateway.htm (R1)

or

<http://server.domain:1810> (9iAS R2 & 10gAS)




SUNGARD SCT HIGHER EDUCATION

www.sungardict.com 20

Oracle 9iAS Architecture - Details

Browser Requests

- The Oracle HTTP Server receives a PL/SQL Server Page request from a client browser



SUNGARD SCT HIGHER EDUCATION

www.sungardict.com 21

Browser Requests

- The Oracle HTTP Server receives a PL/SQL Server Page request from a client browser
- The Oracle HTTP Server routes the request to the PL/SQL Gateway



Browser Requests

- The Oracle HTTP Server receives a PL/SQL Server Page request from a client browser
- The Oracle HTTP Server routes the request to the PL/SQL Gateway
- The request is forwarded by the PL/SQL Gateway to the Oracle9i Database. By using the configuration information stored in your DAD, the PL/SQL Gateway connects to the database



Browser Requests

- The Oracle HTTP Server receives a PL/SQL Server Page request from a client browser
- The Oracle HTTP Server routes the request to the PL/SQL Gateway
- The request is forwarded by the PL/SQL Gateway to the Oracle9i Database. By using the configuration information stored in your DAD, the PL/SQL Gateway connects to the database
- The PL/SQL Gateway prepares the call parameters, and invokes the PL/SQL procedure in the application



Browser Requests

- The Oracle HTTP Server receives a PL/SQL Server Page request from a client browser
- The Oracle HTTP Server routes the request to the PL/SQL Gateway
- The request is forwarded by the PL/SQL Gateway to the Oracle9i Database. By using the configuration information stored in your DAD, the PL/SQL Gateway connects to the database
- The PL/SQL Gateway prepares the call parameters, and invokes the PL/SQL procedure in the application
- The PL/SQL procedure generates an HTML page using data and the PL/SQL Web Toolkit accessed from the database



Browser Requests

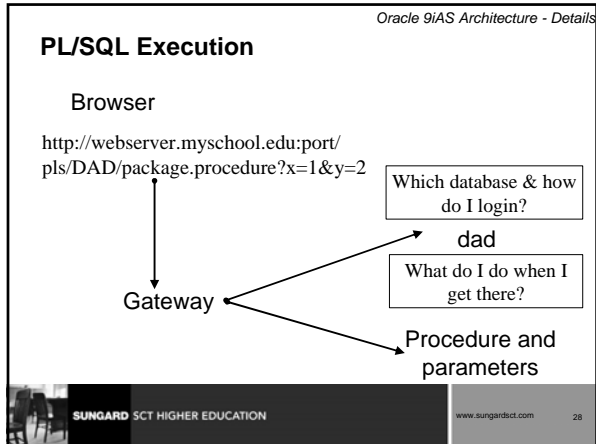
- The Oracle HTTP Server receives a PL/SQL Server Page request from a client browser
- The Oracle HTTP Server routes the request to the PL/SQL Gateway
- The request is forwarded by the PL/SQL Gateway to the Oracle9i Database. By using the configuration information stored in your DAD, the PL/SQL Gateway connects to the database
- The PL/SQL Gateway prepares the call parameters, and invokes the PL/SQL procedure in the application
- The PL/SQL procedure generates an HTML page using data and the PL/SQL Web Toolkit accessed from the database
- The response is returned to the PL/SQL Gateway

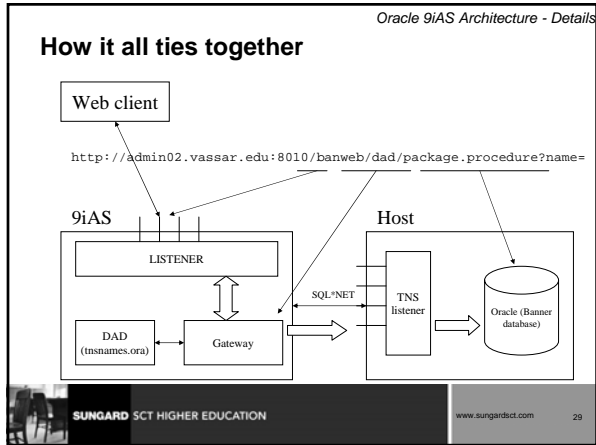


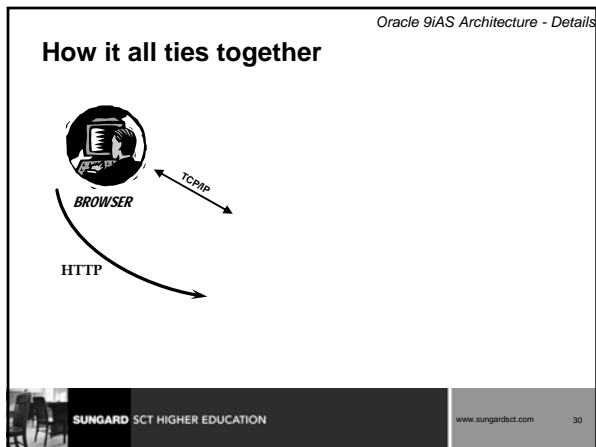
Browser Requests

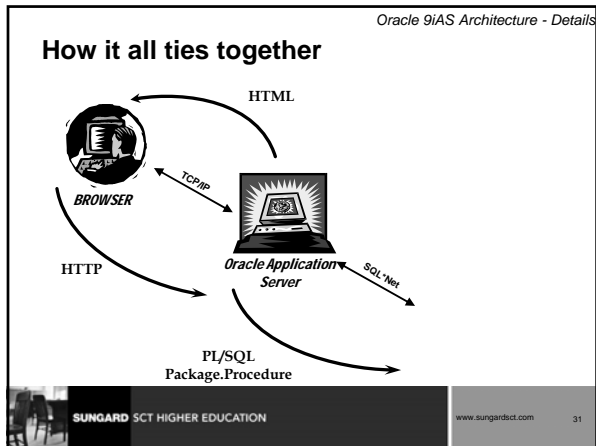
- The Oracle HTTP Server receives a PL/SQL Server Page request from a client browser
- The Oracle HTTP Server routes the request to the PL/SQL Gateway
- The request is forwarded by the PL/SQL Gateway to the Oracle9i Database. By using the configuration information stored in your DAD, the PL/SQL Gateway connects to the database
- The PL/SQL Gateway prepares the call parameters, and invokes the PL/SQL procedure in the application
- The PL/SQL procedure generates an HTML page using data and the PL/SQL Web Toolkit accessed from the database
- The response is returned to the PL/SQL Gateway
- The Oracle HTTP Server sends the response to the client browser

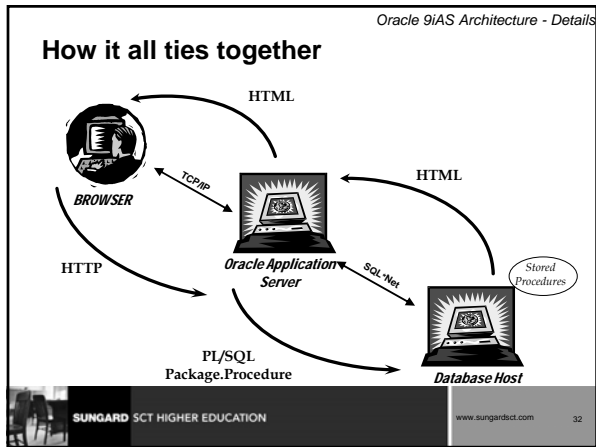












Oracle 9iAS Architecture

Self Check

- What is in the middle tier in Oracle 9iAS?
- What is the DAD configuration file?
- What helps establish a SQL * NET connection?

SUNGARD SCT HIGHER EDUCATION www.sungardst.com 33

HTML Review

- After completing this section, you will be able to create a static HTML page using basic formatting and document tags

HTML Review

Building Blocks

- HTML
- PL/SQL
- Oracle Application Server Toolkit
 - HTP, HTF & OWA%

HTML Review

Hyper Text Markup Language

- Language of the web
- Instructs browser what to do
- Series of tags in an ASCII/text document
- HTML commands are set off by < >
- Tags travel in pairs `<bold> hi </bold>` with what you want to affect in between tags
- `<html>` is a start, and `</html>` is a stop
- You can always 'view source' in browser

HTML Review

HTML Syntax

- Most HTML tags have attributes


```
<FONT color=blue face="Arial" size=6> </FONT>
```

```
<A HREF="homepage.htm"> click here </A>
```

```
<P align=right>
```
- Many tags need to be closed </BODY>
- Some tags need attributes <HREF..>

SUNGARD SCT HIGHER EDUCATION www.sungardct.com 37

HTML Review

HTML Tags

- Basic
- Advanced
 - Tables
 - Forms
 - List tags

SUNGARD SCT HIGHER EDUCATION www.sungardct.com 38

HTML Review

HTML Tables

```
<table> </table> creates a table of data
```

```
<th> </th> specifies the table header
```

```
<td> </td> specifies the table data
```

```
<tr> </tr> specifies table row (where does it end)
```

Example:

```
<table>
```

```
<tr>
```

```
<th ID </th>
```

```
<th Name </th>
```

```
</tr>
```

```
<tr>
```

```
<td> 123-45-6789
```

```
</td>
```

```
<td> Elmer Fudd </td>
```

```
</tr>
```

```
</table>
```

ID	Name
123-45-6789	Elmer Fudd

SUNGARD SCT HIGHER EDUCATION www.sungardct.com 39

HTML Review

Self Check

- Create a web page to mirror the contents you see on-screen.

SUNGARD SCT HIGHER EDUCATION www.sungardstct.com 40

PL/SQL Review

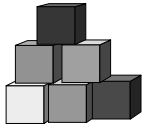
- After completing this section, you will be able to:
 - Define PL/SQL and the PL SQL toolkit
 - Create a package and procedure in the training database
 - Modify a package to create lists and display text
 - Create a form by using a package

SUNGARD SCT HIGHER EDUCATION www.sungardstct.com 41

PL/SQL Review - Overview

Building Blocks

- HTML
- PL/SQL
- Oracle Application Server Toolkit
 - HTP, HTF & OWA%




SUNGARD SCT HIGHER EDUCATION www.sungardstct.com 42

PL/SQL Review - Overview

PL/SQL

- **Procedural Language/Standard Query Language**
- **Basic**
 - Variables
 - Control Structures
 - if-then-else, for-loop, while-loop, exit-when and goto
 - Cursor and cursor FOR loops
- **Modern**
 - Modularity; subprograms and packages
 - Information Hiding and Data Encapsulation
 - Overloading
 - Exception Handling




SUNGARD SCT HIGHER EDUCATION www.sungardict.com 43

PL/SQL Review - Overview

PL/SQL, Continued

- **Block Structured**
 - Logical unit of work (a module)
 - Basic units; named (procedures, functions and packages) and anonymous blocks
 - Supports the idea of *stepwise refinement*
- **Three Parts**
 - Declaration (optional); make variables known
 - Execution
 - Exception (optional) handler for warnings or errors
- **Nest**
 - Sub-blocks allowed in Executable and Exception parts and can define subprograms in Declaration




SUNGARD SCT HIGHER EDUCATION www.sungardict.com 44

PL/SQL Review - Overview

PL/SQL Modularity

- **Break a program down into manageable modules aiding in top-down design**
 - Reusability
 - Maintainability
 - Abstraction




SUNGARD SCT HIGHER EDUCATION www.sungardict.com 45

PL/SQL Review - Overview

PL/SQL Modularity, Continued

- **Subprograms**
 - Procedures
 - takes parameters and performs an action
 - Functions
 - takes parameters and returns computed value
- **Packages**
 - collection of logically common variables, cursors, procedures and functions
 - entire package loaded into memory or can 'Pin' into memory
 - status in Database must be 'VALID' in order to run




SUNGARD SCT HIGHER EDUCATION

www.sungardstct.com 46

PL/SQL Review - Overview

Package Syntax

- **Two Parts to a Package**
 - Package Specification (interface)
 - Also known as package Header
 - Public declarations that are visible to your application
 - Package Body
 - Implementation details (actual code) and private declarations



SUNGARD SCT HIGHER EDUCATION

www.sungardstct.com 47


PL/SQL Review - Overview

Package Specification

```

CREATE OR REPLACE PACKAGE package_name
IS
  <declaration of public variables>
  <declaration of public cursors>
  <declaration of public functions & procedures>
END package_name;

```



SUNGARD SCT HIGHER EDUCATION

www.sungardstct.com 48

Package Body

```

CREATE OR REPLACE PACKAGE BODY package_name
IS
  <declaration of public variables - again!>
  <declaration of public cursors - with SELECT>
  <declaration of public functions & procedures - BODY>
BEGIN
  ...
EXCEPTION
  ...
END package_name;

```



Procedure Syntax

```

PROCEDURE name [ (parameter list) ]
IS
  <declaration section>
BEGIN
  <execution section>
EXCEPTION
  <exception section>
END name;

```



Function Syntax

```

FUNCTION name [ (parameter list) ] RETURN
return_datatype
IS
  <declaration section>
BEGIN
  <execution section>
EXCEPTION
  <exception section>
END name;

```




PL/SQL Review - Overview

Cursors

```

CURSOR name [ (parameter list) ]
IS
SELECT <statement>;

```




SUNGARD SCT HIGHER EDUCATION

www.sungardict.com 52

PL/SQL Review - Overview

Self Check

- Review page 16 of the workbook and complete the exercise.
 - Create a new package and procedure in the training database.
 - Refer to example package syntax on 19 – "Hello World" Example
 - Make Package name unique! (Hello_World_Yourname)



SUNGARD SCT HIGHER EDUCATION

www.sungardict.com 53

PL/SQL Review – PL/SQL Toolkit


PL/SQL Toolkit

- A series of packages and procedures delivered by Oracle which create HTML tags
- Web displays using HTP & HTF packages
- Delivered as part of 9iAS installation and created in the sys schema on database

```

$ORACLE_HOME/Apache/modplsql/owa/owaload.sql

```




SUNGARD SCT HIGHER EDUCATION

www.sungardict.com 54

PL/SQL Review – PL/SQL Toolkit

PL/SQL Toolkit, Continued

- The purpose of the toolkit is to allow you to more easily generate web content from the information contained in the Banner (Oracle) Database
- The packages provide procedures and functions to be used by the Oracle Application Server (OAS) and Self Service Banner




SUNGARD SCT HIGHER EDUCATION www.sungardct.com 55

PL/SQL Review – PL/SQL Toolkit

PL/SQL Toolkit, Continued

- Oracle tools for creating HTML inside of PL/SQL
- Installed into each database and owned by sys schema
- Comprised of Packages;
 - HTP - HyperText Procedures
 - HTF - HyperText Functions
 - OWA - Oracle Web Agent Packages
 - OWA, OWA_SEC, OWA_TEXT, OWA_COOKIE, OWA_UTIL(called internally by the pl/sql cartridge)
- Uses baseline Oracle packages
 - DBMS_OUTPUT, UTL_RAW, ...

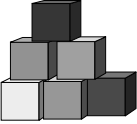



SUNGARD SCT HIGHER EDUCATION www.sungardct.com 56

PL/SQL Review – PL/SQL Toolkit

Building Blocks

- HTML
- PL/SQL
- PL/SQL Toolkit
 - HTP, HTF & OWA%





SUNGARD SCT HIGHER EDUCATION www.sungardct.com 57

PL/SQL Review – PL/SQL Toolkit

HTP & HTF Packages

- **HTP**
 - Package containing one procedure for every HTML tag available
 - These generate HTML
- **HTF**
 - Package containing one function for every HTML tag available
 - These parse HTML
 - The HTF package turns HTP procedures into functions that return the HTML output as a formatted string




SUNGARD SCT HIGHER EDUCATION www.sungardst.com 58

PL/SQL Review – PL/SQL Toolkit

Using HTP and HTF

- Opening and closing tags
- PL/SQL will not catch your HTML syntax errors
- HTML tag attributes will match the parameter list




SUNGARD SCT HIGHER EDUCATION www.sungardst.com 59

PL/SQL Review – PL/SQL Toolkit

OWA Packages

- **OWA_COOKIE**: sub-programs which send and retrieve HTTP cookies from browsers
- **OWA_IMAGE**: sub-programs which retrieve coordinates where a user clicks on an image
- **OWA_SEC**: sub-programs which define authentication function, and retrieve password, username and host information for user authentication
- **OWA_PATTERN, OWA_TEXT**: subprograms which manipulate string data
- **OWA_UTIL**: subprograms used to display and retrieve sets of data, handle CGI environment variables, and run dynamic SQL to improve productivity



SUNGARD SCT HIGHER EDUCATION www.sungardst.com 60

OWA Packages, Continued

- **OWA_UTIL - utility procedures**
 - showsource
print out the source of a PL/SQL stored object
 - showpage
dumps the HTML buffer to a SQL*Plus or Server Manager screen
 - print_cgi_env
prints out all of the HTTP environment variables for the session
 - redirect_url
redirects output to another URL
 - tableprint
prints out an Oracle table



PL/SQL Documentation

- **Oracle 9iAS R1 documentation -- direct your browser to**
http://download-west.oracle.com/docs/cd/A97335_01/index.htm
- **Oracle 10g PL/SQL Reference Guide**
 - **HTP:**
http://download.east.oracle.com/docs/cd/B19306_01/appdev.102/b14258/w_htp.htm#1058734
 - **HTF:**
http://download-east.oracle.com/docs/cd/B19306_01/appdev.102/b14258/w_hft.htm#1011713
- **Login, click on "run web sites and apps", scroll to Oracle PL/SQL and click on the "Using PL/SQL Gateway" link or "Web Toolkit Reference Guide"**



HTML attributes in PL/SQL

- **Passed as parameter values to the HTP or HTF procedure**
- **If the HTML should look like...**

```
<IMG SRC="/images/logo.gif" ALIGN="CENTER" ALT="Logo">
```



HTML attributes in PL/SQL, Continued

- Passed as parameter values to the HTP or HTF procedure

- If the HTML should look like...

```
<IMG SRC="/images/logo.gif" ALIGN="CENTER" ALT="Logo">
```

- Then the PL/SQL will look like:

```
HTP.IMG('/images/logo.gif','CENTER','LOGO');
```



Start with a Template

- Structure Tags

- HTP.htmlOpen creates <html>
- HTP.headOpen creates <head>
- HTP.headClose creates </head>
- HTP.BodyOpen('background', 'attributes') creates <body background='graphic file' attributes>
- HTP.BodyClose creates </body>
- HTP.htmlClose creates </html>



Start with a Template, Continued

```
CREATE OR REPLACE PACKAGE dummy
IS
  Procedure
END ;
/
CREATE OR REPLACE PACKAGE BODY
IS
  Procedure do_something IS
    BEGIN
  END ;
END ;
/
```



Create the Main Document Tags

```

BEGIN

    HTP.htmlOpen;
    HTP.headOpen;
        /* header options */
    HTP.headClose;
    HTP.bodyOpen('mybackground','BASEFONT=');

    /*most other work done here */

    HTP.bodyClose;
    HTP.htmlClose;

END ;

```

Use PL/SQL Toolkit to Create HTML

```

BEGIN

    HTP.htmlOpen;
    htp.headOpen;
    htp.title('Hello World title');
    htp.comment('This is my html heading section. ');
    HTP.headClose;
    case
    insensitive
        HTP.bodyOpen(null,'text="blue"');
        htp.p('Yeah, you guessed it... Hello World. ');

    HTP.bodyClose;
    HTP.htmlClose;

END ;

```

Question: Why did I put NULL in BodyOpen?

Use PL/SQL Toolkit to Create HTML, Continued

Would look like this;

```

<html>
<head>
<title>Hello World title </title>
<!-- This is my html heading section. -->
</head>
<body text="blue">
Yeah, you guessed it... Hello World.
</body>
</html>


```

Question: Where are the blank lines from the previous slide?

PL/SQL Review – Tags

Common Tags

- Head tags
- Body tags
- Formatting tags
- List tags



www.sungardst.com 70


PL/SQL Review – Tags

List Example HTML Output

```

.....
...
<ul>
<lh> Registration Checklist </lh>
<li> Biology 101
<li> English Refresher
<li> Physical Education
</ul>
.....

```




www.sungardst.com 71

PL/SQL Review – Tables

Tables

- These toolkit package.procedure calls are used to create and populate an HTML Table:

HTML	HTP
<TABLE>	HTP.tableOpen()
</TABLE>	HTP.tableClose()
<TR>	HTP.tableRowOpen()
</TR>	HTP.tableRowClose()
<TH>	HTP.tableheader()
<TD>	Htp.tableData()



www.sungardst.com 72

Table Example 1: HTML Output

```

<table>
<tr>
<th> Student ID </th>
<th> Grade in Class </th>
</tr>
<tr>
<td> 382482771 </td>
<td> B+ </td>
</tr>
<tr>
<td> 983278820 </td>
<td> C- </td>
</tr>
</table>

```

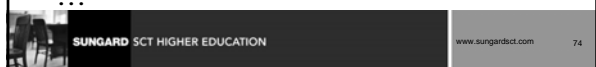


Table Example 2: HTML Output

```

...
<table "border=1" align="center" BGCOLOR=gray>
<tr ALIGN="left" BGCOLOR=yellow>
<th> Account </th>
<th> Balance </th>
</tr>
<tr>
<td> <I> 302076 </I> </td>
<td> <I> 8,204.78 </I> </td>
</tr>
</table>
...

```



Self Check

- Follow the instructions on pages 24 and 25.
 - The "Hello_world_your_name" package will be used to display "Hello World Name" to a browser.
 - Your existing package will be used to produce a list of street addresses.
 - The instructor will demonstrate how to register a page in Web Tailor after you complete Step 7 of the exercise.



Forms

- Variables and values
- Form tags
- Form procedures



Life cycle of a variable/value pair

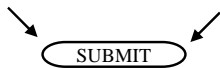
- Variable instantiated at HTML
- Value assigned at POST
- URL of the ACTION receives the pair, manipulates and/or forwards to the next cycle



Life cycle of a variable/value pair, Continued

HTML

```
FORM ACTION="url" METHOD="POST"
<INPUT TYPE="text" NAME="term">
```



PL/SQL

If a student can register for Fall 2005, then display registration and maintain the value Fall 2005 throughout the student's session

” However, only display courses which are offered for Fall 2005.

All of this info is lost after the HTML response is generated.



PL/SQL Review – Forms

Display Form

- HTML output looks like...

```

...
<FORM ACTION="mypackage.P_ProcessTerm",
METHOD=POST>
<TABLE>
<TR>
<TD><B>Select a term:</B></TD>
<TD><INPUT TYPE="Text" NAME=term SIZE=6>
</TD>
</TR>
</TABLE>
<INPUT TYPE="Submit" NAME="Submit Term">
</FORM>
...

```

SUNGARD SCT HIGHER EDUCATION www.sungardct.com 79

PL/SQL Review – Forms

Variables and Values

- ACTION URL now becomes a procedure**
 - package.procedure notation
- Declarations at HTML are posted to the procedure, so the procedure has to be prepared to receive them**
- Variables declared at HTML establish the name of the variable in PL/SQL parameter list**
 - PL/SQL establishes the datatype

SUNGARD SCT HIGHER EDUCATION www.sungardct.com 80

PL/SQL Review – Forms

Variables and Values

- Constant feedback between HTML and PL/SQL**
 - HTML has no 'memory' area
 - Value pairs that are essential to processing have to be 'hidden' on each subsequent page

SUNGARD SCT HIGHER EDUCATION www.sungardct.com 81

Retain the Values

```

...
Procedure P_DisplaySchedule(term in varchar2) is
begin
...
    http.formOpen('mypackage.P_ProcessSchd',
        'POST');
/* set up hidden variables */
http.formHidden('term',term);
/* display schedule options */
http.formSubmit(null,'Submit Schedule Options');
http.formClose;
...

```

Toolkit Procedures

- http.FormOpen
- http.FormClose
- http.FormSubmit
- http.FormImage
- http.FormReset
- http.FormHidden
- http.FormText
- http.FormRadio

Toolkit Procedures, Continued

- http.FormSelectOpen
- twbkwbis.p_FormSelectOption
- http.FormTextArea
- http.FormCheckBox
- http.FormPassword

Forms Example

```

HTP.formOpen ('bwzkwpay.p_calc_refund');

HTP.formHidden('pidm','482911');

HTP.formSelectOpen('refund_class', 'Pick class to request a refund:');
HTP.formSelectOption('English 410');
HTP.formSelectOption('Biology 305');
HTP.formSelectOption('Physics 400');
HTP.formSelectOption('Mathematics 320');
HTP.formSelectClose;

HTP.formSubmit (NULL, ' Press here ');
HTP.formClose;

```



Forms Example generates this

```

<form action="bwzkwpay.p_calc_refund" method="POST">

<input type="hidden" name="pidm" value="482911">

Pick class to request a refund:
<select name="refund_class">
  <Option> English 410
  <Option> Biology 305
  <Option> Physics 400
  <Option> Mathematics 320
</select>

<input type="submit" value=" Press here ">
</form>

```



Formatting Inputs


- Inputs are encapsulated in a HTML form
- Each form input is put into a HTML table data (cell) i.e. Each form input is put into a HTML table data cell
- A label may be placed to the left of the cell



PL/SQL Review – Forms

Self Check

- Review the instructions on page 43.
 - Process and display the value input by the user
 - Display the value input -- refer to 78 & 79 for example code




SUNGARD SCT HIGHER EDUCATION

www.sungardct.com 88

Security

- After completing this lesson, you will be able to:
 - Add a security call to a procedure
 - Describe the data in security tables




SUNGARD SCT HIGHER EDUCATION

www.sungardct.com 89

Security

Security Overview

- **Firewalls**
 - Prevent unauthorized network access and attacks by protecting the points of network entry
- **SSL - Secured Socket Layer**
 - Encrypts packets between client and server
 - Industry leading security for Web servers
 - Requires VERISIGN certificate
- **ANO - Advanced Networking Option**
 - Oracle Add-on to encrypt SQL*Net communication
 - Formerly known as ANO – advanced network option



SUNGARD SCT HIGHER EDUCATION

www.sungardct.com 90

Security

Security Vulnerability

The diagram illustrates a security vulnerability where hackers attempt to intercept HTTP/SSL traffic between the Internet and a Firewall. The Firewall successfully blocks this attempt, indicated by a large 'X' over the hacker's path.

SUNGARD SCT HIGHER EDUCATION www.sungardict.com 91

Security

Security Vulnerability

The diagram illustrates a security vulnerability where sniffers attempt to intercept HTTP/SSL traffic between a Firewall and a 9iAS server. The sniffers' attempt is blocked, indicated by a large 'X' over their path.

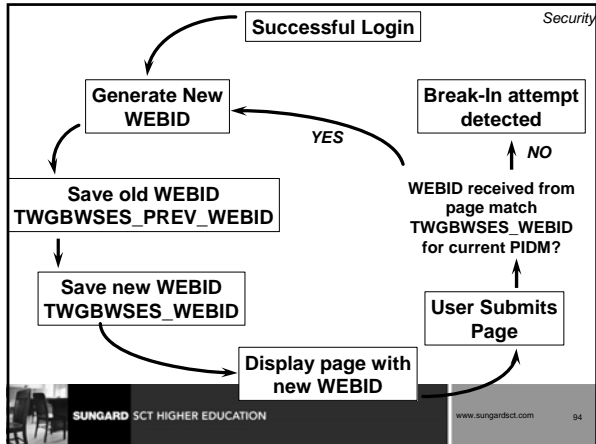
SUNGARD SCT HIGHER EDUCATION www.sungardict.com 92

Security

Security Vulnerability

The diagram illustrates a security vulnerability where sniffers attempt to intercept traffic between a 9iAS server and a Host. The sniffers' attempt is blocked, indicated by a large 'X' over their path. The traffic between the 9iAS server and the Host is labeled as SQL*Net/ANO.

SUNGARD SCT HIGHER EDUCATION www.sungardict.com 93



Security

Security Tables

- GOBPAC – stores PIN
- GOVROLE – role view, determines what menus are available
- TWGRROLE – web tailor roles table
- TWGBWSES – web tailor web ID table, stores ID for user session

95

Security

Security Calls


- Display data to end users
- Call Banner security procedure `twbkwbs.F_ValidUser(pidm number)`
- Add procedure call to beginning of code

96

Security

Self Check

- Follow the instructions on page 46.
 - Add a security call to the P_DisplayHello procedure




SUNGARD SCT HIGHER EDUCATION

www.sungardst.com 97

Development and Standards

- After completing this lesson, you will be able to:
 - Incorporate an existing procedure into Web tailor
 - Display the content of a course validation table in a HTML table
 - Create a form to display student grades for a term
 - Display course number and GPA values for a term




SUNGARD SCT HIGHER EDUCATION

www.sungardst.com 98

Devt & Stds – Web Tailor Integration

Web Tailor Integration

- Add Banner open document code, information text code and close document code
- Compile your package body
- Create links to access the package in Menu
- Register the packages with Web Tailor
- Login into Self Service products




SUNGARD SCT HIGHER EDUCATION

www.sungardst.com 99

Dev & Stds – Web Tailor Integration

Banner Code to Open a Document

- A procedure call creates the header of your HTML document
- Web Tailor allows you to define items that display at the top of document:
 - A header image
 - Top links
 - Page title
 - A page header and header links
- `twbkwbis.p_opendoc('procedure defined in web tailor')`
- Creates the `<HTML>` `<HEAD>` `<BODY>` tags




www.sungardct.com 100

Dev & Stds – Web Tailor Integration

Banner Code to Close a Document

- A procedure call creates the footer of your HTML document
- Displays:
 - Return Link
 - Powered by SCT logo
 - Release Version
- Creates `</BODY>` `</HTML>` tags
- `twbkwbis.p_closedoc('version')`




www.sungardct.com 101

Dev & Stds – Web Tailor Integration

Banner Code to Include Information Text

- A procedure call creates the information text associated with a procedure in Web Tailor
- Displays information text defined in TWGRINFO table
- `twbkwbis.p_dispinfo('top level procedure','label name of text – DEFAULT is chosen if blank')`




www.sungardct.com 102

Dev & Stds – Web Tailor Integration

Adding a Procedure to Web Tailor

- Compile package




SUNGARD SCT HIGHER EDUCATION

www.sungardct.com 103

Dev & Stds – Web Tailor Integration

Adding a Procedure to Web Tailor

- Create package
- Register new procedure (package.procedure)




SUNGARD SCT HIGHER EDUCATION

www.sungardct.com 104

Dev & Stds – Web Tailor Integration

Adding a Procedure to Web Tailor

- Create package
- Create new procedure
- Add a new menu item for each procedure to an existing Menu page




SUNGARD SCT HIGHER EDUCATION

www.sungardct.com 105

Dev & Stds – Web Tailor Integration

Adding a Procedure to Web Tailor

- Create package
- Create new procedure
- Add a new menu item for each procedure
- Insert help link or create HELP info text




www.sungardst.com 106

Dev & Stds – Web Tailor Integration

Adding a Procedure to Web Tailor

- Create package
- Create new procedure
- Add a new menu item for each procedure
- Insert help link or create HELP info text
- Create information text -- using the default label




www.sungardst.com 107

Dev & Stds – Web Tailor Integration

Self Check

- Follow the instructions on page 53 of the workbook.
 - Incorporate the hello_world procedure into Web Tailor by adding procedure calls to P_DisplayHello




www.sungardst.com 108

Dev & Stds – Tables, Text and Links

Formatting Procedures

- **Banner formatting procedures are used instead of Oracle htp and htf procedures**
- **Advantages of Banner formatting procedures**
 - Save development time
 - Create a uniform look and feel
 - Simplify code
 - Only core application display needs to be developed



SUNGARD SGT HIGHER EDUCATION


www.sungardsgt.com 109

Dev & Stds – Tables, Text and Links

Banner Standards for Tables

- **Banner HTML tables have three standard formats:**
 - **Data Display:** Format to display uniform data
 - `twbkffmt.P(F)_TableOpen('DATADISPLAY');`
 - **Data Entry:** Format to display input for forms
 - `twbkffmt.P(F)_TableOpen('DATAENTRY');`
 - **Plain Display:** Format to display items, links or images
 - `twbkffmt.P(F)_TableOpen('PLAIN');`

UI Methodology and Standards.pdf




SUNGARD SGT HIGHER EDUCATION

www.sungardsgt.com 110

Dev & Stds – Tables, Text and Links

Using Tables to Display Data

- **Before you commence developing your first table, please reference the methodology handbook for a full list of table procedures.**
- **Some examples are:**
 - `P_TableOpen`
 - `P_TableRowOpen`
 - `P_Tabledataheader`
 - `P_Tabledatalabel`



SUNGARD SGT HIGHER EDUCATION

www.sungardsgt.com 111

How do I link to another page?

- Any Page that displays in the browser url box needs to be registered with Web Tailor
- Use this Banner procedure call syntax:
 - `twbkfrmt.P_PrintAnchor('package.procedure', ctext => 'link text');`
- The TWGBRUL table is used to find the /pls/dad parameter
 - The following code allows you to connect to another procedure
 - `twbklbns.twgbwrul_rec.twgbwrul_cgibin_dir`
 - `twbkfrmt.p_printanchor(twbklbns.twgbwrul_rec.twgbwrul_cgibin_dir||'/hello_world_peter.p_displaycourse', ctext => 'go to my hello world');`
 - `twbkfrmt.P_PrintAnchor('Hello World_Peter.DisplayHello', ctext => 'Click to Peter's hello world page');`



Self Check

- Follow the instructions on page 60 in the workbook
 - Create a procedure to display the content of the course validation table, SVWCRSE in an HTML table
 - Display the course number, course description, credit hours and activity date
 - Create a link in your Hello World_P_DisplayHello to your SVWCRSE (ssb_yourname.p_displaycourse) procedure



Data Entry - Basic Form Structure

```

-- Procedure to paint form..
http.formOpen(twbkwbis.f_cgibin||'/hello_chris.Process','post');
http.formtext('param1','25');
http.formtext('param2','25');
http.formSubmit;
http.formClose;

-- Procedure to process forms parameters
create or replace procedure hello_chris (
param1 IN varchar2,
param2 IN varchar2 )
begin ..

```



Banner Form Rules

- **Each Banner Form needs two procedures**
 - Procedure to paint HTML form
 - Procedure to accept input parameters
- **Use Banner Data Entry Table to display input types**
- **Use Oracle htp and htf packages for all form tags except for:**
 - Form text fields
 - Entry required indicator
- **Match HTML Name to Parameter**

 SUNGARD SCT HIGHER EDUCATION www.sungardct.com 115

Table Data Entry, Part 1

- **Example: you want to create an application which allows a student to change his/her address**
- **The student should see a drop-down listing of all address types they have records for...**
 - Perhaps a business address (BA) and a personal address (PA)
- **By choosing an address type from a drop-down and clicking on "submit," the correct address will display**
- **You can do this with a HTML form and a table**

 SUNGARD SCT HIGHER EDUCATION www.sungardct.com 116

Table Data Entry, Part 2

- **Create a cursor to select from spraddr table**

```

..
CREATE OR REPLACE PACKAGE BODY Hello_Chris
IS
Procedure P_Address_Types IS
    pidm number;
    cursor address_cursor(pidm_v number) is
        select * from saturn.spraddr where
            spraddr_pidm = pidm_v;
BEGIN
...

```

 SUNGARD SCT HIGHER EDUCATION www.sungardct.com 117

Table Data Entry, Part 3

- Draw a basic plan of the columns, rows and input types you want on your form
- In this case, you are just displaying a single forms selection box in one table row and one table column
- Hint: sometimes it useful to find an existing form that looks like the one you want to create and use the code as a template
- Let's see the code...



Table Data Entry, Part 3

- Procedure to paint form

```

http.formOpen(twbkwhis.f_ogibin[1]/display_address_your_name.P_process','post');
twbkfmr.P_TableOpen('DATAENTRY');
twbkfmr.P_TableRowOpen;
twbkfmr.P_TableDataLabel('Your Address Types', 'left');
twbkfmr.P_TableDataOpen;
http.formSelectOpen('atype', 'pick the address type');
for address_rec in address_cursor(pidm) loop
  if address_rec.spraddr_atyp_code = 'PA' then
    http.formSelectOption(address_rec.spraddr_atyp_code,
      null, 'SELECTED');
  else
    http.formSelectOption(address_rec.spraddr_atyp_code);
  end if;
end loop;

```



Table Data Entry, Part 4

- Procedure to process form and display data

```

create or replace procedure P_process (atype IN varchar2 )
pidm number;
cursor spraddr_cursor(pidm_v number, atyp_v varchar2) is
select * from saturn.spraddr where
spraddr_pidm = pidm_v
and spraddr_atyp_code = atyp_v;
BEGIN -- security calls to collect PIDM value
for address_rec in spraddr_cursor(pidm, atype) loop
twbkfmr.P_TableOpen('DATADISPLAY', ccaption => 'Your Address');
twbkfmr.P_TableRowOpen;
twbkfmr.P_TableLabel('Address', 'Left');
twbkfmr.P_TableDataOpen;
twbkfmr.P_PrintText(address_rec.SPRADDR_STREET_LINE1);
twbkfmr.P_TableDataClose;
twbkfmr.P_TableRowClose;
twbkfmr.P_TableClose;
end loop; END; /

```



Storing Parameters

- Use the TWGRWPRM table
- Insert a parameter – store the value in the twgrwprm table
- Fetch a parameter – return the value for pidm and parameter name



Set a Parameter

```

pidm number;
last_name_v spriden.spriden_last_name%type;
BEGIN
  if not twgbwis.F_ValidUser(pidm) then
    return;
  end if;
  --
  twgbwis.F_OpenDoc('Hello_Chris.P_DisplayHello');

  -- insert a parameter:
  Select spriden_last into last_name_v where spriden_pidm = pidm and spriden_change_ind is null;
  Twgbwis.F_SetParam(pidm, 'Last_Name', last_name_v);
  -- print Last name from TWGRWPRM
  Htp.p(twgbwis.F_GetParam(pidm, 'Last_Name'));
  ...

```



Self Check

- Follow the instructions on page 69 of the workbook.
 - Create a form to display a student's grades for a term.
 - After a student selects a valid term from a drop down and clicks submit, a page displays each course number and corresponding GPA for that term.



Advanced Topics

- After completing this lesson, you will be able to:
 - Process form checkbox entries using PL/SQL tables
 - Manage the appearance of web pages with styles and style sheets



PL/SQL Tables

- Some forms' input types allow multiple choices
- PL/SQL tables are used to process more than one choice



California



New York



Indiana



PL/SQL Tables, Continued

- PL/SQL tables have two columns, the key and the value
 - Key: type binary integer
 - Value: user defined, for SSB varchar2(1000)
- To reference an individual element
 - tablename(index)



PL/SQL Table syntax

```

DECLARE
  TYPE varchar2_tabtype IS TABLE OF VARCHAR2 (100)
    INDEX BY BINARY_INTEGER;
  v_emails varchar2_tabtype;
BEGIN
  v_emails(1) := 'johnsmith@yahoo.com';
  v_emails(2) := 'fredsmith@yahoo.com';
  v_emails(-4) := 'fredsmith@yahoo.com';
END;

```

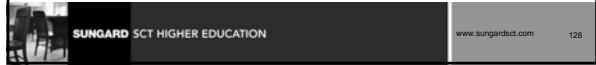


Multiple Choices

```

<HTML>
<HEAD><TITLE>Form Example</TITLE> </HEAD>
<BODY>
<H2> <font color="blue"> States you have visited </font>
</BR>
<FORM Method=Post ENCTYPE="text/plain"
  Action="/pls/trng/process_states">
<input type="checkbox" name="State" value="California">
  California
<input type="checkbox" name="State" value="New York">   New
  York
<input type="checkbox" name="State" value="Indiana">
  Indiana </h2>
</BODY> </HTML>

```



Processing Multiple Choices

```

Procedure process_states
( states in twbkslib.nvarchar2_tabtype);

  I number;
begin
for I in 0 to 10 loop
  if states(I) is not null then
    htp.p('You visited ');
    htp.p(states(I));
    htp.p('<BR>');
  end if;
end loop;
exception
when no_data_found then
  null;
end process_states;

```



Styles

- Styles can be defined externally, at the document level, or a tag level. (SSB uses externally defined styles)
- You can apply a uniform style to all instances of the one tag in a document.
 - All <P> (paragraph) tags could be left justified
- (OR) You can apply a style to a subset of tags
 - Every other <P> (paragraph) could be right justified



Style syntax

selector {property: value}

body {color: blue}

- Selector: html/element tag you want to control
- Property: attribute for the html tag
- Value: value for the property

h1,h2,h3 {font-family:"sans serif ";color:red}

- Use commas to group selectors
- Use quotes for multiple word values
- Use semi-colons for more than one property/value style



Default Style Sheets

- By default, SSB uses four external style sheets to control the look and feel of content:
 - web_defaultapp: all application pages
 - web_defaultmenu: all menu pages
 - web_defaulthome: dynamic homepage displayed before user logins
 - web_defaulthelp: all help files accessed by link at top right hand corner of each page
- These style sheets have been designed to go together
- SCT recommends that your organization use them



SCT-Delivered Style Sheets

- **web_defer.css**
 - This CSS contains the settings provided in the previous releases of the SCT Banner Web products
 - You may want to use this CSS if you have Web for Students or another product that has not yet been converted to the new UI to keep the look and feel of your products consistent
 - FAQ# 6012
 - Defect# 78274
- **web_color.css**
 - This CSS demonstrates the flexibility of the CSS technology
 - It is to be used as an example, and is not recommended for normal use on your institution's web site



CSS Levels

- **Using Web Tailor, you can specify a style sheet file location at the following levels:**
 - System level – updated via **Customize Global User Interface Settings**
 - Module level – updated via **Customize a Web Module**
 - Web page level – updated via **Customize a Web Menu or Procedure**
- **These file locations are expressed as URLs; for example, /css/**



Styles in SSB

- **The external sheet used by a web page is defined by Web Tailor parameter associated with the procedure**
- **You do not need to call a specific style sheet in your code**
- **Default style sheets are called if no style sheet is explicitly associated with a procedure in Web Tailor**

```
<LINK REL="stylesheet" HREF="/css/web_defaulthome.css"
TYPE="text/css">
<LINK REL="stylesheet" HREF="/css/web_defaultprint.css"
TYPE="text/css" media="print">
```



Default Style Sheets

- A class may differ based on the state of the HTML object. For example, the submenulinktext2 class is used for links (<A> tags)
 - If a user has clicked on the link then the color changes:

```
A.submenulinktext2 {
color: #1E2B83;
text-decoration: none;}
A.submenulinktext2:visited {
color: #800080;
text-decoration: none;}
```



Calling Classes in Code

- The following TWBKFRMT procedures have a "class_in" parameter which allow you to specify a new style class for a tag:
 - P_PrintStrong, P_PrintMenu, P_PrintText
 - P_PrintText prints text with global font setting (default size = 3)
- The rest of the formatting procedures call a style sheet internally. Therefore, you can only change the style properties by modifying the existing style
 - If browser is Netscape or Internet Explorer, the css_enable indicator is set to TRUE
 - Enable Indicator determines whether to use the style tag for formatting



Package Initialization for Twbkfrmt


```
begin
/* Try to determine client browser type. If it looks to be a 4.x or 5.x
version of MSIE or Netscape Navigator, set css_enabled to TRUE. */
begin
http_user_agent := owa_util.get_cgi_env('HTTP_USER_AGENT');
exception
when others then
http_user_agent := 'SQLPLUS';
end;
IF http_user_agent IS NOT NULL THEN
IF (instr(upper(http_user_agent),'MSIE') > 0) THEN
IF (instr(http_user_agent,'MSIE 4.') > 0) OR
(instr(http_user_agent,'MSIE 5.') > 0)
THEN css_enabled := TRUE;
END IF;
..
```



Advanced Topics – Styles

Style Sheet Demonstration

- The instructor will demonstrate opening a style sheet in WordPad




SUNGARD SCT HIGHER EDUCATION

www.sungardstct.com 139

Conclusion

- Could you explain the 9iAS Architecture?
- What is the PL/SQL Toolkit?
- Can you integrate your custom code into Web Tailor?
- What are some advanced features you would find useful?
- Where can I find documentation?



SUNGARD SCT HIGHER EDUCATION

www.sungardstct.com 140

Questions and Answers



SUNGARD SCT HIGHER EDUCATION

www.sungardstct.com

