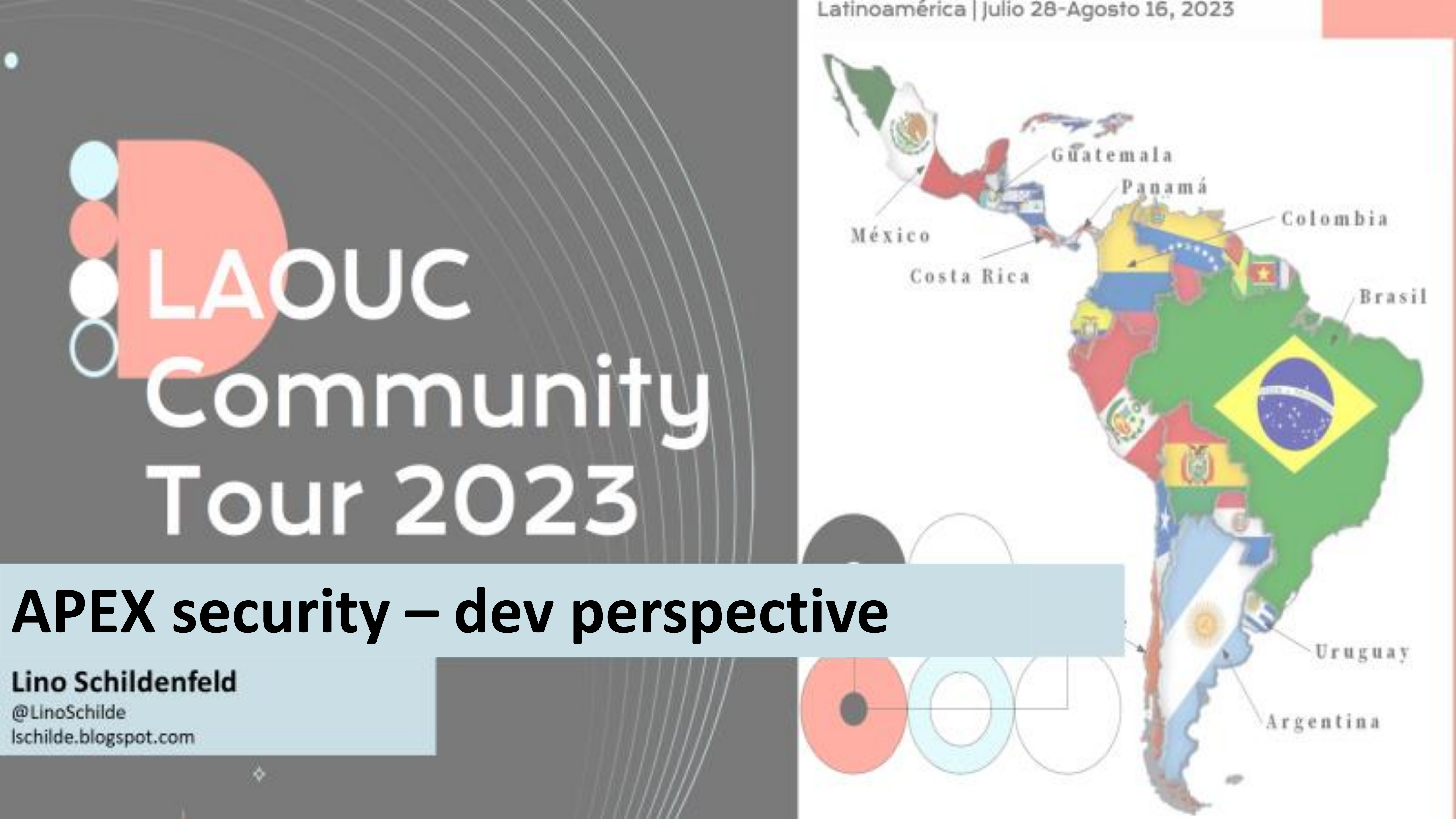


APEX security – dev perspective

Lino Schildenfeld
@LinoSchilde
lschilde.blogspot.com

AUSOUG APEX News

https://www.ausoug.org.au/news/apex-special-series-2023/

MASH PROGRAM

Mentor and Speaker Hub

- Our goal is to *connect* speakers with mentors to assist in *preparing* technical sessions and *improving* presentation skills

Interested? Read more and get in touch

https://mashprogram.wordpress.com

My story

MASH

SKILLBUILDERS

_SYM42

AUSOUG fabe reduce • rescue • reconnect FOEX AOP APEX R&D research & development

- AUSOUG APEX webinars
- NZ APEX meetup organizer

- APEX World Member of the Month
- Conference speaker

- Blogger

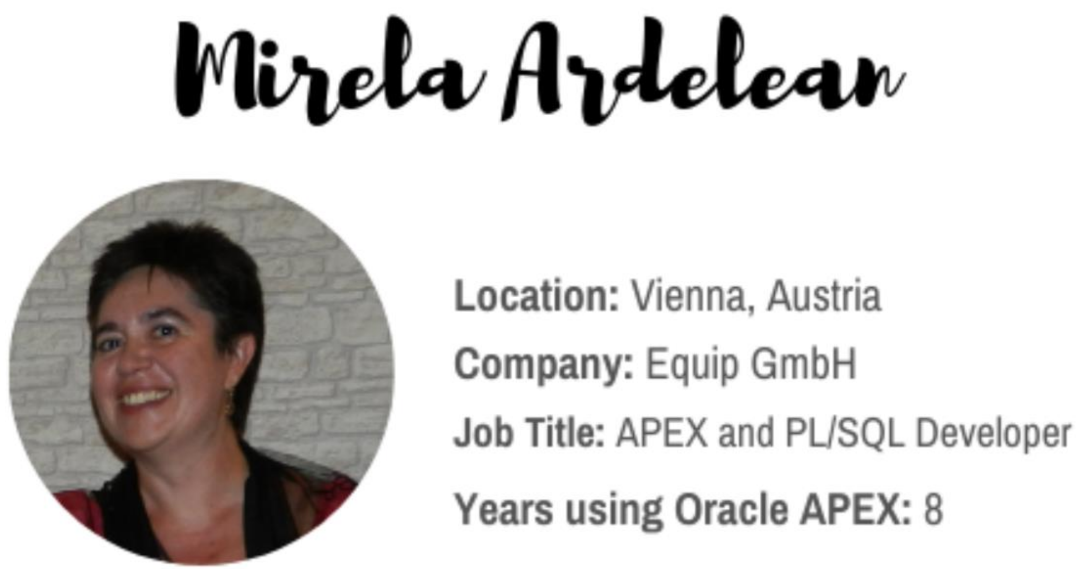
@LinoSchilde

Oracle ACE Pro

www.linkedin.com/in/lschilde

Intro

- It works best if it is applied in layers
- Poor or no security practices/standards
- Staying ahead of the game is hard
- We developers are first line of defence
- Session is focused on us developers



Mirela Ardelean

Location: Vienna, Austria

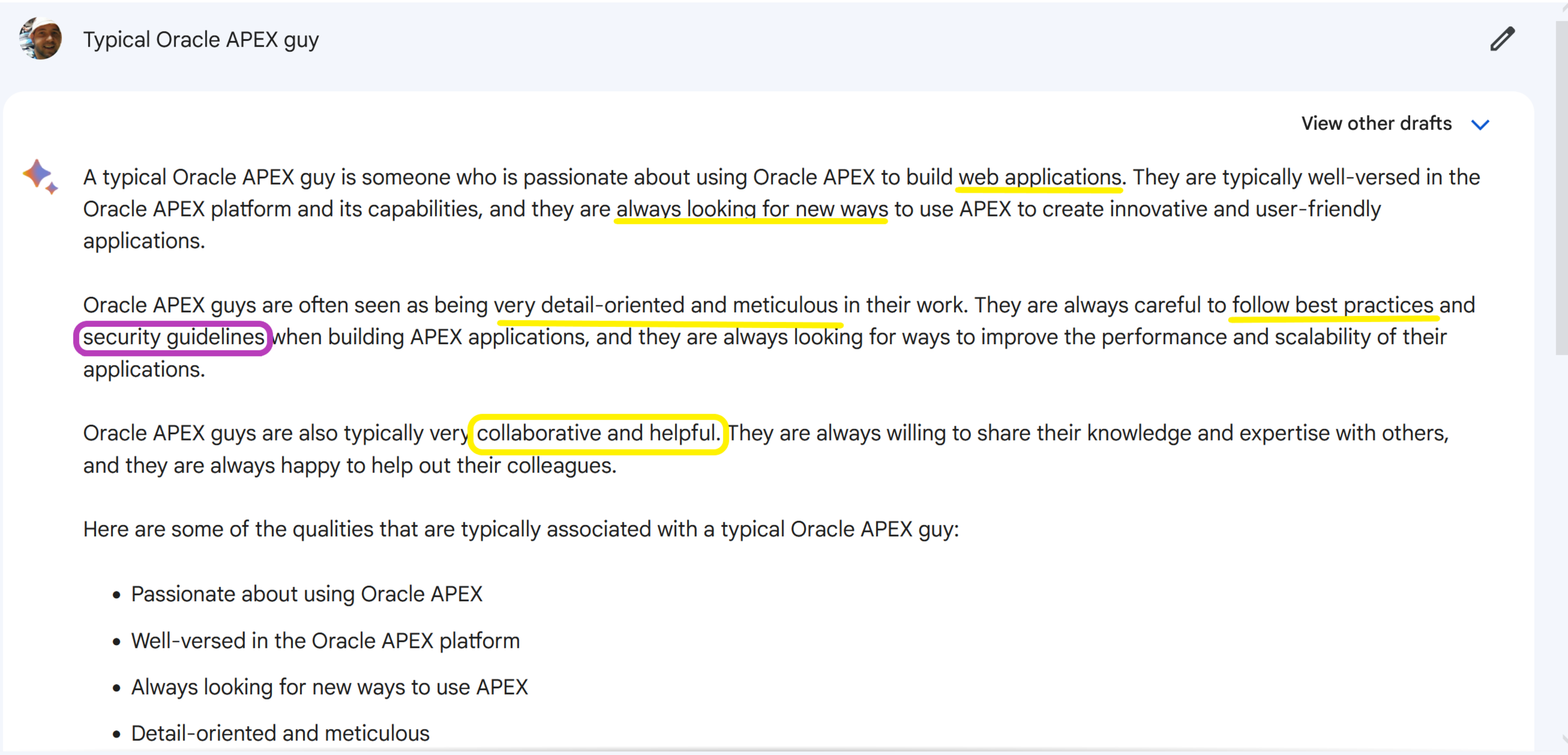
Company: Equip GmbH

Job Title: APEX and PL/SQL Developer

Years using Oracle APEX: 8

https://www.alamy.com

What does the world think of APEX developers?



Typical Oracle APEX guy

A typical Oracle APEX guy is someone who is passionate about using Oracle APEX to build web applications. They are typically well-versed in the Oracle APEX platform and its capabilities, and they are always looking for new ways to use APEX to create innovative and user-friendly applications.

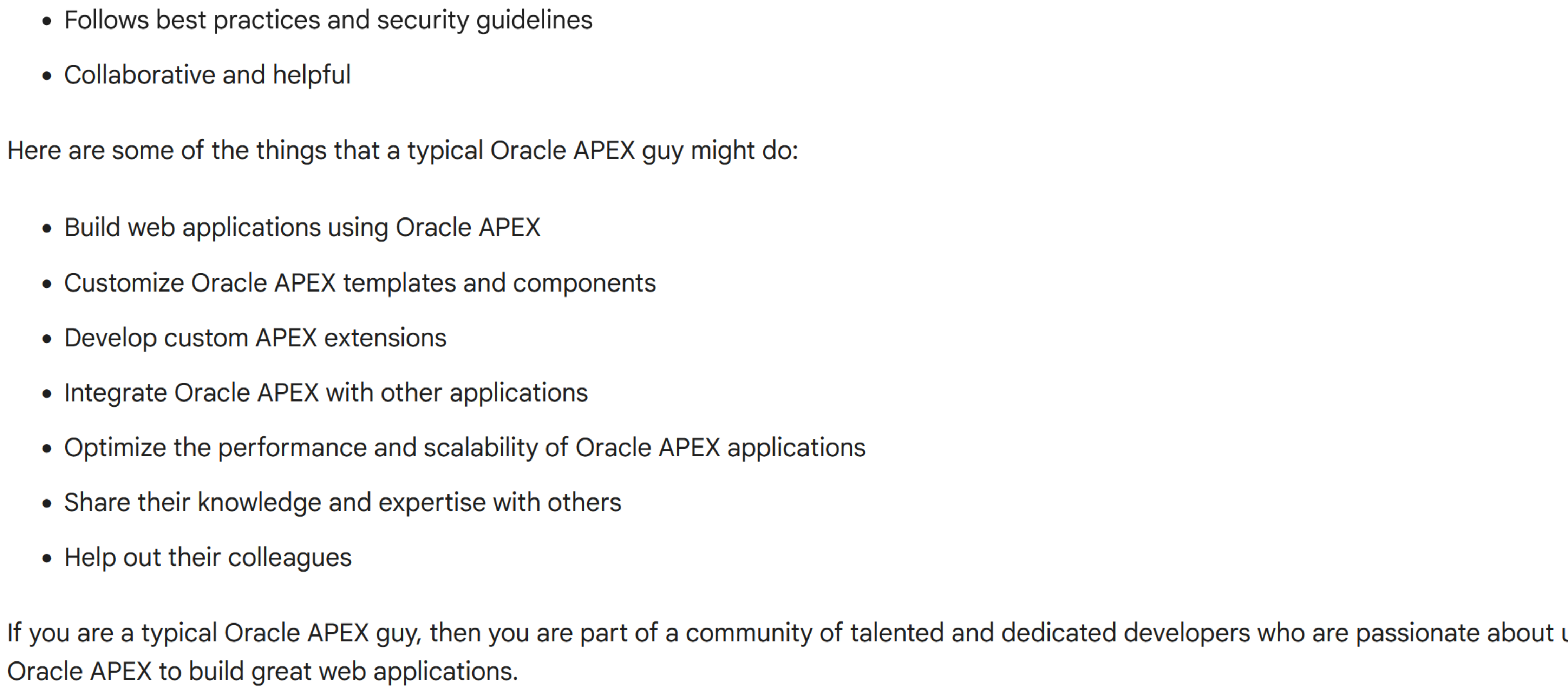
Oracle APEX guys are often seen as being very detail-oriented and meticulous in their work. They are always careful to follow best practices and security guidelines when building APEX applications, and they are always looking for ways to improve the performance and scalability of their applications.

Oracle APEX guys are also typically very collaborative and helpful. They are always willing to share their knowledge and expertise with others, and they are always happy to help out their colleagues.

Here are some of the qualities that are typically associated with a typical Oracle APEX guy:

- Passionate about using Oracle APEX
- Well-versed in the Oracle APEX platform
- Always looking for new ways to use APEX
- Detail-oriented and meticulous

https://bard.google.com/



- Follows best practices and security guidelines
- Collaborative and helpful

Here are some of the things that a typical Oracle APEX guy might do:

- Build web applications using Oracle APEX
- Customize Oracle APEX templates and components
- Develop custom APEX extensions
- Integrate Oracle APEX with other applications
- Optimize the performance and scalability of Oracle APEX applications
- Share their knowledge and expertise with others
- Help out their colleagues

If you are a typical Oracle APEX guy, then you are part of a community of talented and dedicated developers who are passionate about u Oracle APEX to build great web applications.

https://bard.google.com/

What do we know about APEX security!!!!!

ā'pěks
(#orclapex)

is
SAFE

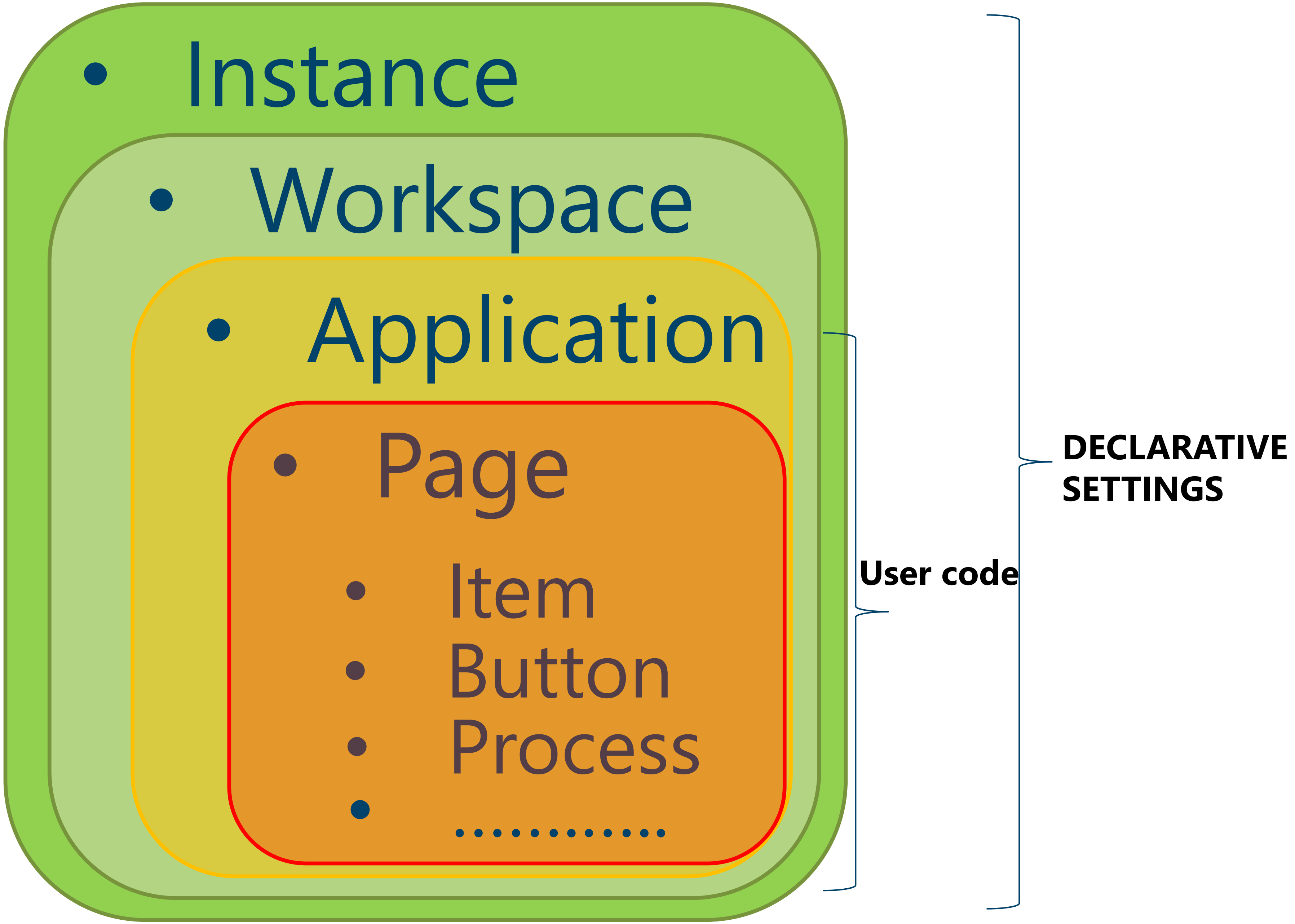
APEX features

Out of the box features

- Features we have available:
  - APEX URL
  - Authentication/Authorisation
  - Security settings
  - Session handling
  - Session state protection
  - Encoding APIs
  - Advisor

APEX provides mechanisms to secure your apps!

APEX settings



- Instance
  - Workspace
    - Application
      - Page
        - Item
        - Button
        - Process
        - ....

User code

DECLARATIVE SETTINGS

Settings – Application Definition

- 150+ APEX settings in total
- Rejoin session & Deep Linking
- Session Timeout & Session idle in seconds
- Error handling function
- Session State Protection (SSP)
- Authorization & Authentication

Settings – Page Security

- Authorization Scheme
- Authentication
- Rejoin Sessions
- Deep Linking

- Page Access Protection (PAP)
  - PAP does not provide any security but is required when SSP is enabled
  - Unrestricted
  - Arguments Must Have Checksum
  - No Arguments Supported
  - No URL Access
- Form Auto Complete & Double submission

Good practice

- Exporting and importing to production as run only
- Automate deployments

Error handling function

- We can easily define one
- Minimize amount of information we give back to the users
- Use it to improve user experience

GEM statements

- We heard about things like SQL injection and XSS scripting
- Session state values are stored as text - wwv_flow_data

“All our apps are internal so we do not have to worry about it”

HOW DO WE measure how secure are we coding?!!!!!

www.canva.com

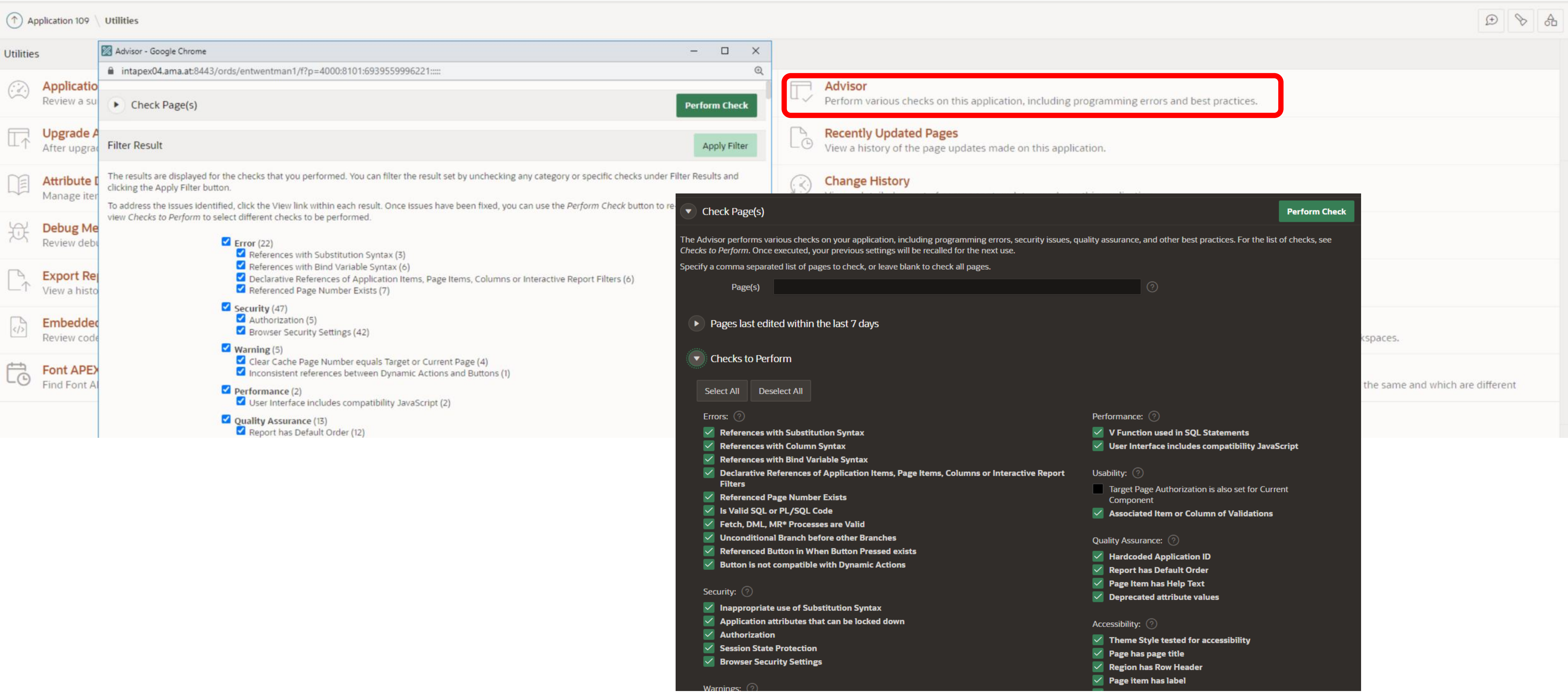
Here are some ideas

- Breadcrumbs
- Region title
- IR column header
- PL/SQL success message
- Built in substitution strings

How do we detect/prevent this!!!!!!

We need Security tools

APEX Advisor



The results are displayed for the checks that you performed. You can filter the result set by unchecking any category or specific checks under Filter Results and clicking the Apply Filter button.

To address the issues identified, click the View link within each result. Once issues have been fixed, you can use the *Perform Check* button to re view *Checks to Perform* to select different checks to be performed.

- ☑ **Error** (22)
  - ☑ References with Substitution Syntax (3)
  - ☑ References with Bind Variable Syntax (6)
  - ☑ Declarative References of Application Items, Page Items, Columns or Interactive Report Filters (6)
  - ☑ Referenced Page Number Exists (7)
- ☑ **Security** (47)
  - ☑ Authorization (5)
  - ☑ Browser Security Settings (42)
- ☑ **Warning** (5)
  - ☑ Clear Cache Page Number equals Target or Current Page (4)
  - ☑ Inconsistent references between Dynamic Actions and Buttons (1)
- ☑ **Performance** (2)
  - ☑ User Interface includes compatibility JavaScript (2)
- ☑ **Quality Assurance** (13)
  - ☑ Report has Default Order (12)

Check Page(s) **Perform Check**

The Advisor performs various checks on your application, including programming errors, security issues, quality assurance, and other best practices. For the list of checks, see *Checks to Perform*. Once executed, your previous settings will be recalled for the next use.

Specify a comma separated list of pages to check, or leave blank to check all pages.

Page(s)

Pages last edited within the last 7 days

Checks to Perform

Select All Deselect All

Errors:
- ☑ **References with Substitution Syntax**
- ☑ **References with Column Syntax**
- ☑ **References with Bind Variable Syntax**
- ☑ **Declarative References of Application Items, Page Items, Columns or Interactive Report Filters**
- ☑ **Referenced Page Number Exists**
- ☑ **Is Valid SQL or PL/SQL Code**
- ☑ **Fetch, DML, MR* Processes are Valid**
- ☑ **Unconditional Branch before other Branches**
- ☑ **Referenced Button in When Button Pressed exists**
- ☑ **Button is not compatible with Dynamic Actions**

Security:
- ☑ **Inappropriate use of Substitution Syntax**
- ☑ **Application attributes that can be locked down**
- ☑ **Authorization**
- ☑ **Session State Protection**
- ☑ **Browser Security Settings**

Warnings:

Performance:
- ☑ **V Function used in SQL Statements**
- ☑ **User Interface includes compatibility JavaScript**

Usability:
- ☐ Target Page Authorization is also set for Current Component
- ☑ **Associated Item or Column of Validations**

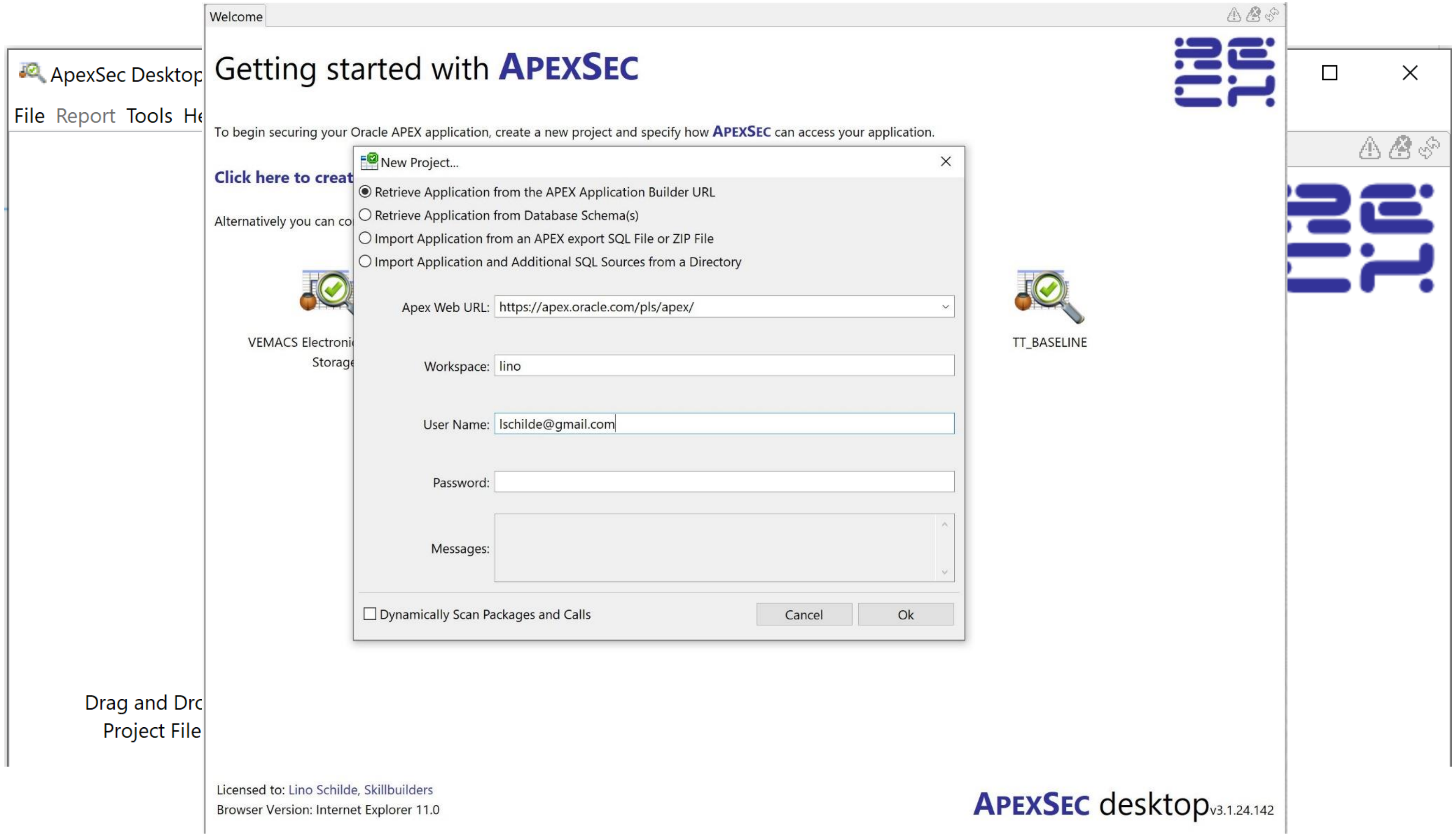
Quality Assurance:
- ☑ **Hardcoded Application ID**
- ☑ **Report has Default Order**
- ☑ **Page Item has Help Text**
- ☑ **Deprecated attribute values**

Accessibility:
- ☑ **Theme Style tested for accessibility**
- ☑ **Page has page title**
- ☑ **Region has Row Header**
- ☑ **Page item has label**

Security tools - APEX Advisor

- Takes time getting used to
- It is not a professional tool
- **App Builder -> Application X -> Utilities**
- It checks for errors, security issues, usability and quality assurance

APEXSec



Getting started with APEXSEC

To begin securing your Oracle APEX application, create a new project and specify how APEXSEC can access your application.

Click here to creat

Alternatively you can co

New Project...

● Retrieve Application from the APEX Application Builder URL
○ Retrieve Application from Database Schema(s)
○ Import Application from an APEX export SQL File or ZIP File
○ Import Application and Additional SQL Sources from a Directory

Apex Web URL: https://apex.oracle.com/pls/apex/

Workspace: lino

User Name: lschilde@gmail.com

Password:

Messages:

☐ Dynamically Scan Packages and Calls

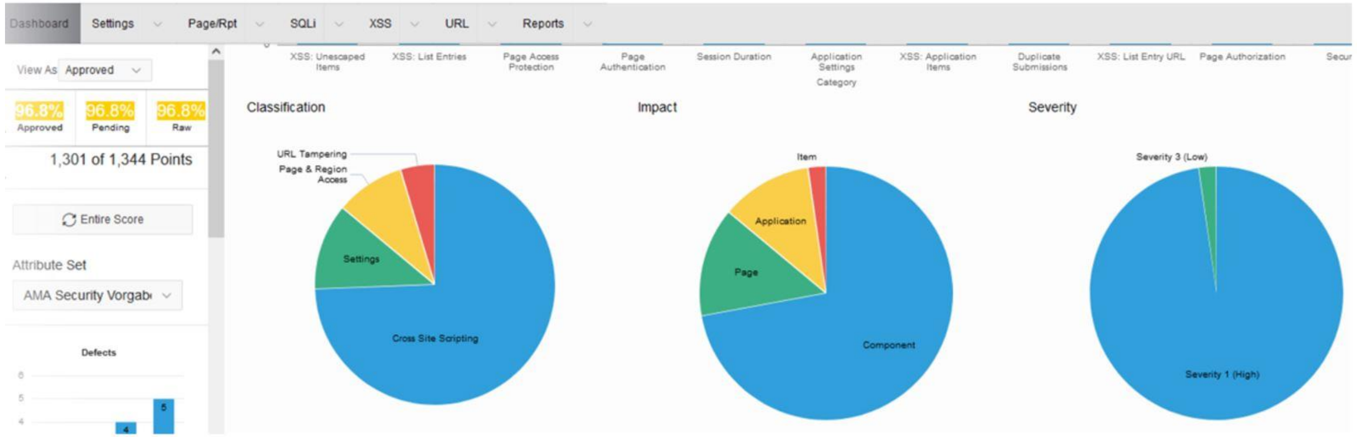
Licensed to: Lino Schilde, Skillbuilders
Browser Version: Internet Explorer 11.0

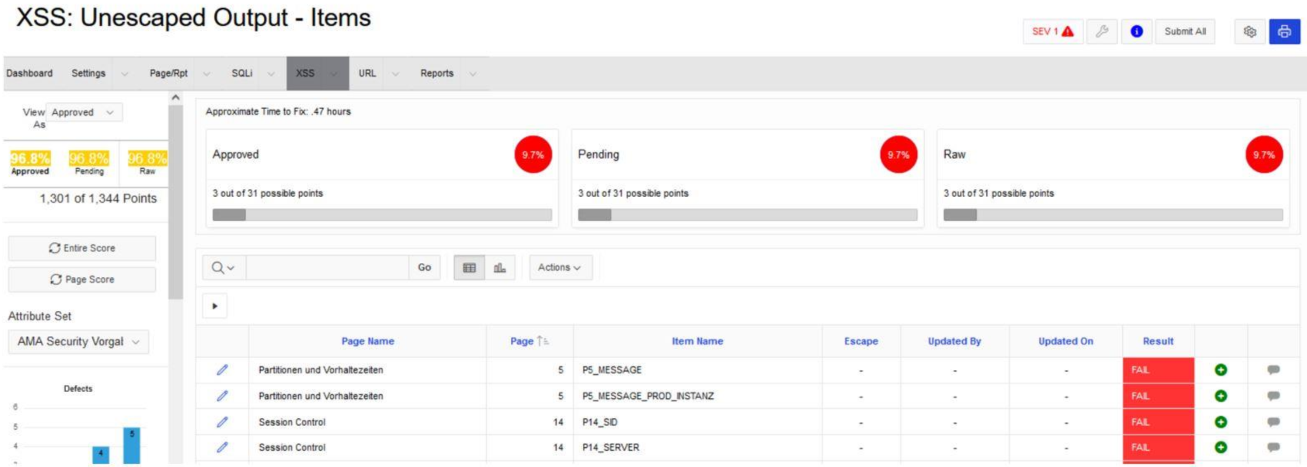
APEXSEC desktop v3.1.24.142

Security tools - APEXSec

- Licensing cost
- Ability to scan from live apps vs uploaded files
- Tests your PLSQL code too
- Available as desktop version
- Fairly simple to run
- With detailed instruction how to sanitize security flaws found

APEX SERT – free tool





XSS: Unescaped Output - Items

1,301 of 1,344 Points

Approximate Time to Fix: .47 hours

Approved 9.7%
3 out of 31 possible points

Pending 9.7%
3 out of 31 possible points

Raw 9.7%
3 out of 31 possible points

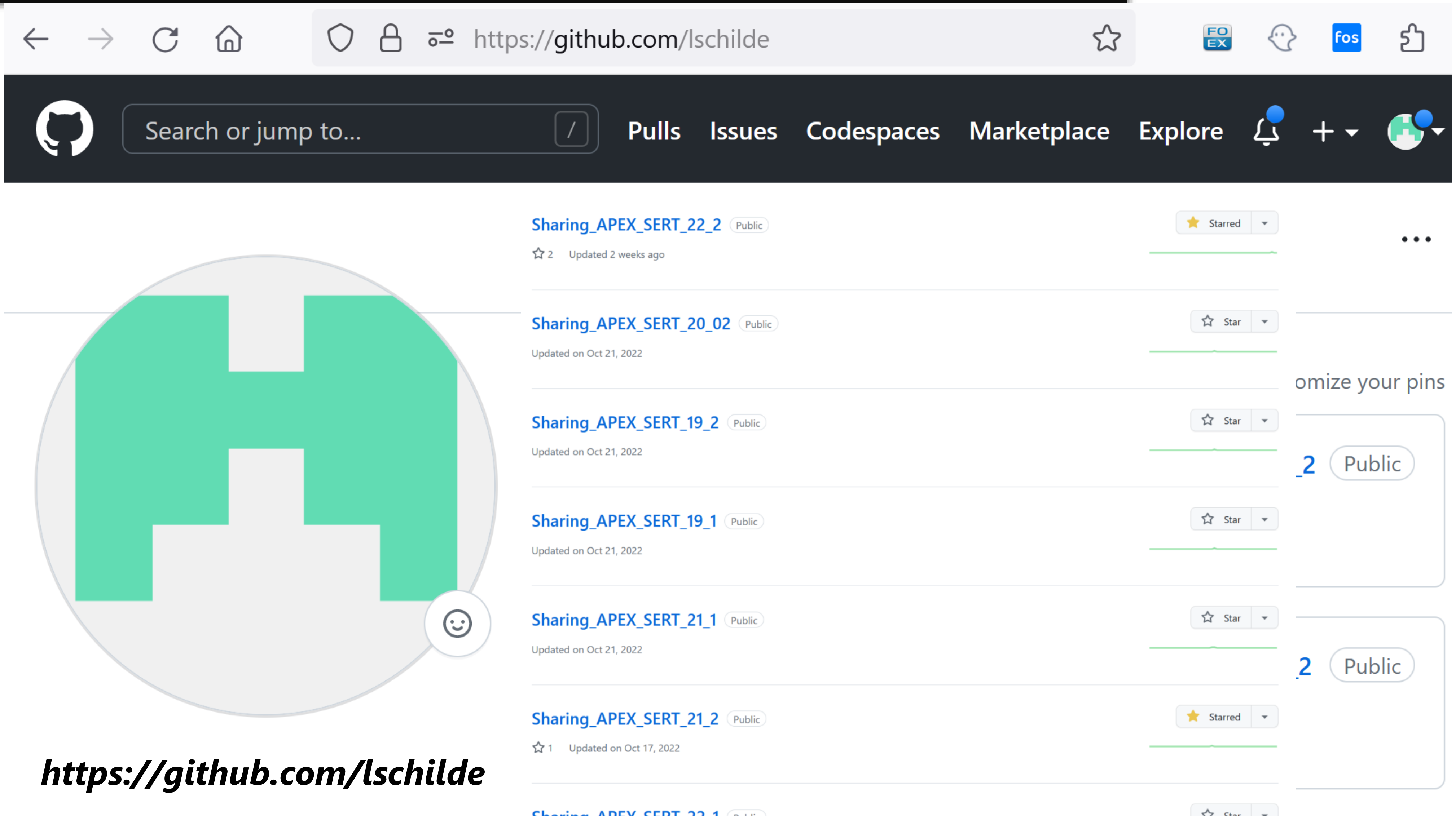
	Page Name	Page	Item Name	Escape	Updated By	Updated On	Result		
	Partitionen und Vorhaltezeiten	5	P5_MESSAGE	-	-	-	FAIL		
	Partitionen und Vorhaltezeiten	5	P5_MESSAGE_PROD_INSTANZ	-	-	-	FAIL		
	Session Control	14	P14_SID	-	-	-	FAIL		
	Session Control	14	P14_SERVER	-	-	-	FAIL		

https://github.com/OraOpenSource/apex-sert

Security tools - APEX SERT

- No licencing cost
- Ability to scan only ‘live’
- Will not tests your PLSQL code
- Community edition available? Original was outdated
- OpenSource tool needed on the market

Security tools - APEX SERT



Sharing_APEX_SERT_22_2 Public — Starred
2 · Updated 2 weeks ago

Sharing_APEX_SERT_20_02 Public — Star
Updated on Oct 21, 2022

Sharing_APEX_SERT_19_2 Public — Star
Updated on Oct 21, 2022

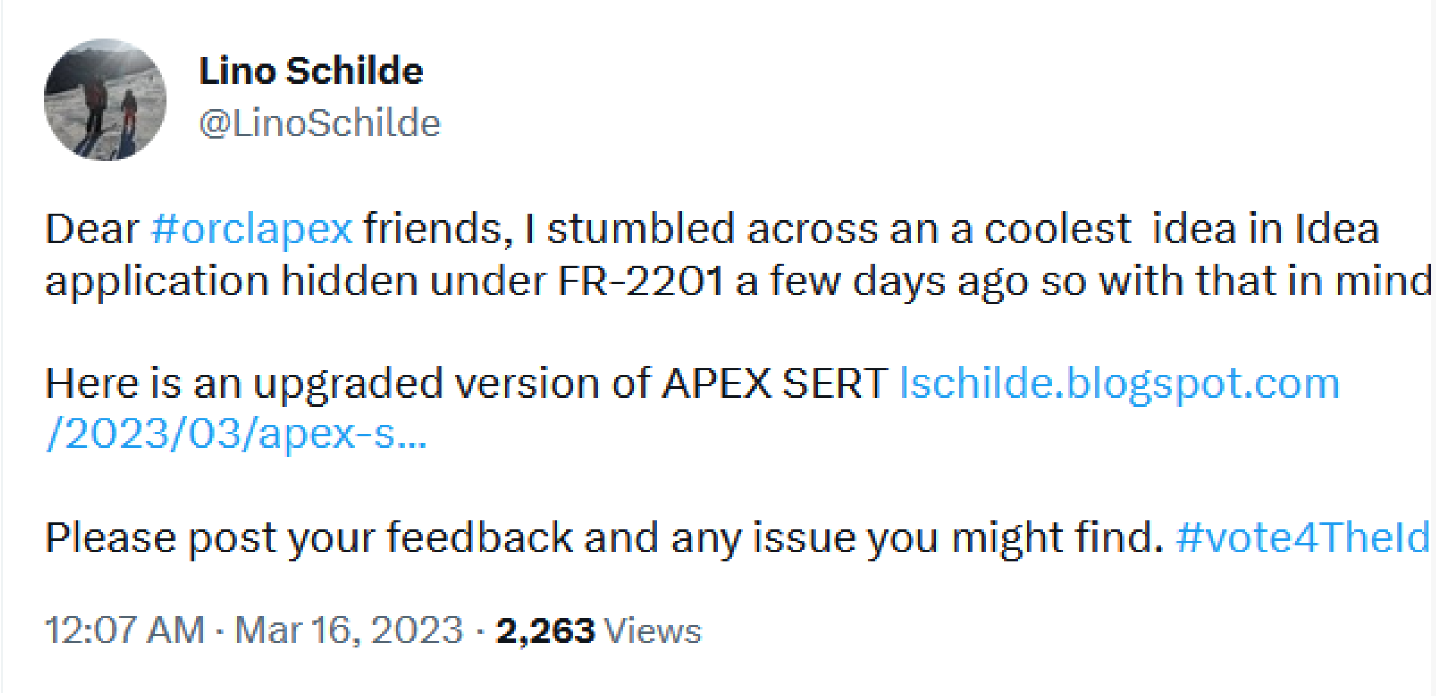
Sharing_APEX_SERT_19_1 Public — Star
Updated on Oct 21, 2022

Sharing_APEX_SERT_21_1 Public — Star
Updated on Oct 21, 2022

Sharing_APEX_SERT_21_2 Public — Starred
1 · Updated on Oct 17, 2022

Sharing_APEX_SERT_22_1 Public — Star

https://github.com/lschilde



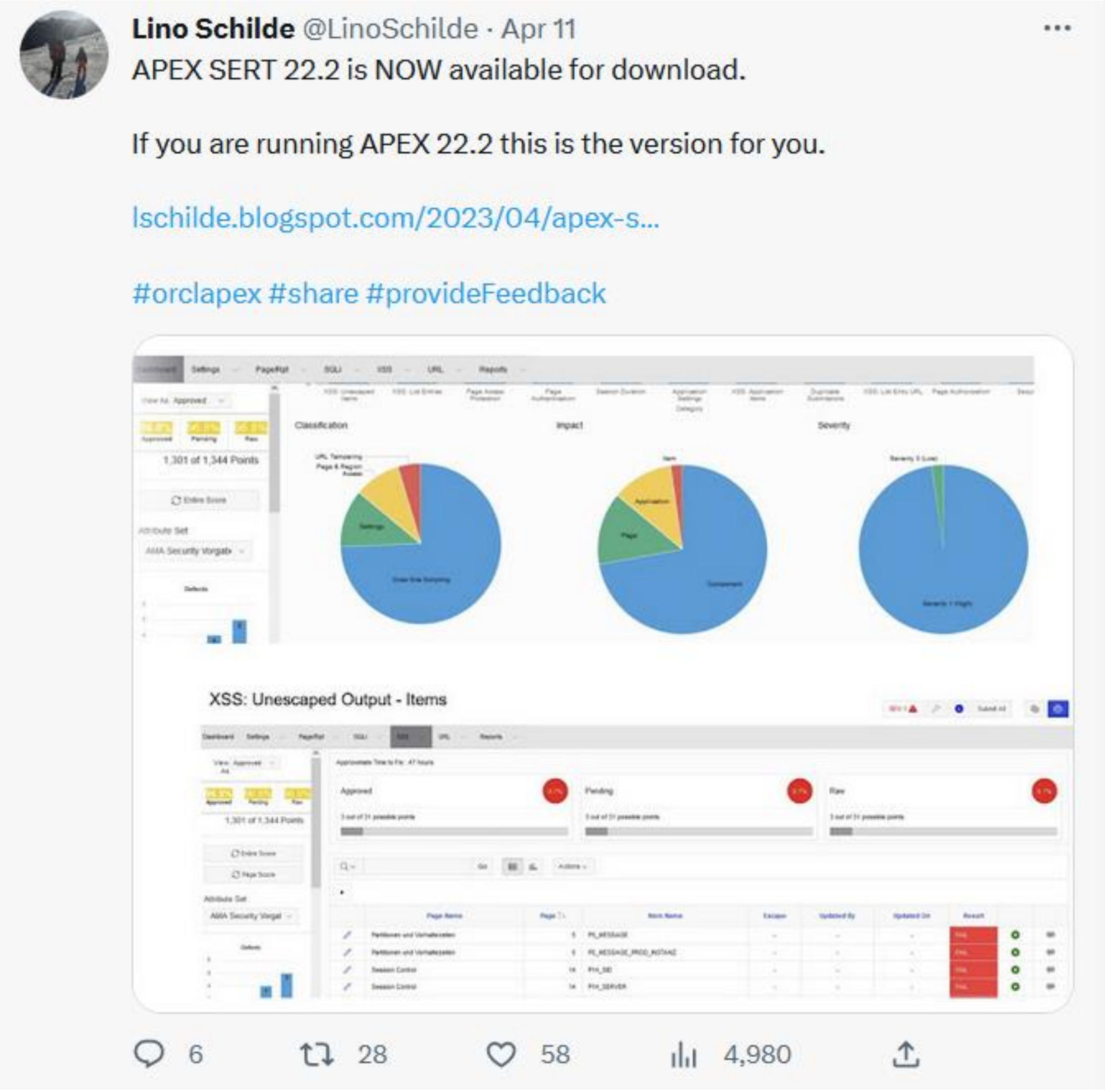
Lino Schilde
@LinoSchilde

Dear #orclapex friends, I stumbled across an a coolest idea in Idea application hidden under FR-2201 a few days ago so with that in mind

Here is an upgraded version of APEX SERT lschilde.blogspot.com/2023/03/apex-s...

Please post your feedback and any issue you might find. #vote4TheId

12:07 AM · Mar 16, 2023 · **2,263** Views



Lino Schilde @LinoSchilde · Apr 11

APEX SERT 22.2 is NOW available for download.

If you are running APEX 22.2 this is the version for you.

lschilde.blogspot.com/2023/04/apex-s...

#orclapex #share #provideFeedback

6 28 58 4,980

http://lschilde.blogspot.com

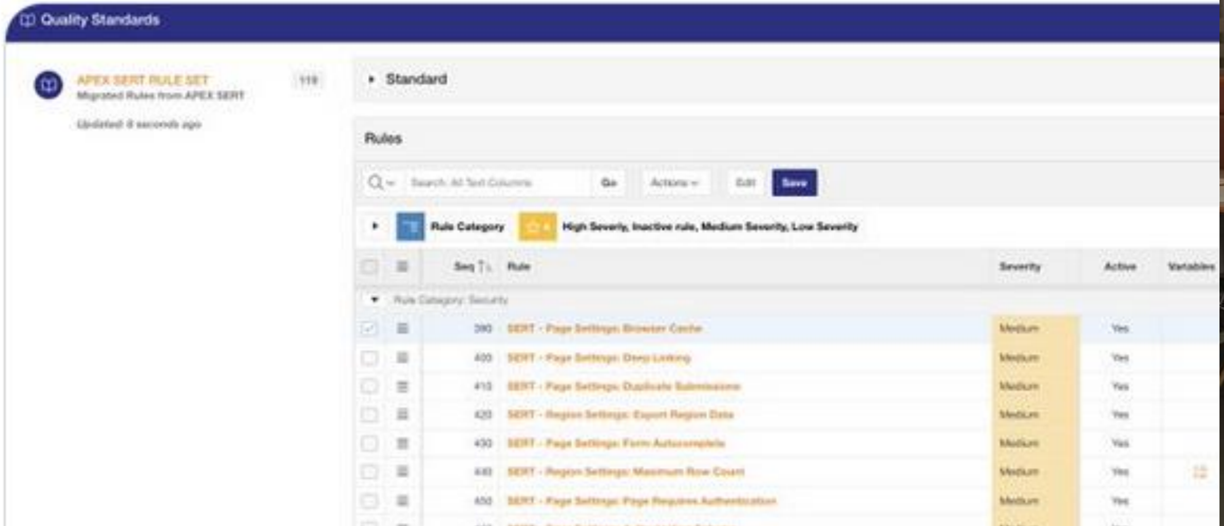
Security tools - APEX Project eye

APEX Project Eye (APE)
@APEXProjectEye

APEX-SERT quality assurance rules are now available to use with APEX Project Eye!

This rule set includes 100+ rules covering SQL injection, XSS, URL tampering, & much more. Helping you to create & maintain better more secure #orclapex apps.

#lowcode #qualityassurance



Quality Standards

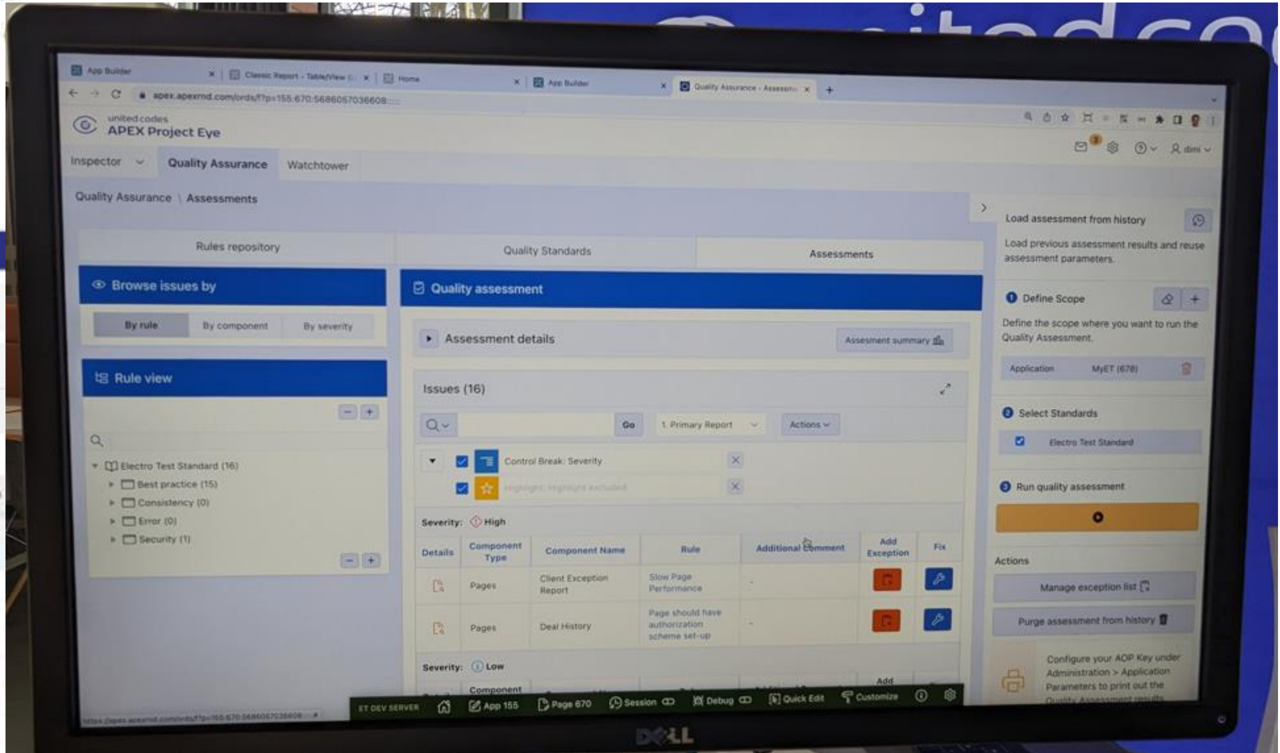
APEX SERT RULE SET 119
Migrated Rules from APEX SERT
Updated 8 seconds ago

Standard

Rules

Rule Category | High Severity, Inactive rule, Medium Severity, Low Severity

Seq	Rule	Severity	Active	Variables
Rule Category: Security				
390	SERT - Page Settings: Browser Cache	Medium	Yes	
400	SERT - Page Settings: Deep Linking	Medium	Yes	
410	SERT - Page Settings: Duplicate Submissions	Medium	Yes	
420	SERT - Region Settings: Export Region Data	Medium	Yes	
430	SERT - Page Settings: Form Autocomplete	Medium	Yes	
440	SERT - Region Settings: Maximum Row Count	Medium	Yes	
450	SERT - Page Settings: Page Requires Authentication	Medium	Yes	



Electro Test Standard (16)
Best practice (15)
Consistency (0)
Error (0)
Security (1)

Issues (16)

Control Break: Severity

Severity: High

Details	Component Type	Component Name	Rule	Additional Comment	Add Exception	Fix
	Pages	Client Exception Report	Slow Page Performance	-		
	Pages	Deal History	Page should have authorization scheme set-up	-		

Severity: Low

Load previous assessment results and reuse assessment parameters.

Define the scope where you want to run the Quality Assessment.

Application MyET (678)

Electro Test Standard

Configure your AOP Key under Administration > Application Parameters to print out the Quality Assessment results

Demo - Security tools

What does it look for?

Security scans

- Various settings
- Page and Item SSP
- Potential XSS and SQL Injections
- Button and page processes
- Authentications and authorizations

It is all about inconsistency

TIP #1
Keep your data clean

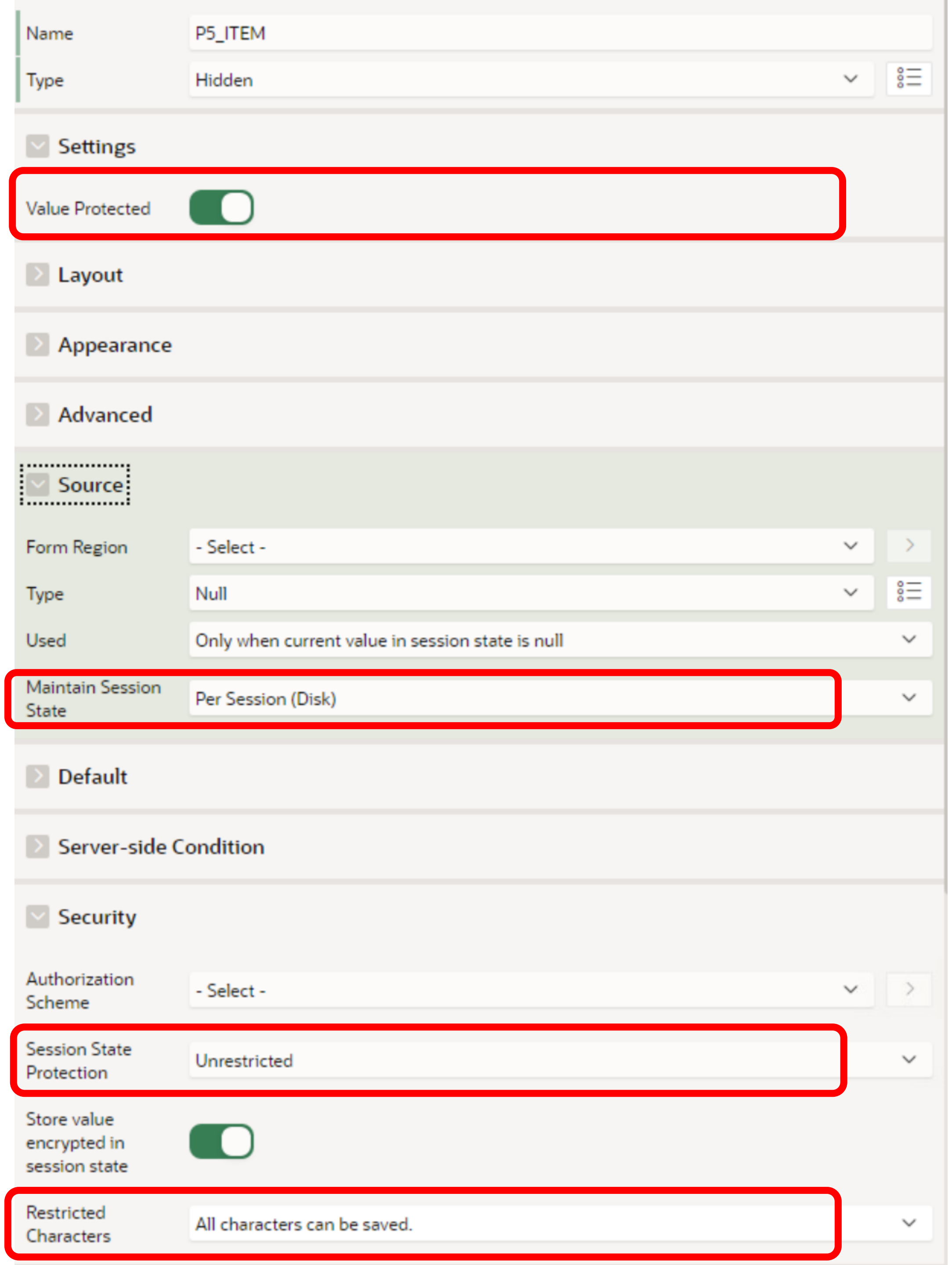
User inputs checks

- Sanitize and validate it
- Do not trust it
- >50% of security risk
- Normally never done and ignored

TIP #2
Know your items

Items tips

- Not all items are the same
- Use SSP appropriately
- Store encrypted in session state
- Maintain Session State
- Use
  - Value protected vs escape special character or (Format Settings)
- Restricted characters



Name	P5_ITEM
Type	Hidden

Settings

Value Protected	●

Layout

Appearance

Advanced

Source

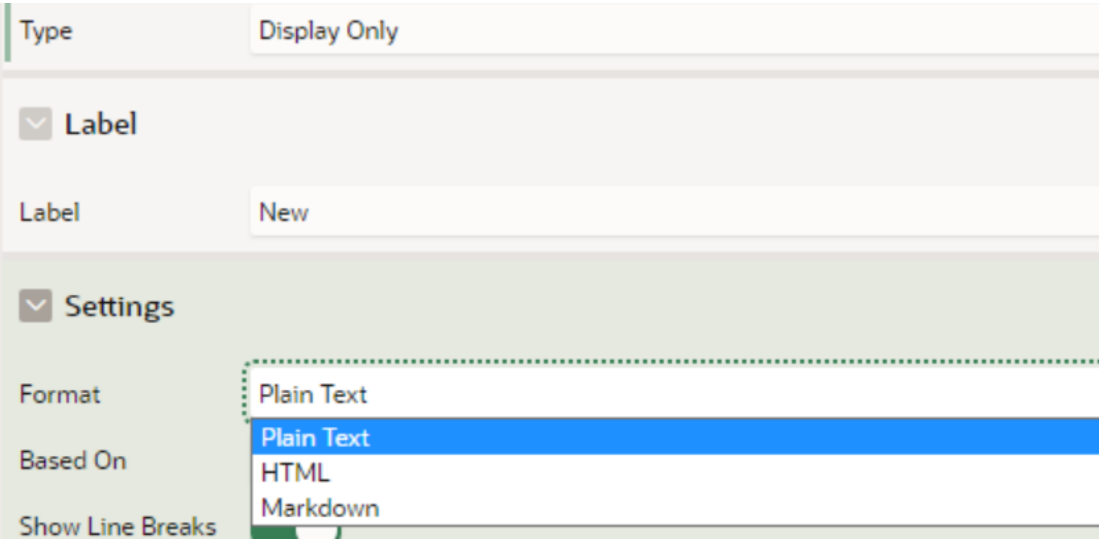
Form Region	- Select -
Type	Null
Used	Only when current value in session state is null
Maintain Session State	Per Session (Disk)

Default

Server-side Condition

Security

Authorization Scheme	- Select -
Session State Protection	Unrestricted
Store value encrypted in session state	●
Restricted Characters	All characters can be saved.



Type	Display Only

Label

Label	New

Settings

Format	Plain Text
Based On	
Show Line Breaks	

Items

Application items

- Used for Server-side logic items
- Set SSP to Restricted
- Very often forgotten and left exposed

User editable items

- Page - Set SSP to Checksum
- Validate and sanitize before submitting it to DB
- Do not trust user inputs

Hidden items

- Not displayed but rendered in an HTML
- Hidden item can be modified
- If JavaScript modifies the value then it cannot be fully protected
- For Items passed as data between pages:
  - Set Page SSP to Checksum
  - Set PAP of receiving page to Checksum
  - Set Value Protected to Yes

Display items

- Non-enterable text item
- Values can still be changed
- Restricted - May not be set from browser
- Use escape special character or Format Settings like plain/HTML/markdown
- Make sure to check:
  - if it can be set by end users or if it used in JS as substitution string

TIP #3
Where (should) do we use HTML?

like substitution variables

APEX and built in HTML

- Everywhere where APEX lets us?
- Watch out for Substitution variables!

OK

- Page title, region title and process success message

NOT OK

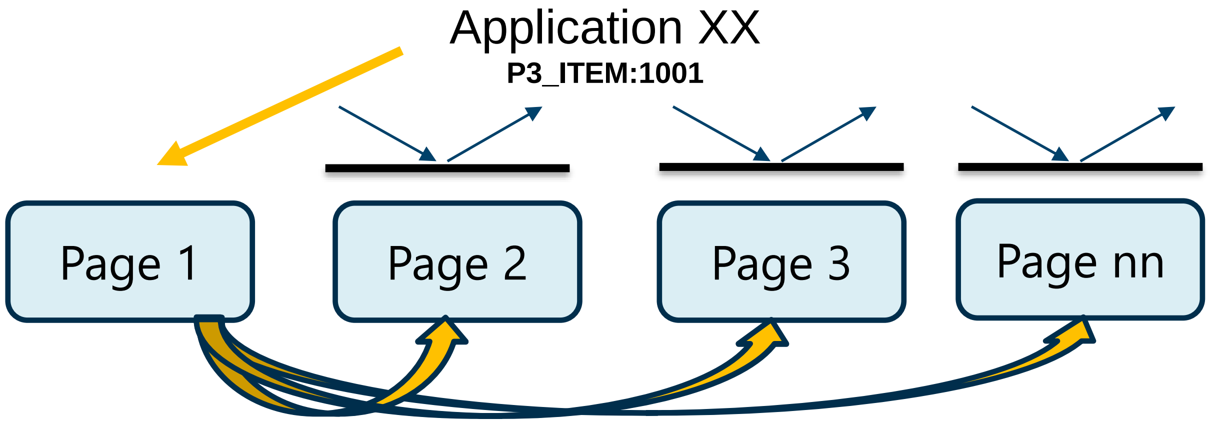
- Substitution variable in JavaScript
- Report columns title
- Breadcrumbs entries

TIP #4 APEX URL

Remember

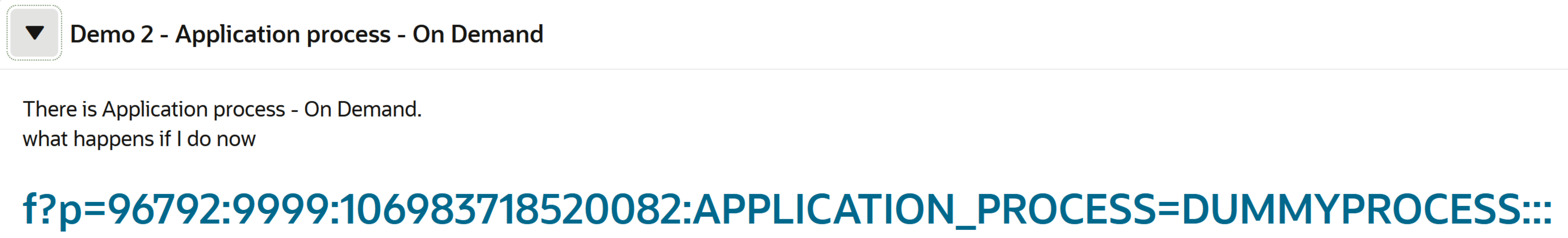
f?p=App:Page:Session:Request:Debug:ClearCache:itemNames:itemValues

- Tampering is the first way
- Changing URL parameters or changing the values using JavaScript



URL is powerful

- It can run your processes too
  - Page or Application process



▼ **Demo 2 - Application process - On Demand**

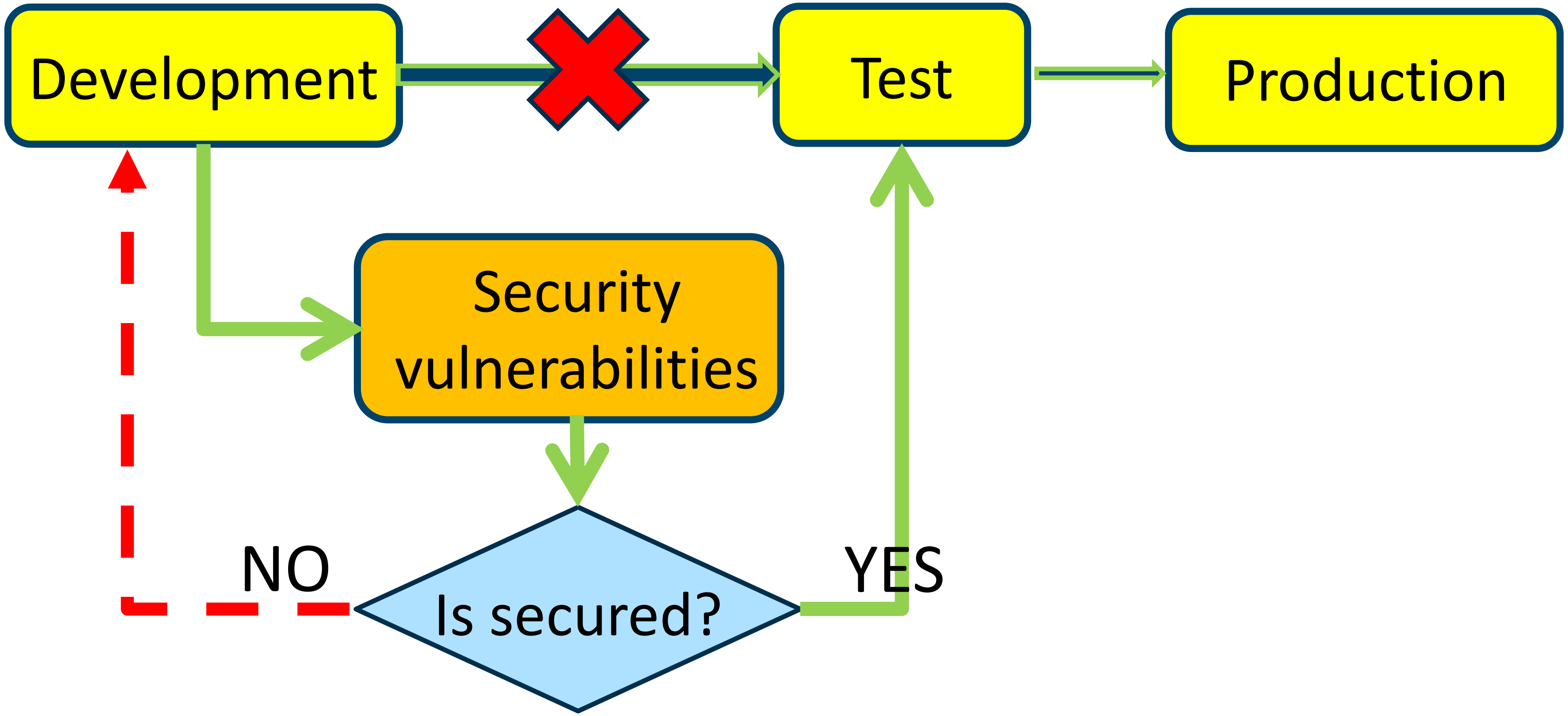
There is Application process - On Demand.
what happens if I do now

f?p=96792:9999:106983718520082:APPLICATION_PROCESS=DUMMYPROCESS:::

TIP #5
Being consistent is hard

Improvements to DEV practice

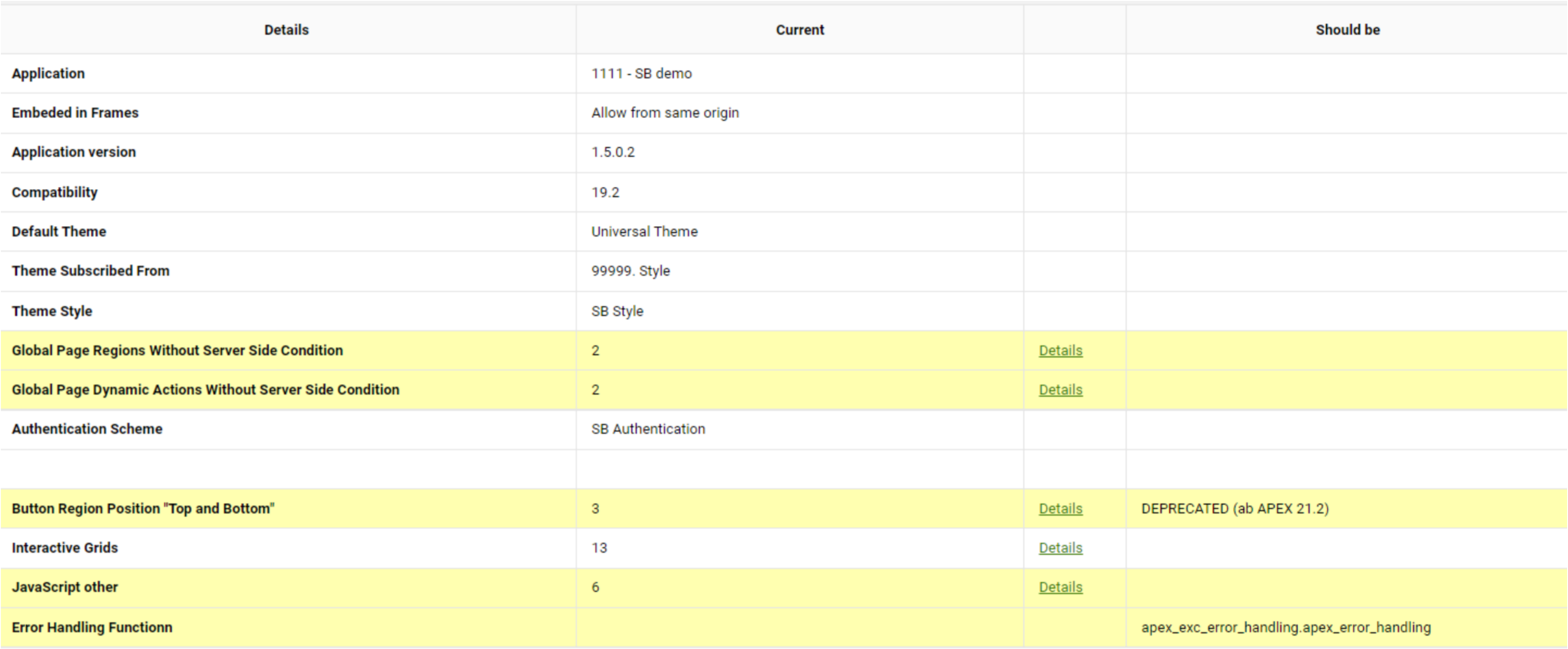
Idea?


Development
Test
Production
Security vulnerabilities
NO
Is secured?
YES

DEV Processes

- Export apps
    - With No debug and as Run Application Only
- Run your APEX Advisors/testing tools
- Have internally built tool to monitor common settings

DEV Apps



Details	Current		Should be
Application	1111 - SB demo		
Embeded in Frames	Allow from same origin		
Application version	1.5.0.2		
Compatibility	19.2		
Default Theme	Universal Theme		
Theme Subscribed From	99999. Style		
Theme Style	SB Style		
Global Page Regions Without Server Side Condition	2	Details	
Global Page Dynamic Actions Without Server Side Condition	2	Details	
Authentication Scheme	SB Authentication		
Button Region Position "Top and Bottom"	3	Details	DEPRECATED (ab APEX 21.2)
Interactive Grids	13	Details	
JavaScript other	6	Details	
Error Handling Functionn			apex_exc_error_handling.apex_error_handling

Selenium

LCT

Cypress

Automated testing

Playwright?!!!!

Ghost Inspector

WHY Automated testing?!

- **Improves software quality**
- **Helps with APEX security**
- Shorter test cycles
- Saves Time and Money
- **Faster APEX Upgrading**
- Return on investment is high

- Reduces manual errors
- Improves Standards
- Faster time to market
- Faster Feedback
- Improved Productivity
- Faster Debugging

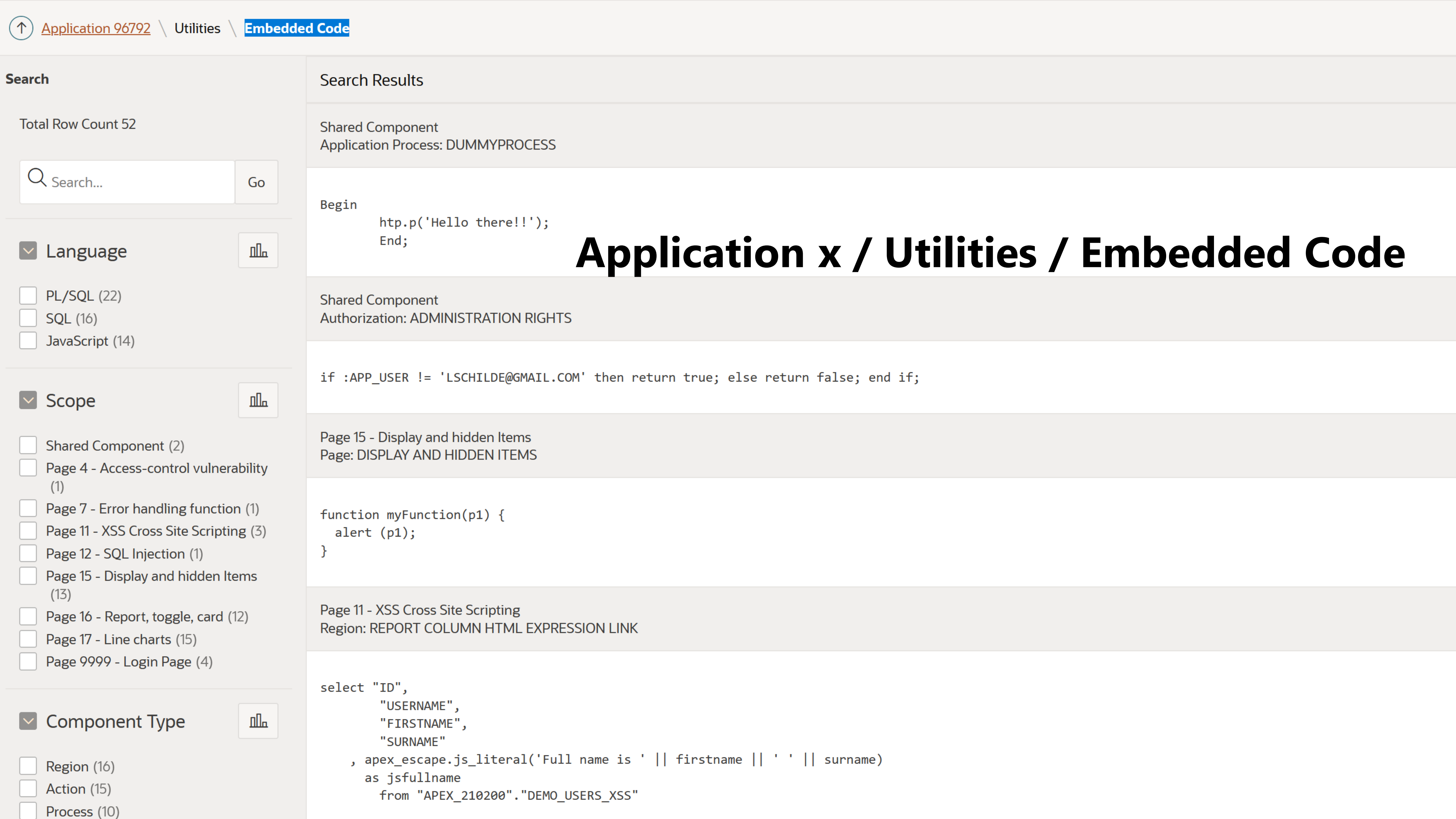
Monitor and logging

- All invalid authentication are logged

```
  select * from apex_workspace_access_log
OR
   select * from apex_workspace_activity_log
```

- Attackers will try to break into the system
- Logs are kept for 2 weeks so consider backing them up

Application x / Utilities / Embedded Code



Total Row Count 52

Language

- □ PL/SQL (22)
- □ SQL (16)
- □ JavaScript (14)

Scope

- □ Shared Component (2)
- □ Page 4 - Access-control vulnerability (1)
- □ Page 7 - Error handling function (1)
- □ Page 11 - XSS Cross Site Scripting (3)
- □ Page 12 - SQL Injection (1)
- □ Page 15 - Display and hidden Items (13)
- □ Page 16 - Report, toggle, card (12)
- □ Page 17 - Line charts (15)
- □ Page 9999 - Login Page (4)

Component Type

- □ Region (16)
- □ Action (15)
- □ Process (10)

Search Results

Shared Component
Application Process: DUMMYPROCESS

```
Begin
        htp.p('Hello there!!');
        End;
```

Shared Component
Authorization: ADMINISTRATION RIGHTS

```
if :APP_USER != 'LSCHILDE@GMAIL.COM' then return true; else return false; end if;
```

Page 15 - Display and hidden Items
Page: DISPLAY AND HIDDEN ITEMS

```
function myFunction(p1) {
  alert (p1);
}
```

Page 11 - XSS Cross Site Scripting
Region: REPORT COLUMN HTML EXPRESSION LINK

```
select "ID",
        "USERNAME",
        "FIRSTNAME",
        "SURNAME"
    , apex_escape.js_literal('Full name is ' || firstname || ' ' || surname)
      as jsfullname
        from "APEX_210200"."DEMO_USERS_XSS"
```

APEX Views - Information

- It is all in meta data – APEX views:
  - **apex_applications**
  - apex_application_static_files
  - apex_application_processes
  - apex_application_pages
  - **apex_application_page_regions**
  - **apex_application_page_proc**
  - apex_application_page_items
  - apex_application_page_buttons

apex_application_page_reg_cols
apex_application_page_rpt_cols
apex_appl_page_igs
apex_appl_page_ig_columns
apex_application_page_ir
apex_application_page_ir_col
apex_application_page_da_acts
apex_application_list_entries
apex_application_lists
apex_application_lovs
apex_application_page_map_layers

APEX Views - We can find/check

- **Application settings**
- Application processes with no authorizations
- Application processes for Dynamic SQL
- Application computations for Dynamic SQL
- **Page settings**
- Page regions with no authorizations
- Page processes with no authorizations
- Page regions with Dynamic SQL
- Page regions with substitution variables
- Buttons with no authorizations

- Reports with JS links
- Reports with non escaped items
- Global and on page JS
- **Region sources for SQL injection**
- **Dynamic PLSQL region**
- Deprecated APIs
- Deprecated APEX features

(**tabular forms**, date pickers)

…and address

- List items with no authorizations
- Items with invalid settings (hidden or display)
- Items default values (source) checks for Dynamic SQL
- **Editable IGs with no authorizations**
- Page Computations
- Page Validations
- **Dynamic actions with substitution variables**
- **Page JS with substitution variable**

ā'pěks
(#orclapex)

is SAFE

Summary

- Do not trust user inputs
  - Sanitize & validate
  - Make sure it gets escaped when used
- Protect your items
  - We can not change values using URL or JavaScript
  - Use SSP, restricted characters and value protected
- Be careful with using &PX_ITEM. syntax
  - Region, error & success messages are safe places
- Use security & auto-testing tools
- Build your own checks based on APEX views

Summary

- Minimum is to utilize authentication and authorization policies as minimum
- Authentication - “best practices”
- Error handling function
- Inconsistencies can lead to vulnerabilities
- Keep APEX up-to date

Summary SQL Injection

- Do use bind variables with care:
  - ! If used in Dynamic SQL bind variable needs to embedded in the string
- If you are forced to use &ITEM. syntax:
  - Check where data is coming from
  - Use appropriate escaping methods available
- DBMS_ASSERT and APEX_ESCAPE
- Reduce the impact surface

Summary SQL Injection

- APEX_COLLECTION
  - CREATE_COLLECTION_FROM_QUERY
  - CREATE_COLLECTION_FROM_QUERY2
  - CREATE_COLLECTION_FROM_QUERY_B
  - CREATE_COLLECTION_FROM_QUERY_B2
  - MERGE_MEMBERS
- APEX_ITEM
  - POPUP_FROM_QUERY
  - POPUPKEY_FROM_QUERY
  - SELECT_LIST_FROM_QUERY
  - SELECT_LIST_FROM_QUERY_XL
  - TEXT_FROM_LOV_QUERY
- DBMS_SQL.PARSE

Summary XSS

- Restrict, sanitize, validate and escape user inputs per context
- APEX_ESCAPE
- APEX_JAVASCRIPT.ESCAPE
- APEX_UTIL.URL_ENCODE
- HTF.ESCAPE_SC
- apex.util
  - escapeCSS, escapeHTML, escapeHTMLAttr

Summary XSS

- &PX_ITEM!HTML.
- &PX_ITEM!JS.
- &PX_ITEM!ATTR.
- &PX_ITEM!RAW.
- &PX_ITEM!STRIPHTML.

How much can we do ourself?

Yes we can do most!

Q&A

Thank you for attending

Typical examples

- URL tempering
- Access control inconsistency
- Hidden or Display only items
- JavaScript submit page
- Unprotected application processes or items

XSS examples

- Reports Column Display Type
- Disabling Escape special characters
- Report Column Formatting - HTML Expressions
- Report Column Formatting - Column Link
- Report Column - List of Values
- HTP.p

SQL injection examples

- Input changes the expected results
- Function returning Query
- Most things with Substitution variables
- Htp.p
- DYNAMIC SQL
  - Cursors
  - EXECUTE IMMEDIATE
  - APEX API