



Rafał Grzegorczyk
Senior APEX Developer



Oracle ACE
Pro

Oracle's SQLcl CI/CD bombshell: It will forever change your APEX & Database deployments

UKOUG2025 - Birmingham, UK

Sunday, 30 Nov 2025 4:00pm – 4:45pm



pretius

ORACLE

Partner

Liquibase



Oracle SQLcl “Project”

— —

Is Oracle's latest SQLcl feature
the only CI/CD Tool
for APEX
and Oracle Database
YOU will ever need?

— —



Scan to meet us
Pretius.com

Providing what you **need**, not what you **want**

300+
People

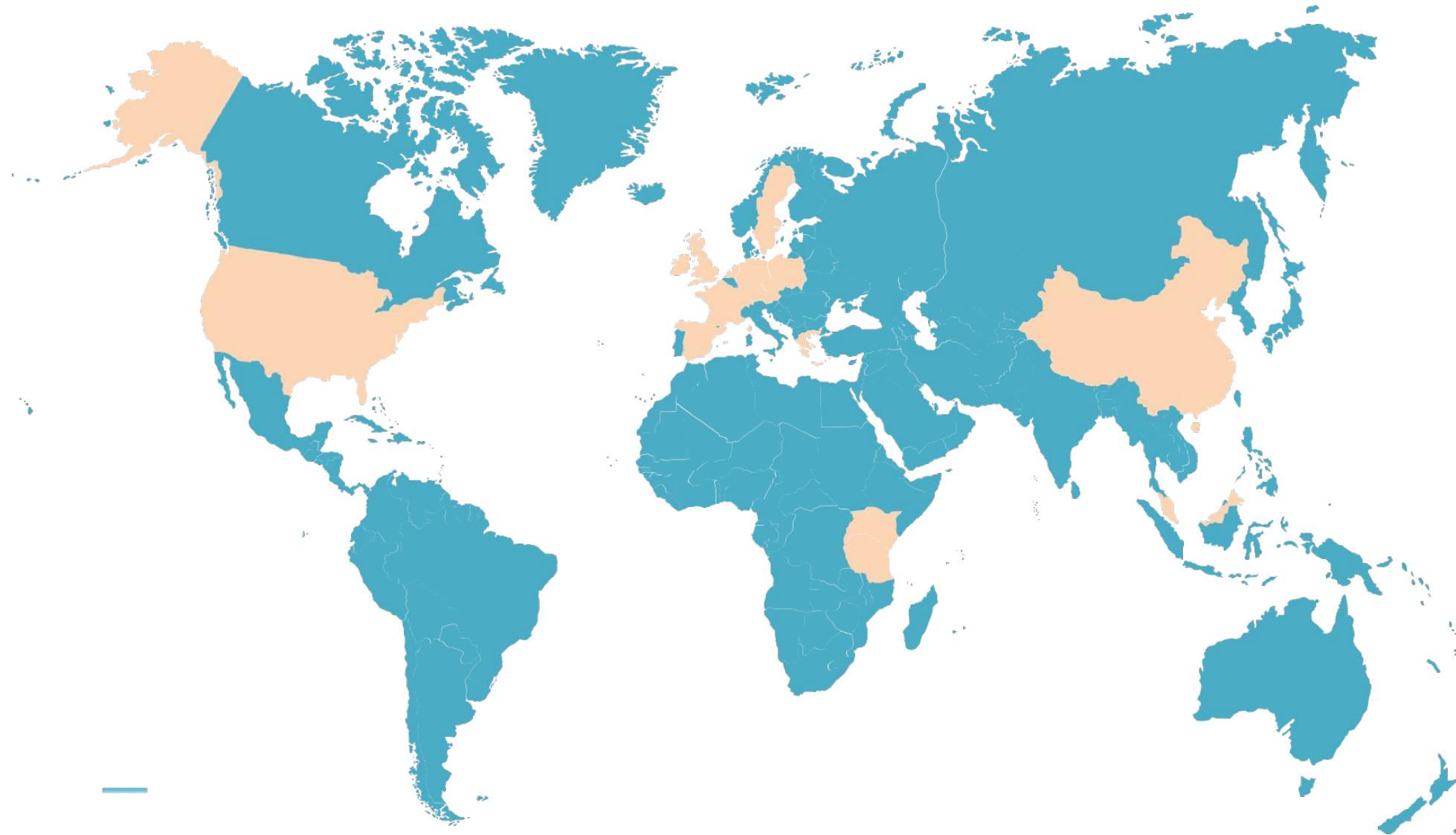
60+
APEX Devs

5
Oracle ACEs

70+
Customers

4
Continents

2
Warsaw, PL
London, UK



ORACLE

Partner

Liquibase



ISO 27001

About me

- 1 Kielce, Poland - 200km (125miles) south from Warsaw
 - 2 Oracle APEX and PL/SQL Developer
 - 3 Worldwide speaker since 2021
 - 4 Oracle ACE Pro
- Find me at:

rgrzegorczyk@pretius.com
[rgrzegorczyk](https://www.linkedin.com/in/rgrzegorczyk/) 
<https://rafal.hashnode.dev/>



Rafal Grzegorczyk

What is Oracle SQLcl?

Oracle SQLcl



- Oracle SQL Developer Command Line (SQLcl)
- Free command line interface for Oracle Database
- Allows to execute SQL and PL/SQL
- Provides in-line editing, statement completion
- Supports SQL*Plus scripts, allows to access Cloud Storage
- Data Pump, Codescan, MCP..... and much more

Title and Copyright Information

Preface

1 Changes in Release 25.3 for Oracle SQLcl User's Guide

- ▶ 2 Using Oracle SQLcl
- ▶ 3 Using the Oracle SQLcl MCP Server
- ▶ 4 Running SQLcl as a Scheduler Daemon
- ▶ 5 Using Liquibase
- ▶ 6 Database Application CI/CD
- ▶ 7 Using Cloud Storage
- ▶ 8 Using Data Pump
- ▶ 9 Using the Cloud Premigration Advisor Tool
- ▶ 10 Using the PGQL Plug-in
- ▶ 11 SQLcl Command Reference

A List of Unsupported SQL*Plus Commands and Features

B SQL Performance Troubleshooting

C Third-Party License Information

“Project” command

“Project” command

- “Project” is an CI/CD extension in SQLcl
- Introduced in 2nd half of 2024
- Contains several sub-commands to help developers manage database & APEX changes
- It uses Liquibase & GIT in a simple & developer-friendly way

Title and Copyright Information

Preface

1 Changes in Release 25.3 for Oracle SQLcl User's Guide

▶ 2 Using Oracle SQLcl

▶ 3 Using the Oracle SQLcl MCP Server

▶ 4 Running SQLcl as a Scheduler Daemon

▶ 5 Using Liquibase

▶ 6 Database Application CI/CD

▶ 7 Using Cloud Storage

▶ 8 Using Data Pump

▶ 9 Using the Cloud Premigration Advisor Tool

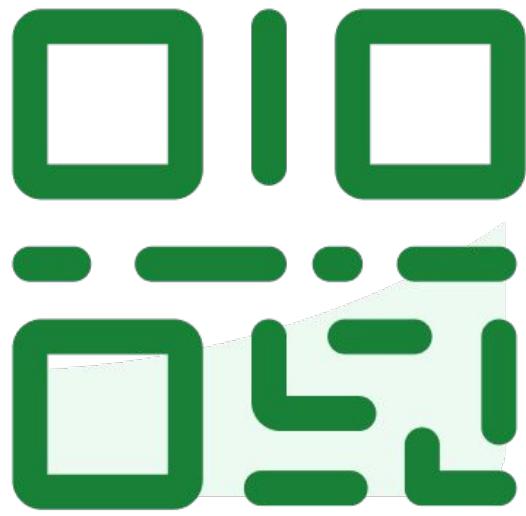
▶ 10 Using the PGQL Plug-in

▶ 11 SQLcl Command Reference

A List of Unsupported SQL*Plus Commands and Features

B SQL Performance Troubleshooting

C Third-Party License Information



**Join at [slido.com](https://www.slido.com)
#2417443**



Presenting with animations, GIFs or speaker notes? Enable our [Chrome extension](#)

slido

Questions & Answers

app.sli.do

R_Gregorczyk.Oracle's SQLcl CI/CD b...

Q&A Polls

Type your question

Ask the first one!

slido

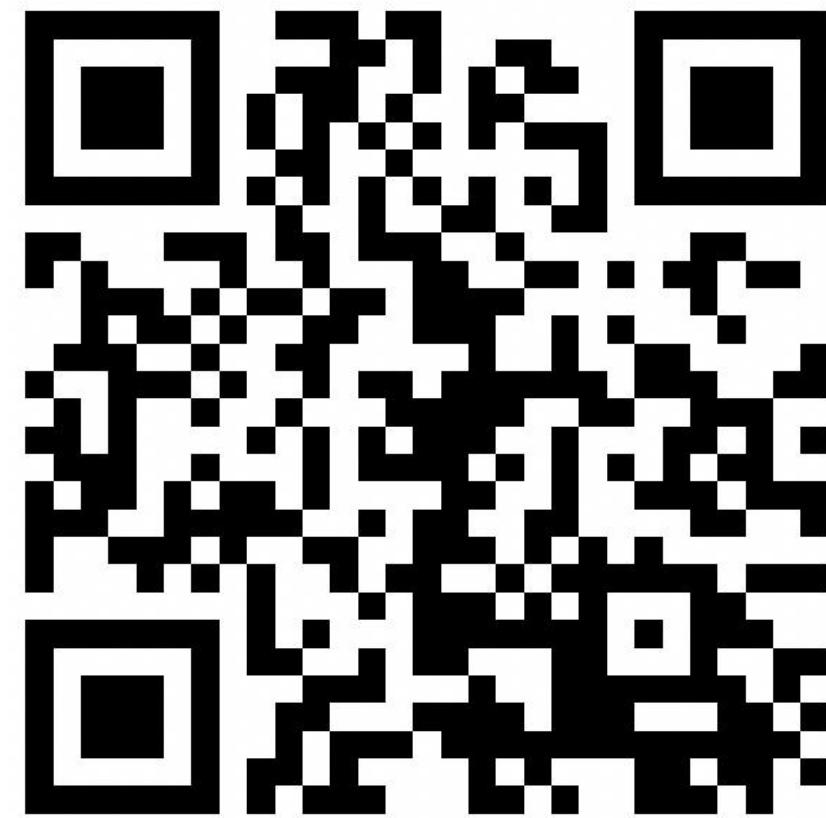
Acceptable Use - Slido Privacy - Cookie Settings

← → + 46 ...

Join at slido.com

#2417443





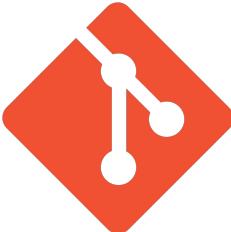
**GitHub - rgrzegorczyk/conferences: A place
for public content related to my
presentations at various conferences**

github.com

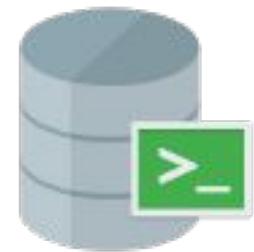
<https://github.com/rgrzegorczyk/conferences>

Demo

GIT + Fork (GUI Client)

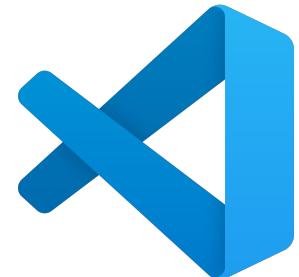


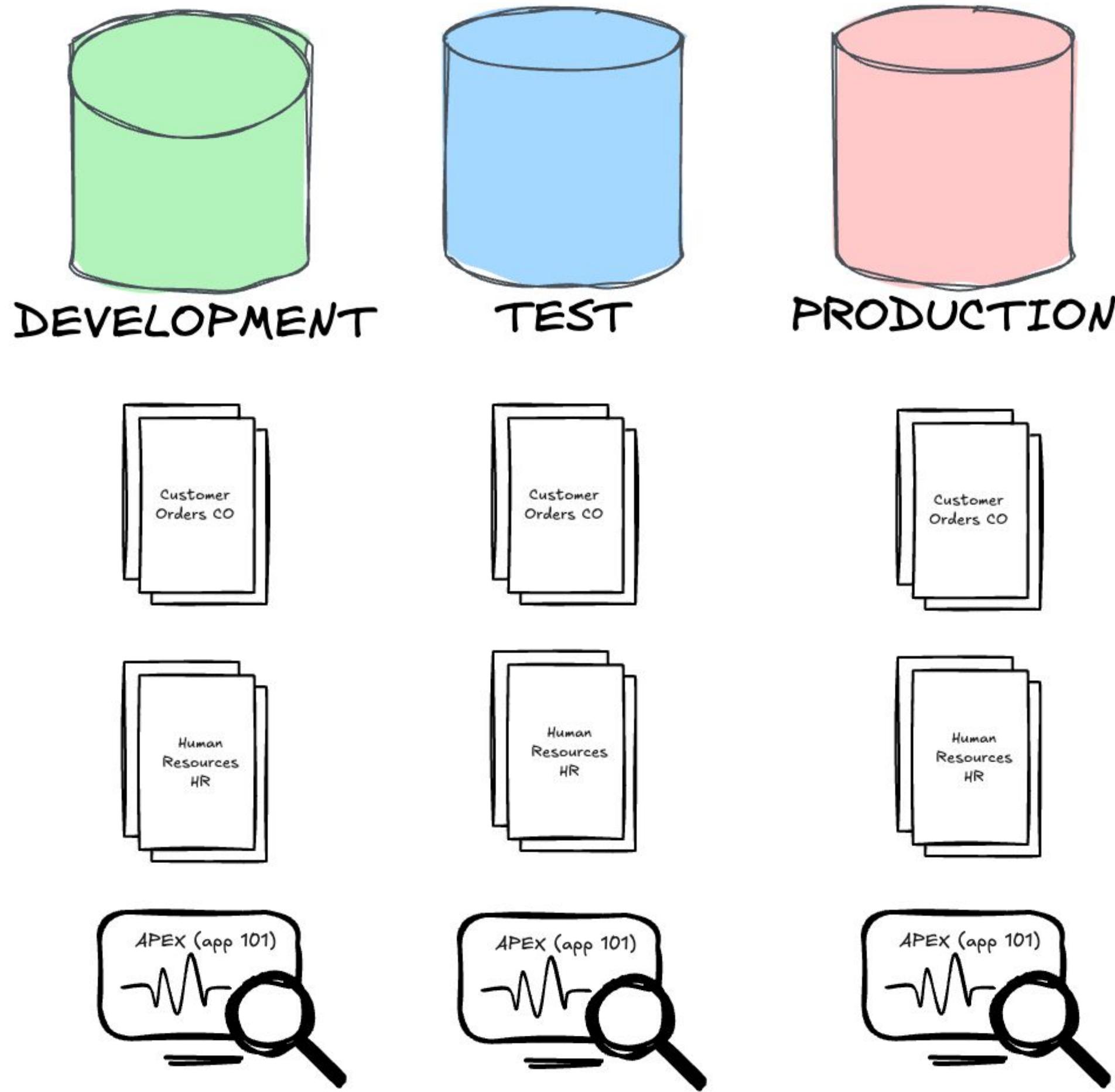
Oracle SQLcl



Get latest version 25.3.2 (14th Nov 2025)

Visual Studio Code

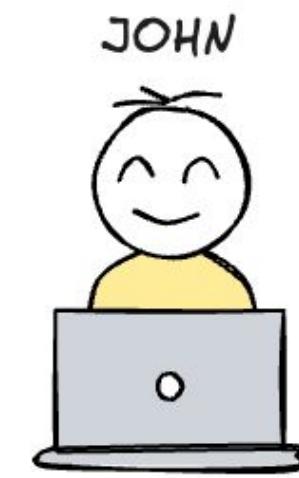




Database PROXY users:



RAFAL



JOHN



ADMIN

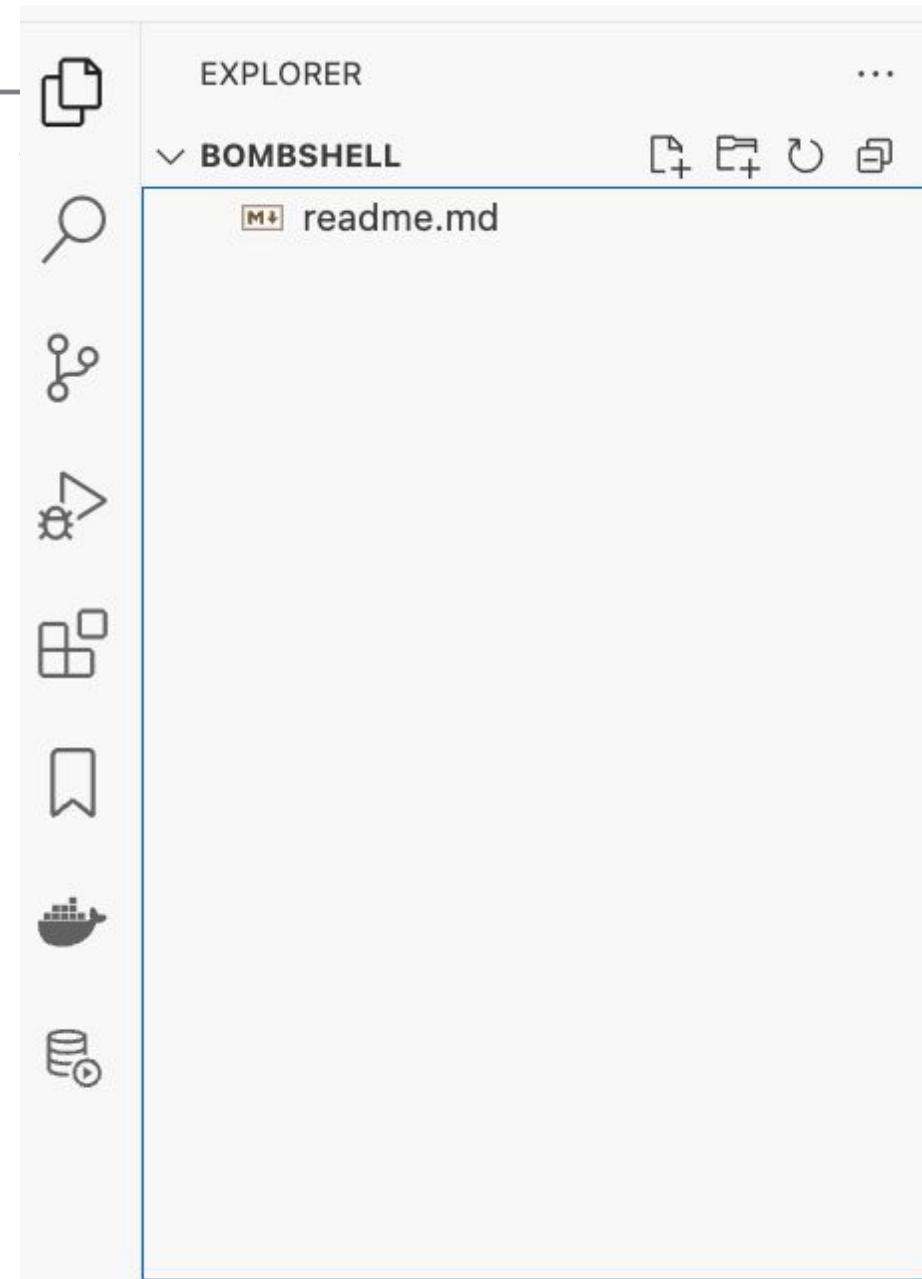
SQLcl named connections:

dev_hr_rafal
dev_co_rafal

dev_hr_admin
dev_co_admin

dev_admin
test_admin
prod_admin

Empty folder "Bombshell":



The screenshot shows the GitHub Desktop application. A new repository named 'bombshell' is initialized. The repository view shows three branches: 'main', 'dev' (which is checked out), and 'test'. A commit for 'Initialize repository' by Grzegorczyk Rafal is listed, along with a file tree containing 'readme.md'. The GitHub logo is visible in the top right corner.

- GIT repository in "bombshell" folder
- 3 branches - dev, test and main(for production)
- All branches are equal at this moment



1.Kick-off new project

Initialize a new project:

```
> sql -name dev_admin  
> project init -name bombshell -schemas HR,CO
```

Parameters:

- “name” - project name
- “schemas” - comma separated schemas to be tracked in project

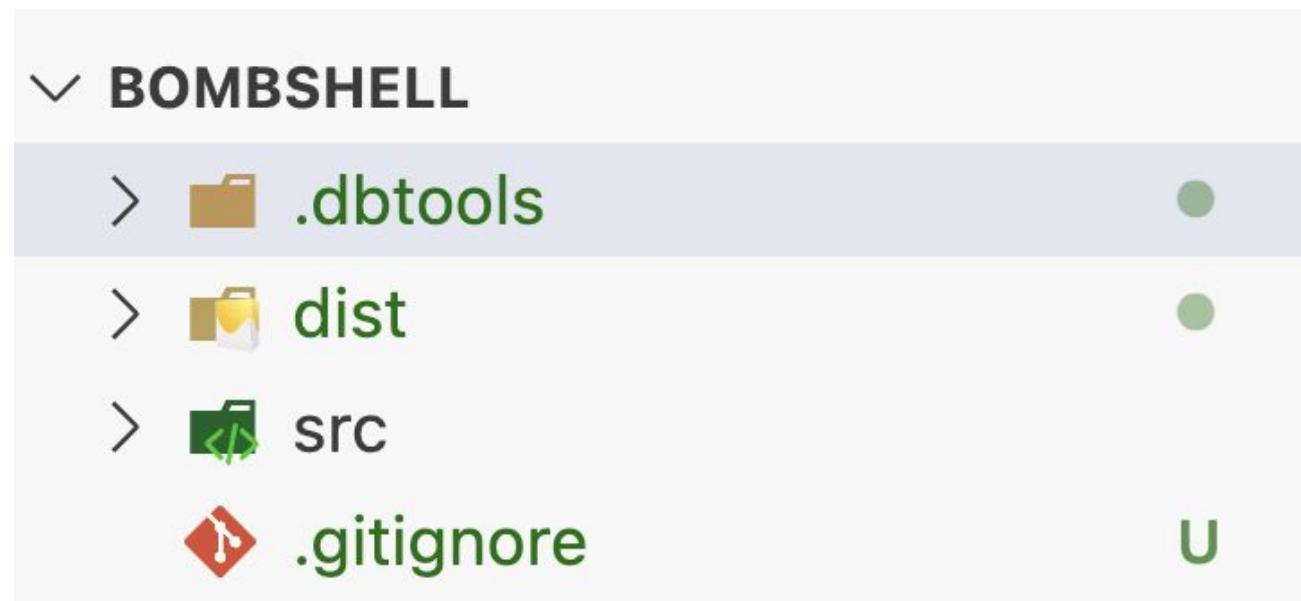


The screenshot shows a code editor interface with the following details:

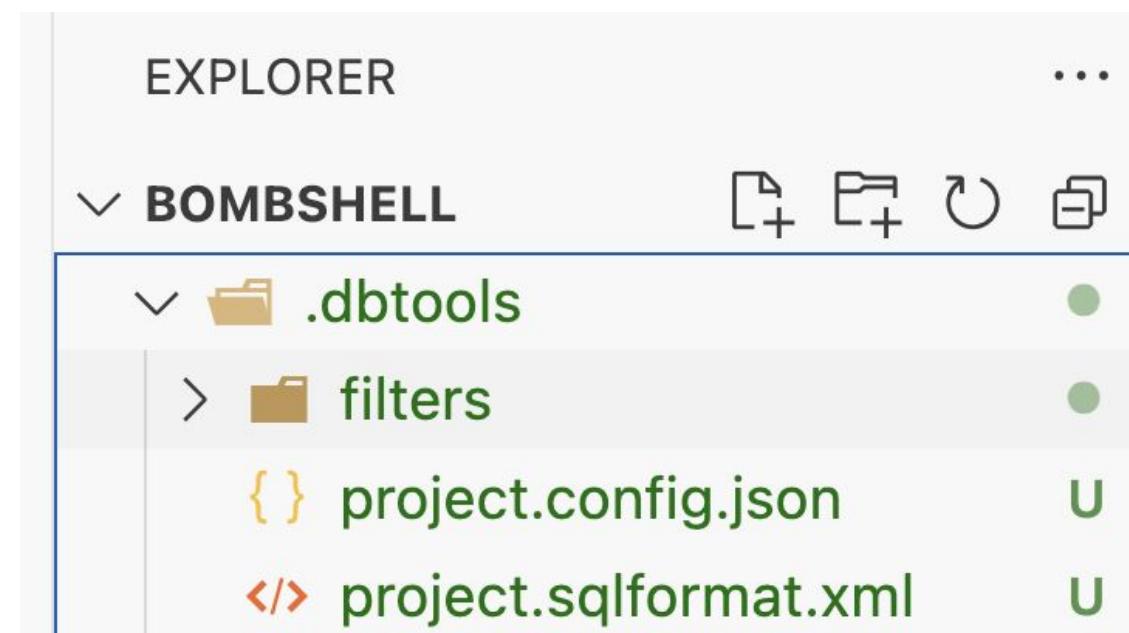
- Explorer View:** Shows a folder named "BOMBSHELL" containing a file named "readme.md".
- Terminal Tab:** Active tab, showing the following output:

```
o → bombshell git:(dev) ✘ sql -name dev_admin
SQLcl: Release 25.1 Production on Wed May 14 15:40:34 2025
Copyright (c) 1982, 2025, Oracle. All rights reserved.
```
- Bottom Status Bar:** Shows project names "dev*", "Git Graph", and "bombshell".
- Bottom Icons:** Includes icons for search, user profile, and notifications.

Several files and folders were created:



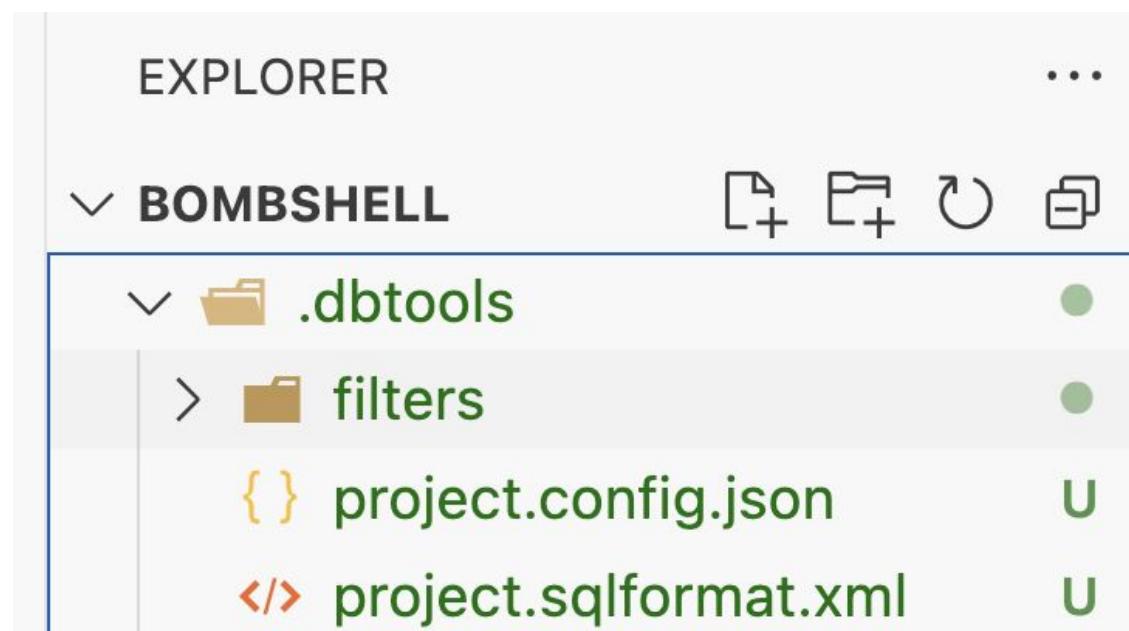
.dbtools - project configuration files



The screenshot shows the content of the "project.config.json" file. The JSON object defines a project named "bombshell" with specific database connection details and export settings.

```
{  
  "project": "bombshell",  
  "sqlcl": {  
    "connectionName": "dev_admin",  
    "version": "25.1.1.0"  
  },  
  "schemas": [ "HR", "CO" ],  
  "lastReleaseVersion": "",  
  "export": {  
    "format": {  
      "enable": true  
    },  
    "setTransform": {  
      "collationClause": "NEVER",  
      "constraints": true,  
      "constraintsAsAlter": true,  
      "emitSchema": true,  
      "emitStatistics": true  
    }  
  }  
}
```

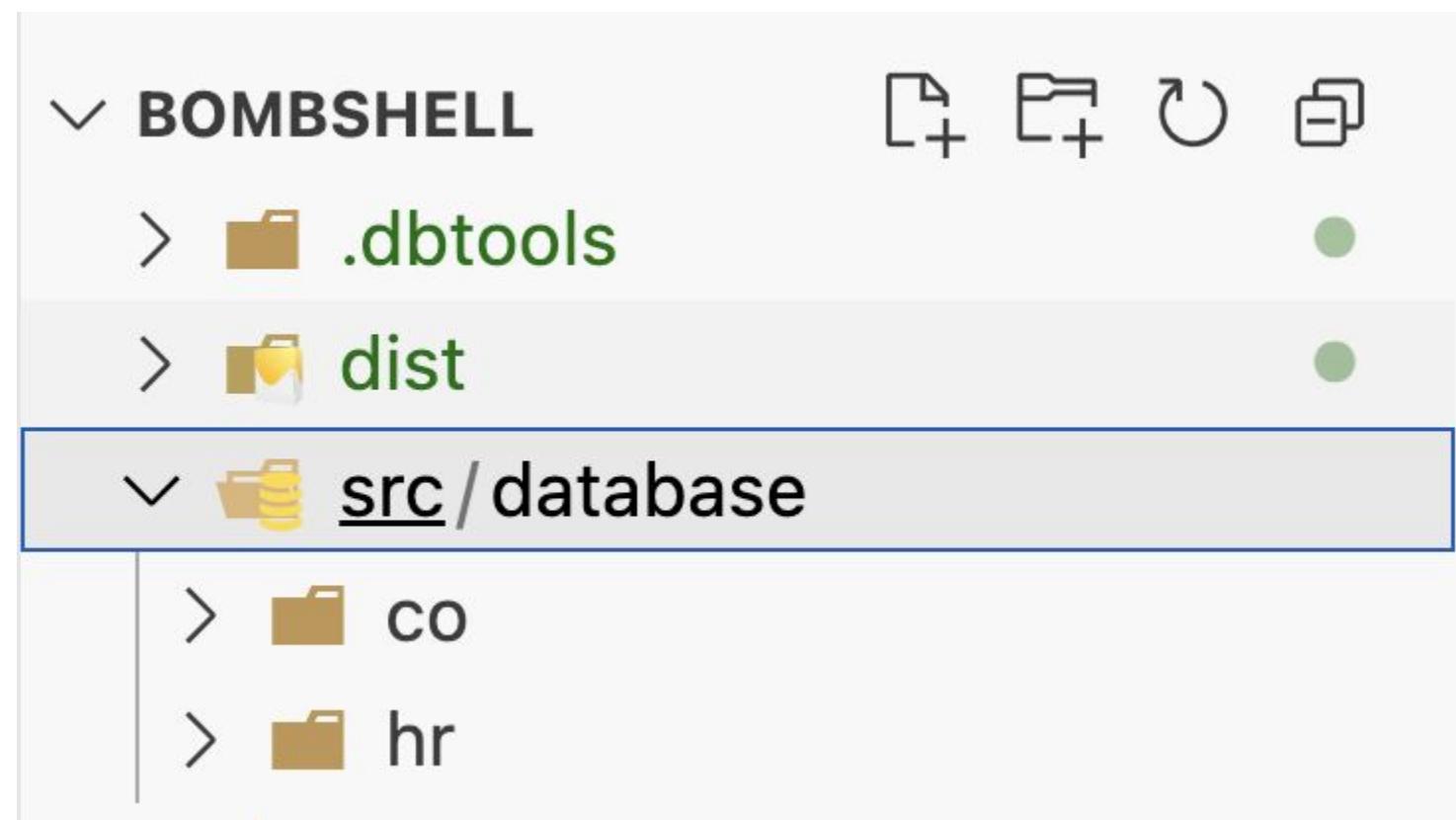
.dbtools - project configuration files



```
project.filters U X
.dbtools > filters > project.filters
1 /* Comma separated list of predicates to be fine-tuned for individual project.
2 Run
3   project export -debug
4   to double check the dictionary queries
5   with internal and custom filters applied.
6 */
7
8
9 -- Uncomment the line below if don't want to export grants:
10 -- export_type not in ('ALL_TAB_PRIVS','USER_SYS_PRIVS'),
11 /* Reference list of export_types from internal.fixed.filters
12   'ALL_OBJECTS',
13   'APEX_APPLICATIONS',
14   'ALL_COL_COMMENTS',
15   'ALL_TAB_COMMENTS',
16   'ALL_TAB_PRIVS',
17   --'USER_SYS_PRIVS',
18   'ALL_DEPENDENCIES',
19   'ALL_MVIEW_LOGS',
```

The code editor displays the 'project.filters' file. The content defines a list of predicates for fine-tuning projects. It includes instructions for running the project export with specific filters applied. Below this, it lists various export types such as 'ALL_OBJECTS', 'APEX_APPLICATIONS', etc.

.src - source files from exported schema(s)



- Export will be done in next steps

.dist - generated install files and releases

The screenshot shows a code editor interface with the following details:

- EXPLORER** sidebar:
 - BOMBSHELL folder
 - .dbtools
 - dist (selected)
 - install.sql
 - src/database
 - .gitignore
 - readme.md
- install.sql** file in the main editor area:

```
dist > install.sql
1 -- SQLcl uses the SQLCL engine for formatted sql changelog not the JDBC engine
2 -- By default, a project will not use substitution variables and allows blank
3 -- lines in sql statements.
4
5 set define off
6 set sqlblanklines on
7
8 -- Prechecks modifiable helper
9 -- Check running with SQLcl
10 -- Check minimum DB version
11 -- Check character set
```

2. Take a snapshot of

existing

Database & APEX code

(Proceed to step 4 for new applications)

Export ALL database objects

> project **export**

Questions? Ask below:



The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The title bar at the top has the text "bomshell". The left sidebar contains various icons for navigation and management, including a file icon, a search icon, a connection icon with a "10" notification, a preview icon, a folder icon with a "4" notification, a bookmark icon, a ship icon, a cylinder icon, an alert icon with a "1" notification, and a gear icon with a "1" notification.

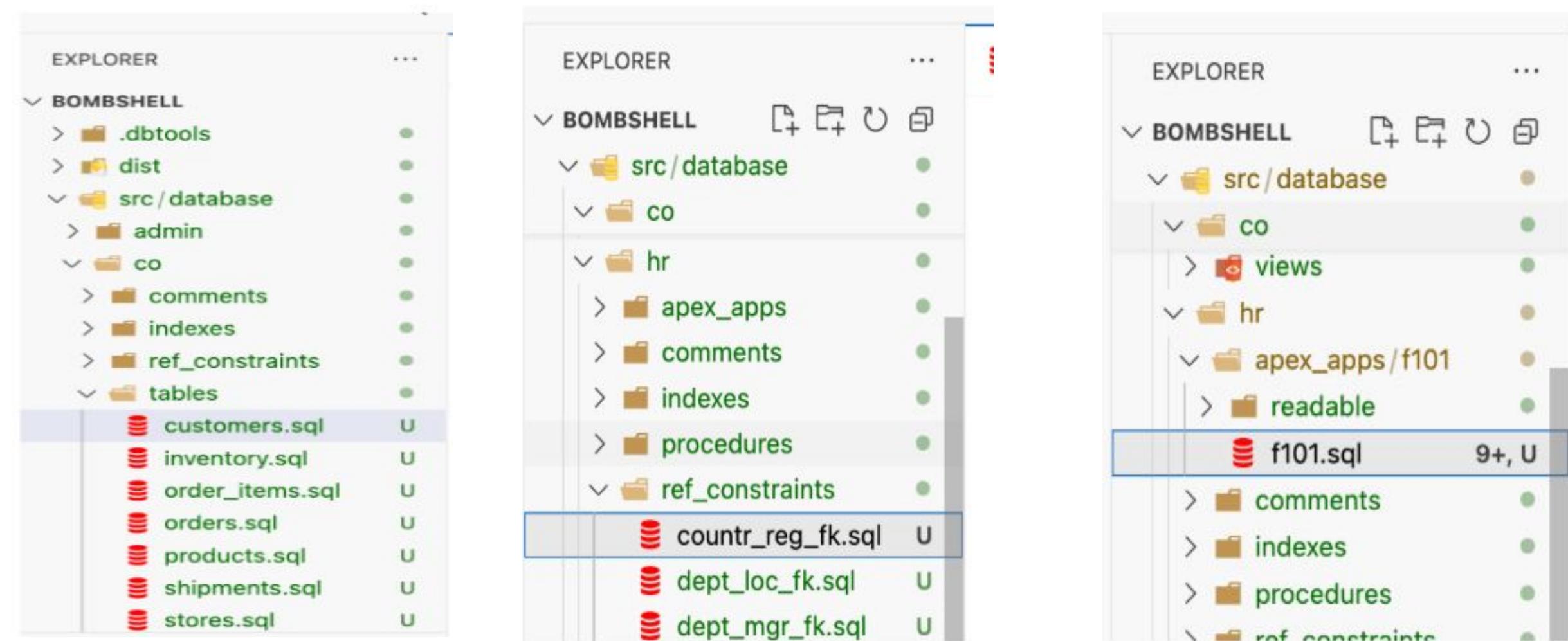
The main area displays the "EXPLORER" view for the project "BOMBSHELL". The "dist" folder is currently selected. Other visible files and folders include ".dbtools", "src", ".gitignore", and "readme.md".

The "TERMINAL" tab is active, showing the following output:

```
07:16:55 SQL> project export
The current connection DEVELOPMENT_HIGH ADMIN will be used for all operations
```

At the bottom, the status bar shows the path "viins | 1:14 | ADMIN | dev_admin", the version "1.1*", and the Git Graph extension status.

All selected objects CO, HR and APEX were exported



—
— Optionally export only selected objects:

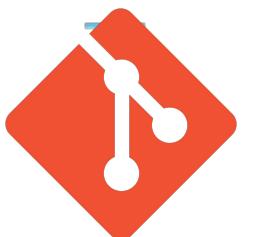
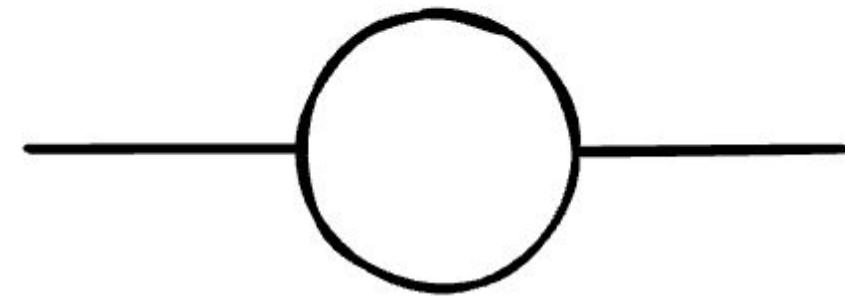
> project **export -objects** HR.COUNTRIES, CO.CUSTOMERS

> project **export -objects** APEX.101

3. Mark current state of
DB & APEX
[release 1.0]

Commit current state of DB & APEX [release 1.0]

git commit

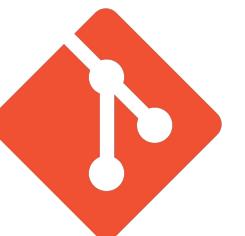
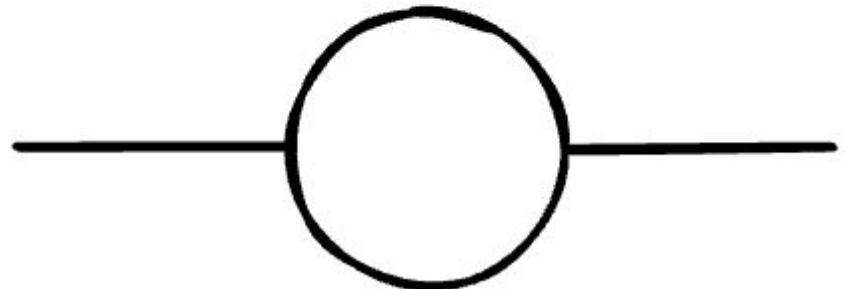




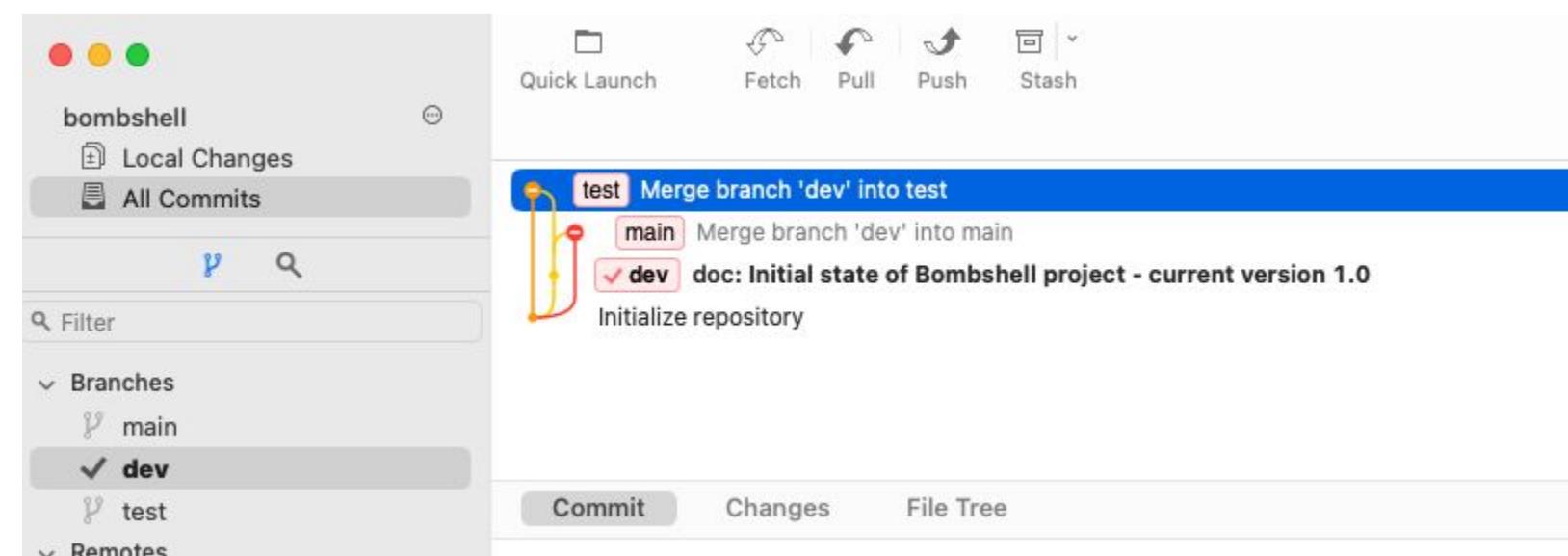
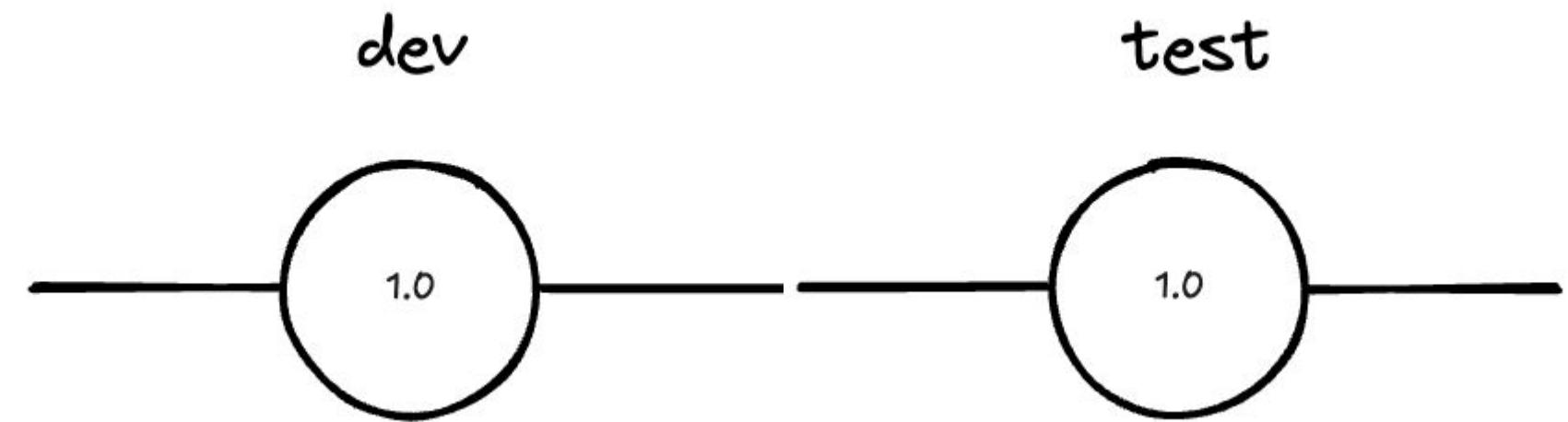
release 1.0

Current state of DB & APEX [release 1.0]

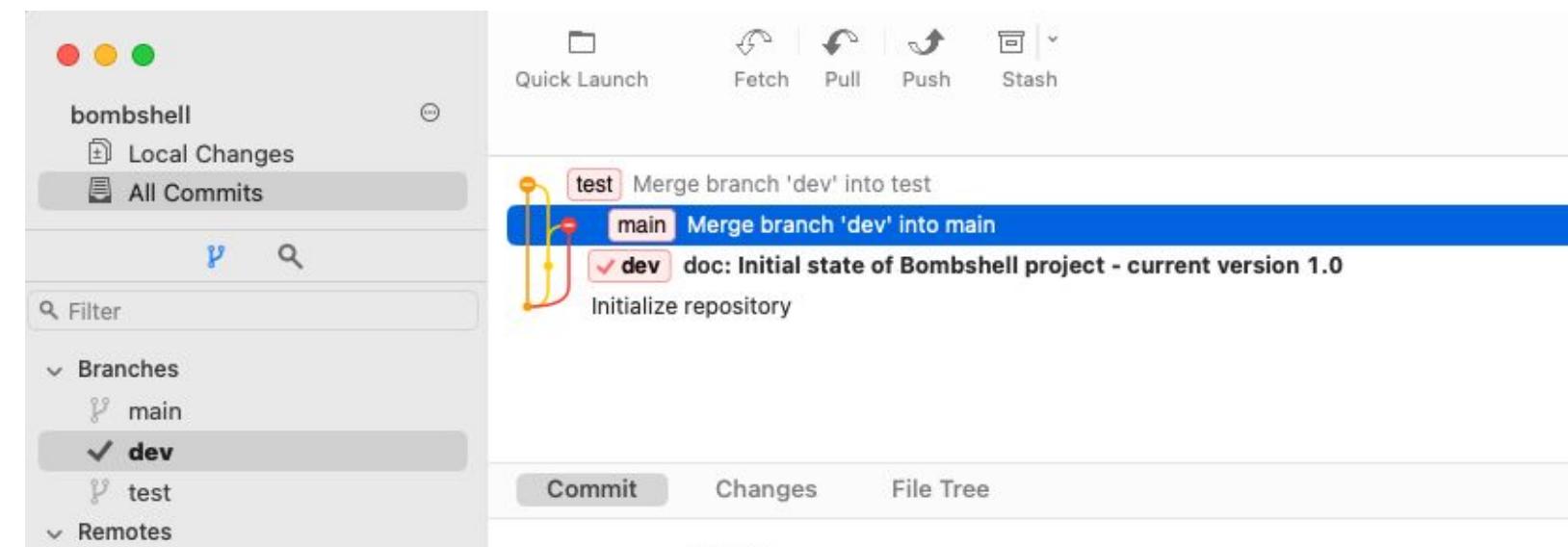
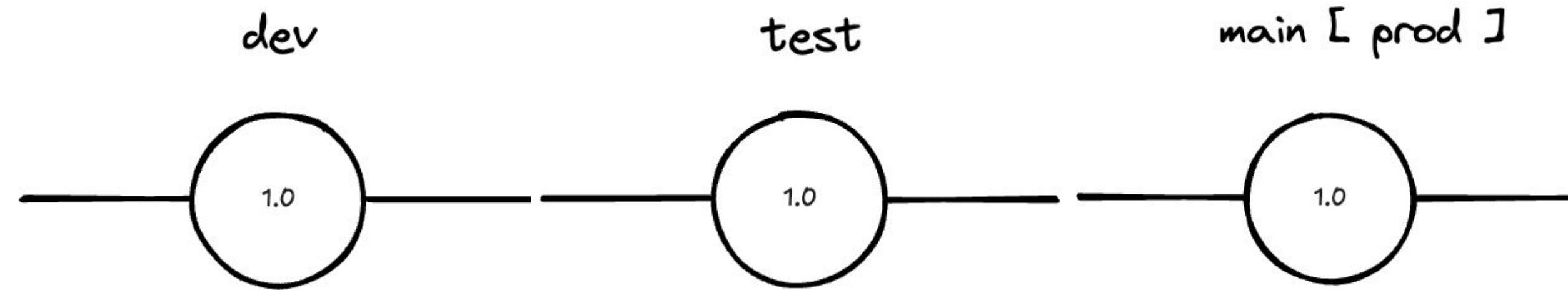
git commit



Merge dev into test



Merge dev into main [prod]

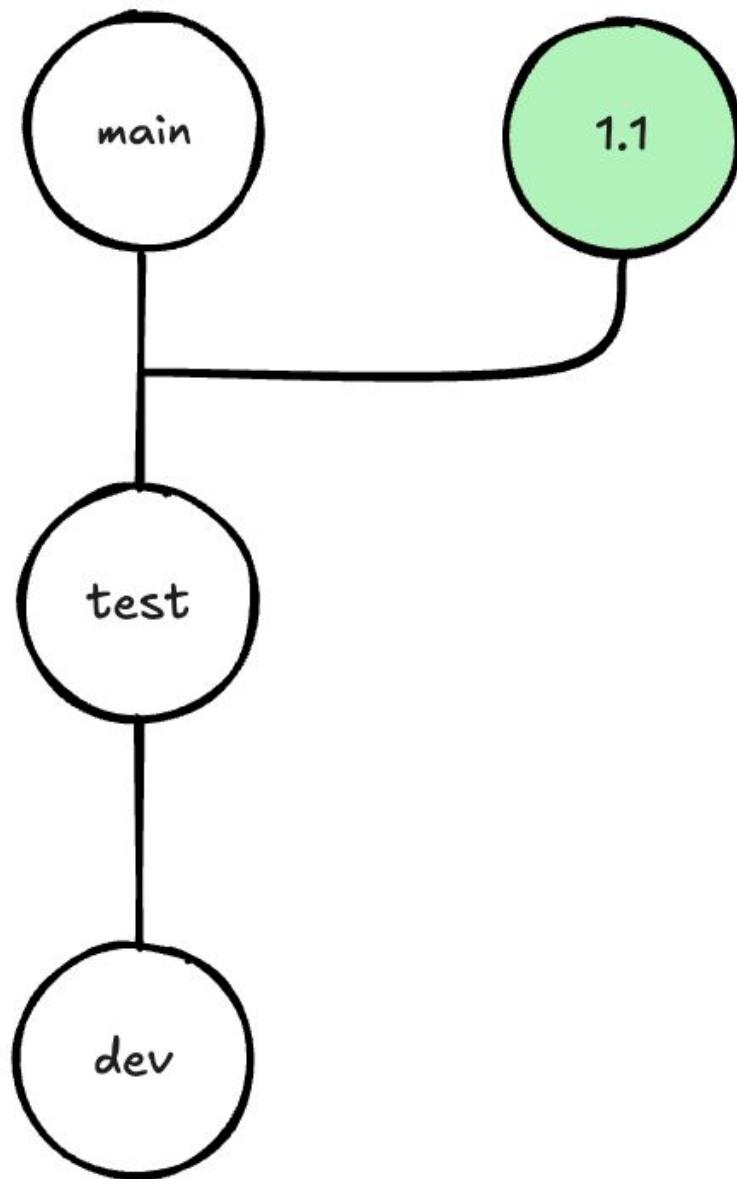


4.New development

[release 1.1]

Create new GIT branch [1.1]

1.0

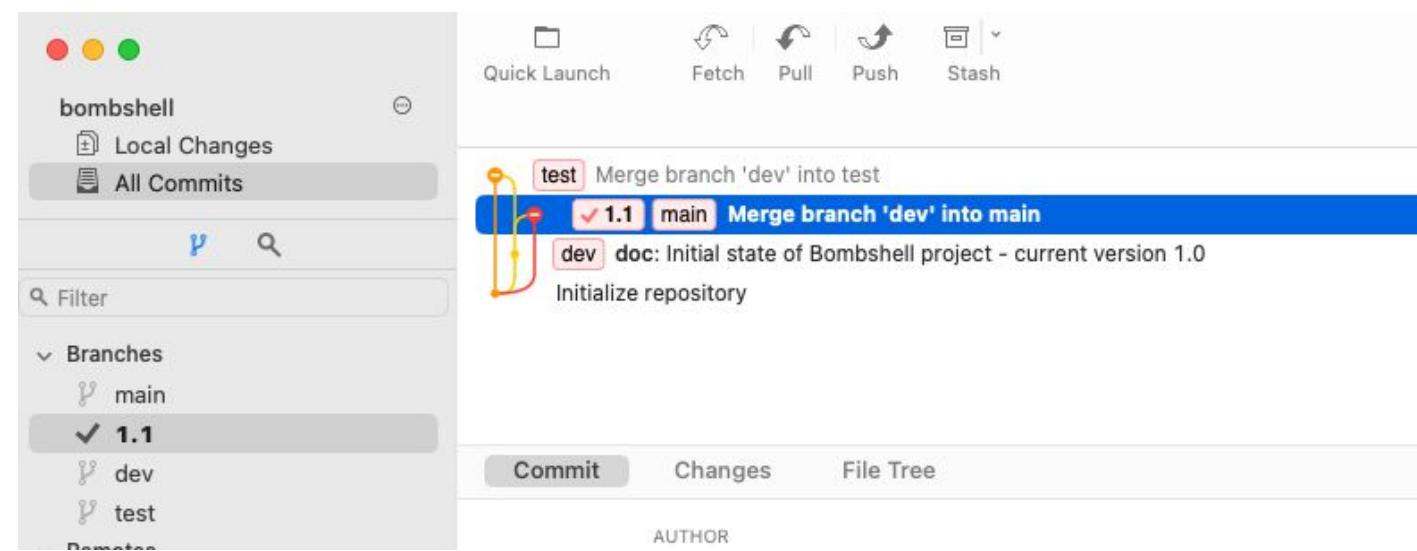
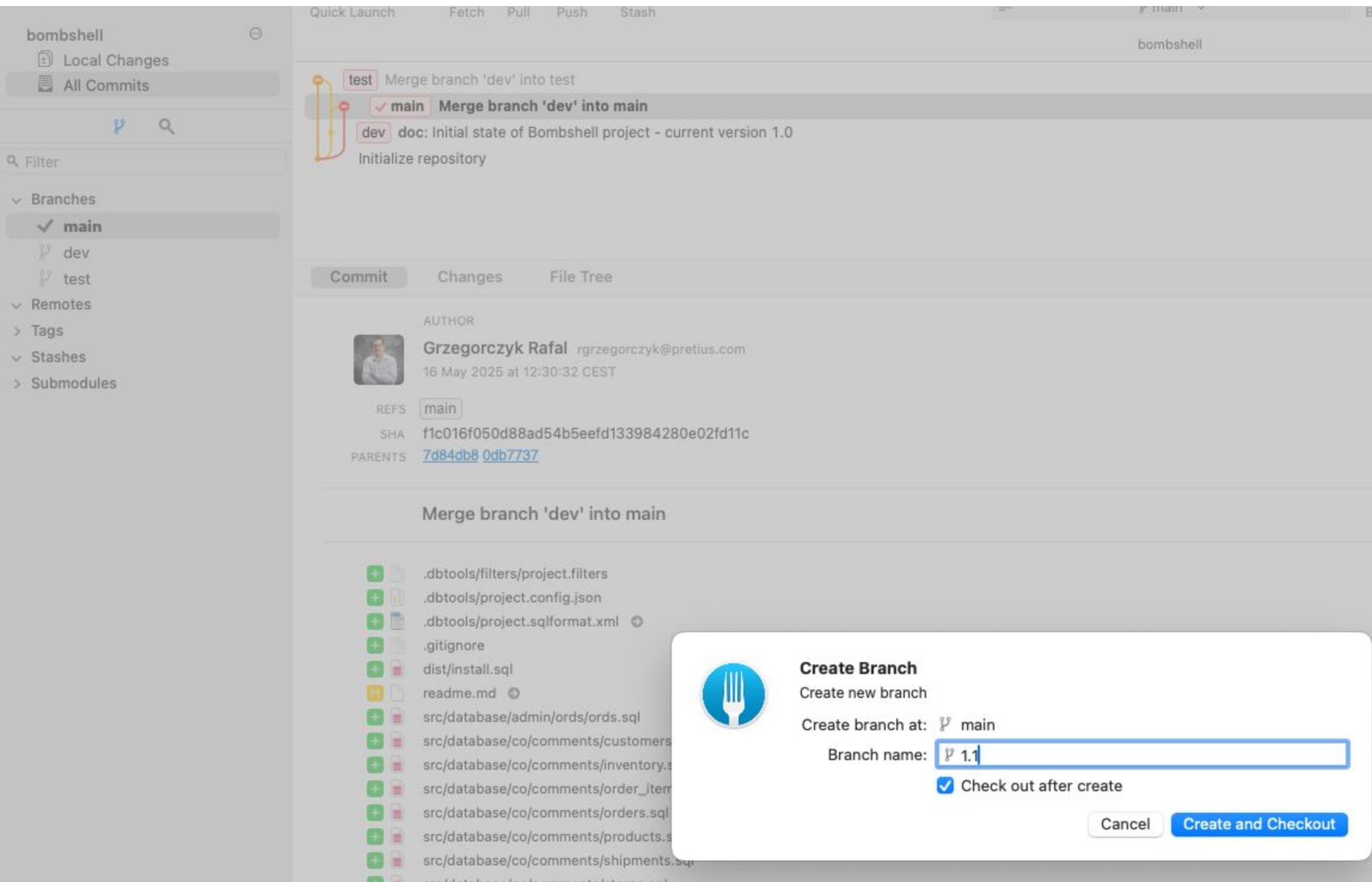


Questions? Ask below:



pretius

release 1.1

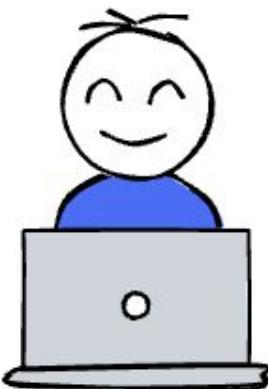


Developer RAFAL makes some changes on DEV

```
ALTER TABLE hr.employees ADD (
    phone_prefix VARCHAR2(10)
);
COMMENT ON COLUMN hr.employees.phone_prefix IS 'Phone number prefix';
```

```
CREATE OR REPLACE VIEW hr.v_emp_names_phones AS
SELECT e.first_name,
       e.last_name,
       e.phone_number,
       e.phone_prefix
FROM employees e;
```

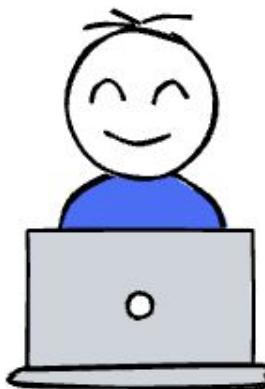
RAFAL



— RAHAL exports changes using SQLcl Project Export

```
> project export -objects HR.EMPLOYEES,HR.V_EMP_NAMES_PHONES
```

RAHAL



The screenshot shows the pretius IDE interface. On the left, there's a sidebar with icons and a connection tree:

- SQL DEVELOPER (1)
- SQL SNIPPETS
- CONNECTIONS (3):
 - Priv
 - BOMBSHELL
 - DEVELOPMENT (4): dev_admin, dev_co, dev_hr, dev_hr_john, dev_hr_rafal
- REPORTS

The main area is a code editor titled "bombshell" containing the following SQL script:

```
ALTER TABLE hr.employees ADD (
    phone_prefix VARCHAR2(10)
);
COMMENT ON COLUMN hr.employees.phone_prefix IS 'Phone number prefix';

CREATE OR REPLACE VIEW hr.v_emp_names_phones AS
SELECT e.first_name,
       e.last_name,
       e.phone_number,
       e.phone_prefix
  FROM employees e;
```

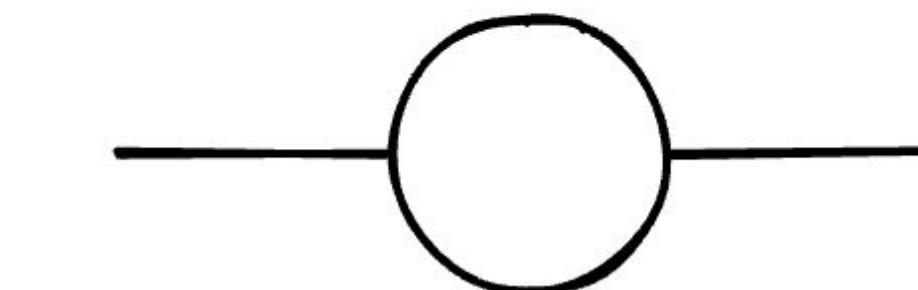
Below the code editor, there are tabs: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, SCRIPT OUTPUT (selected), SQL HISTORY, and TASK MONITOR.

In the bottom right corner, there's a user profile icon for "RAFAL" and some status indicators: 1 notification, 1 message, and 1 gear.

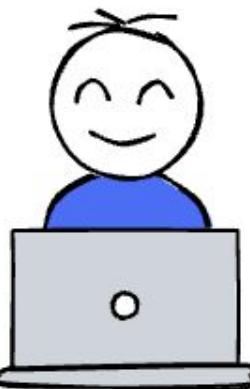
At the very bottom, there's a footer bar with various icons and text: 1.1*, 0 errors, 0 warnings, Git Graph, bombshell, Ln 1, Col 1 (290 selected), Spaces: 2, UTF-8, LF, PL/SQL, (Dedicated) dev_hr_rafal, and a bell icon.

RAFAL commits changes into GIT 1.1 branch

git commit



RAFAL



The screenshot shows a Git interface with three code files:

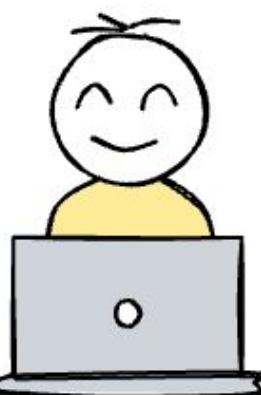
- employees.sql**: Adds comments for phone_prefix and salary.
- employees_tables.sql**: Creates the hr.employees table with columns manager_id, department_id, and phone_prefix.
- v_emp_names_phones.sql**: Creates a view hr.v_emp_names_phones.

The interface includes a sidebar titled "bombshell" showing "Local Changes" and "All Commits". The commit history for branch 1.1 is displayed, with the most recent commit being "1.1 feature 1.1 [RAFAL] - Added phone prefix to employees". Other commits shown include "test Merge branch 'dev' into test", "main Merge branch 'dev' into main", and "dev doc: Initial state of Bombshell project - current version 1.0".

Developer JOHN makes some changes on DEV

```
CREATE TABLE CO.APPLICATION_PARAMS  
  (ID NUMBER GENERATED ALWAYS AS IDENTITY NOT NULL,  
   NAME VARCHAR2(20) UNIQUE,  
   VALUE VARCHAR2(20),  
   CONSTRAINT APPLICATION_PARAMS_PK PRIMARY KEY (ID)  
 );  
COMMENT ON COLUMN CO.APPLICATION_PARAMS.ID IS 'Unique ID';  
COMMENT ON COLUMN CO.APPLICATION_PARAMS.NAME IS 'Parameter name';  
COMMENT ON COLUMN CO.APPLICATION_PARAMS.VALUE IS 'Parameter value';
```

JOHN

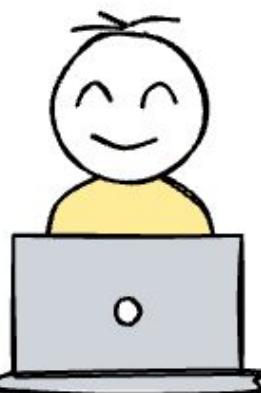


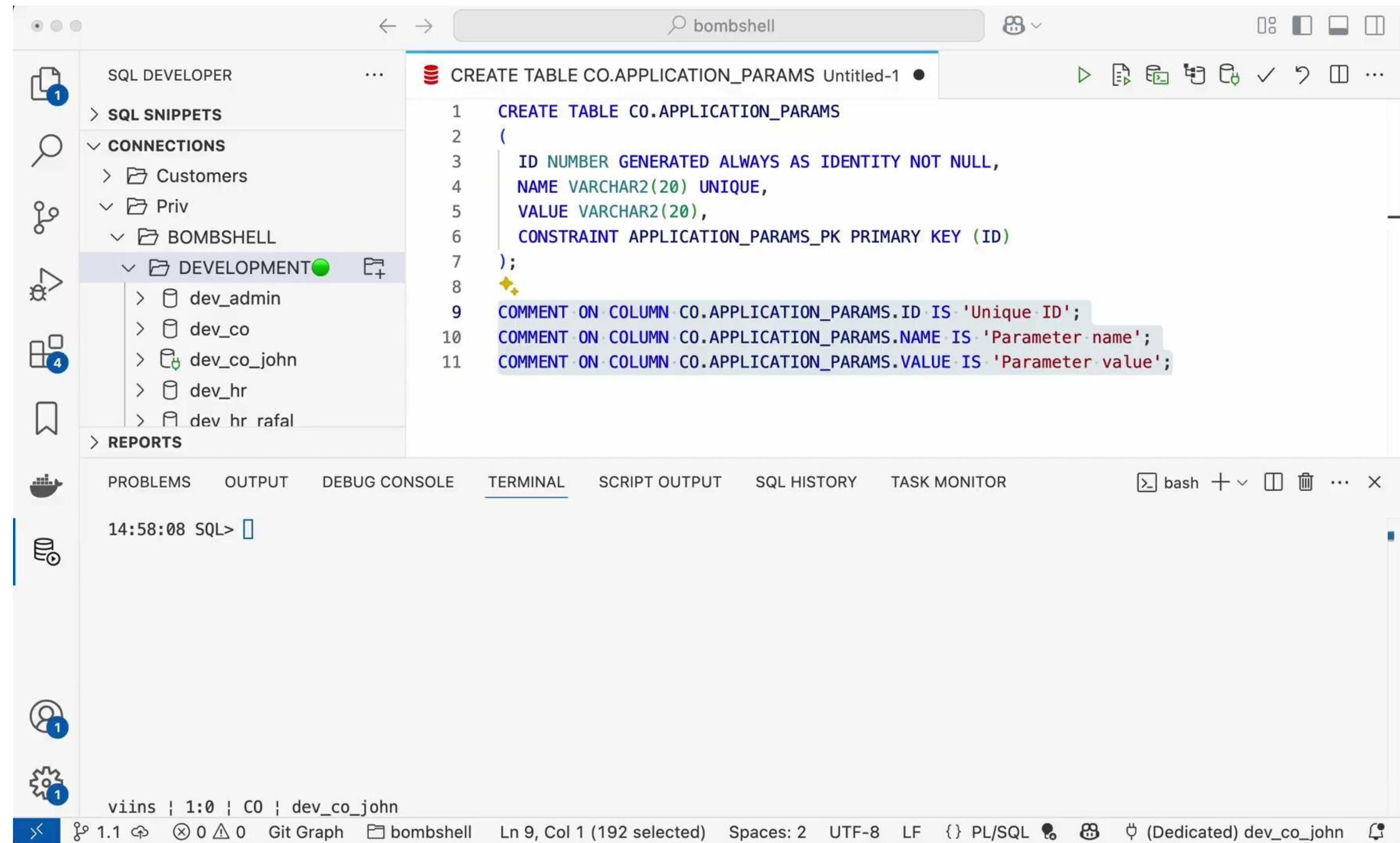
```
INSERT INTO CO.APPLICATION_PARAMS (NAME, VALUE) VALUES  
  ('support_email', 'hello@pretius.com');
```

— JOHN exports changes using SQLcl Project Export

```
> project export -objects CO.APPLICATION_PARAMS
```

JOHN





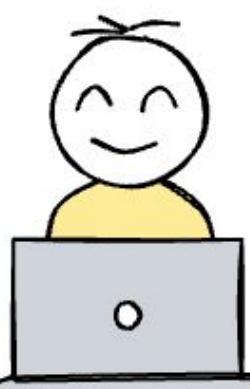
The screenshot shows the pretius IDE interface. On the left is a sidebar with various icons and sections: SQL DEVELOPER, SQL SNIPPETS, CONNECTIONS (Customers, Priv, BOMBSHELL, DEVELOPMENT), REPORTS, PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (selected), SCRIPT OUTPUT, SQL HISTORY, and TASK MONITOR. The main area is a code editor titled "CREATE TABLE CO.APPLICATION_PARAMS Untitled-1". The code is:

```
1 CREATE TABLE CO.APPLICATION_PARAMS
2 (
3     ID NUMBER GENERATED ALWAYS AS IDENTITY NOT NULL,
4     NAME VARCHAR2(20) UNIQUE,
5     VALUE VARCHAR2(20),
6     CONSTRAINT APPLICATION_PARAMS_PK PRIMARY KEY (ID)
7 );
8
9 COMMENT ON COLUMN CO.APPLICATION_PARAMS.ID IS 'Unique ID';
10 COMMENT ON COLUMN CO.APPLICATION_PARAMS.NAME IS 'Parameter name';
11 COMMENT ON COLUMN CO.APPLICATION_PARAMS.VALUE IS 'Parameter value';
```

The bottom status bar shows: JOHN, viins | 1:0 | CO | dev_co_john, 1.1, Git Graph, bombshell, Ln 9, Col 1 (192 selected), Spaces: 2, UTF-8, LF, PL/SQL, (Dedicated) dev_co_john.

JOHN adds new IR in APEX app 101

JOHN



Bombshell App

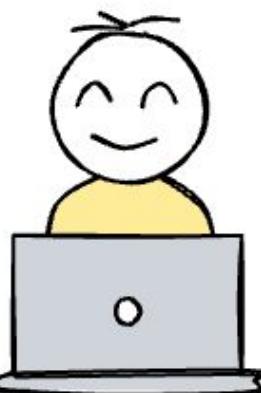
John's Report

Employee Id	First Name	Last Name	Email	Phone Number	Hire Date	Job Id	Salary	Commission Pct	Manager Id	Department Id	Phone Prefix
100	Steven	King	SKING	1.515.555.0100	6/17/2013	AD_PRES	24000			90	
101	Neena	Yang	NYANG	1.515.555.0101	9/21/2015	AD_VP	17000		100	90	
102	Lex	Garcia	LGARCIA	1.515.555.0102	1/13/2011	AD_VP	17000		100	90	
103	Alexander	James	AJAMES	1.590.555.0103	1/3/2016	IT_PROG	9000		102	60	
104	Bruce	Miller	BMILLER	1.590.555.0104	5/21/2017	IT_PROG	6000		103	60	
105	David	Williams	DWILLIAMS	1.590.555.0105	6/25/2015	IT_PROG	4800		103	60	
106	Valli	Jackson	VJACKSON	1.590.555.0106	2/5/2016	IT_PROG	4800		103	60	
107	Diana	Nguyen	DNGUYEN	1.590.555.0107	2/7/2017	IT_PROG	4200		103	60	
108	Nancy	Gruenberg	NGRUENBE	1.515.555.0108	8/17/2012	FI_MGR	12008		101	100	
109	Daniel	Faviet	DFAVIET	1.515.555.0109	8/16/2012	FI_ACCOUNT	9000		108	100	

JOHN also adds custom change on DEV

```
INSERT INTO CO.APPLICATION_PARAMS (NAME, VALUE) VALUES ('support_email', 'hello@pretius.com');
```

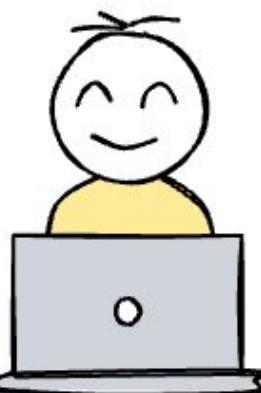
JOHN



JOHN adds custom file to SQLcl Project's stage

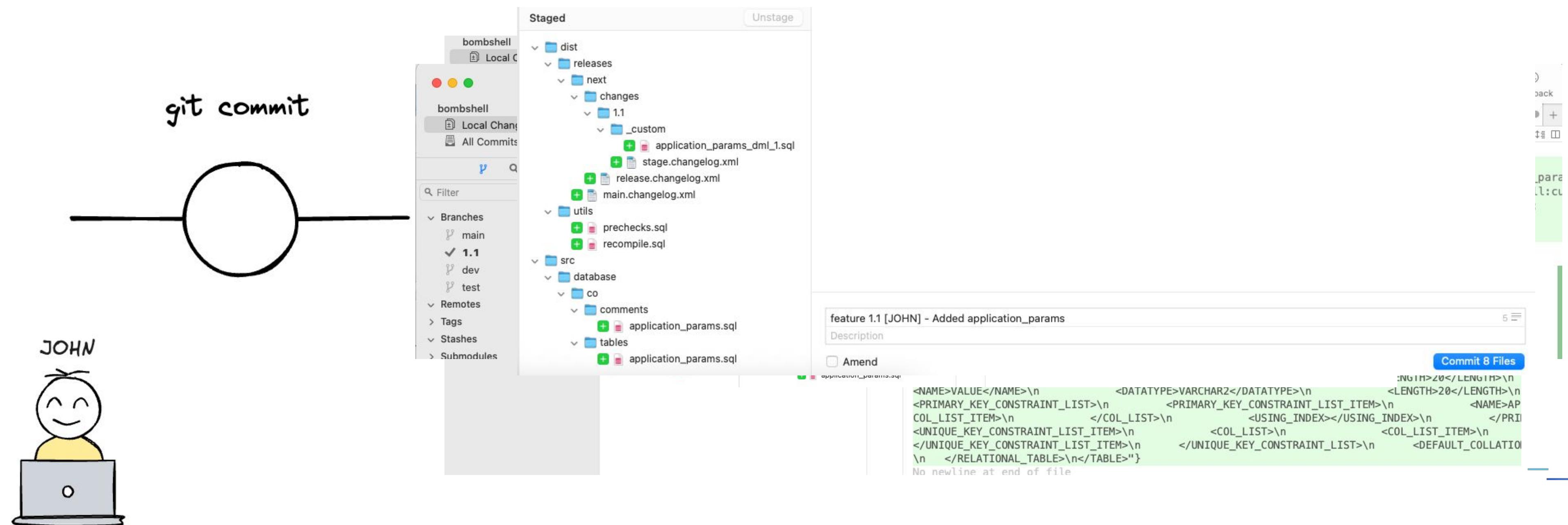
```
> project stage add-custom -file-name application_params_dml_1
```

JOHN



The screenshot shows the pretius IDE interface. On the left is a sidebar with icons for SQL DEVELOPER (1), SQL SNIPPETS, CONNECTIONS (2), REPORTS (4), and a database icon. The CONNECTIONS section is expanded, showing a 'Priv' folder containing a 'BOMBSHELL' folder which has a 'DEVELOPMENT' folder. Inside 'DEVELOPMENT' are connections: dev_admin (yellow), dev_co (yellow), dev_co_john (green), dev_hr (yellow), and dev_hr_rafal (yellow). Below 'DEVELOPMENT' is a 'PRODUCTION' folder with a red exclamation mark. At the top right is a search bar with the text 'bombshell'. The main area displays a script titled 'INSERT INTO CO.APPLICATION_PARAMS (NAME, Untitled-1)'. The first line of the script is: `1 INSERT INTO CO.APPLICATION_PARAMS (NAME, VALUE) VALUES ('support_email', 'hello@pretius.co'`. Below the script are several small icons: a green triangle, a green file, a green folder, a green gear, a green checkmark, a green arrow, a green square, and a green circle. The bottom navigation bar includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, SCRIPT OUTPUT (which is selected), SQL HISTORY, and TASK MONITOR. The status bar at the bottom shows version 1.1*, 0 errors, 0 warnings, a Git Graph, a folder labeled 'bombshell', a magnifying glass icon, 'Ln 1, Col 95', 'Spaces: 2', 'UTF-8', 'LF', a PL/SQL icon, a dedicated icon, and '(Dedicated) dev_co_john'.

JOHN commits changes into GIT 1.1 branch



Liquibase syntax is fully handled by SQLcl Project

The image shows three screenshots of the SQLcl interface, each displaying a different XML changelog file:

- Top Screenshot:** Shows the contents of `dist/releases/next/changes/1.1/stage.changelog.xml`. It includes an XML declaration, a databaseChangeLog element with namespaces and schemaLocation, and an include statement for `_custom/application_params_dml_1.sql`.
- Middle Screenshot:** Shows the contents of `dist/releases/next/release.changelog.xml`. It includes an XML declaration, a databaseChangeLog element with namespaces and schemaLocation, an include statement for `stage.changelog.xml`, and a comment `<!--CHANGES-->`.
- Bottom Screenshot:** Shows the contents of `dist/releases/main.changelog.xml`. It includes an XML declaration, a databaseChangeLog element with namespaces and schemaLocation, an include statement for `next/release.changelog.xml`, and a note "No newline at end of file".

The left sidebar of each screenshot shows a tree view of the project structure under the "Unstaged" tab, including `dist`, `releases`, `next`, `changes`, `1.1`, and `_custom` directories, along with files `application_params_dml_1.sql` and `stage.changelog.xml`.

5. Prepare for deployment of release 1.1

[DEV to TEST]

* Similar steps for deployment to any other environment

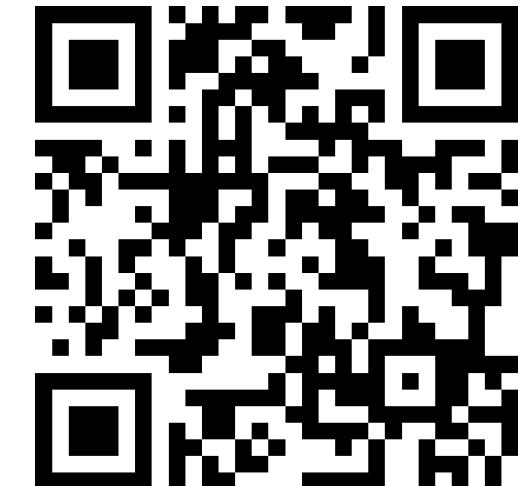
<<time>>

Export ALL database objects

```
> sql -name dev_admin  
> project export
```

- This step ensures you won't miss any changes that were made in DEV environment

Questions? Ask below:



07:14:35 SQL> **project export**

The current connection DEVELOPMENT_HIGH ADMIN will be used for all operations

*** TABLES ***

*** PROCEDURES ***

*** SEQUENCES ***

*** VIEWS ***

*** INDEXES ***

*** TRIGGERS ***

*** COMMENTS ***

*** REF_CONSTRAINTS ***

*** APEX_APPLICATIONS ***

Exporting Workspace BOMBSHELL - application 101:Bombshell App

<u>APEX_APPLICATION</u>	1
COMMENT	15
INDEX	18
PROCEDURE	2
REF_CONSTRAINT	19
SEQUENCE	3
TABLE	15
TRIGGER	2
VIEW	6

Exported 81 objects

Elapsed 74 sec

07:16:06 SQL> ■





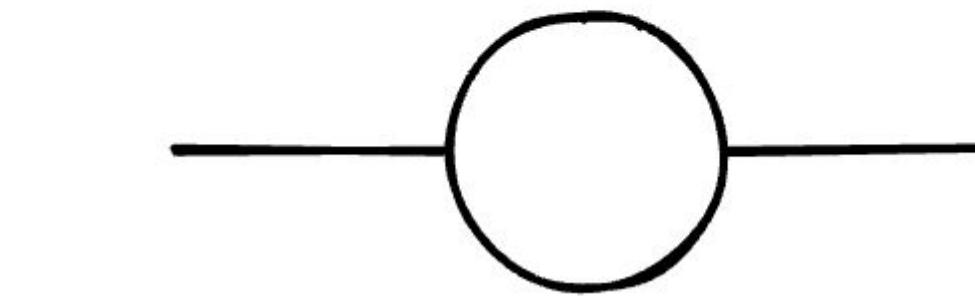
Unstaged Stage

src/database/hr/apex_apps/f101/readable/application/pages/p00001.yaml

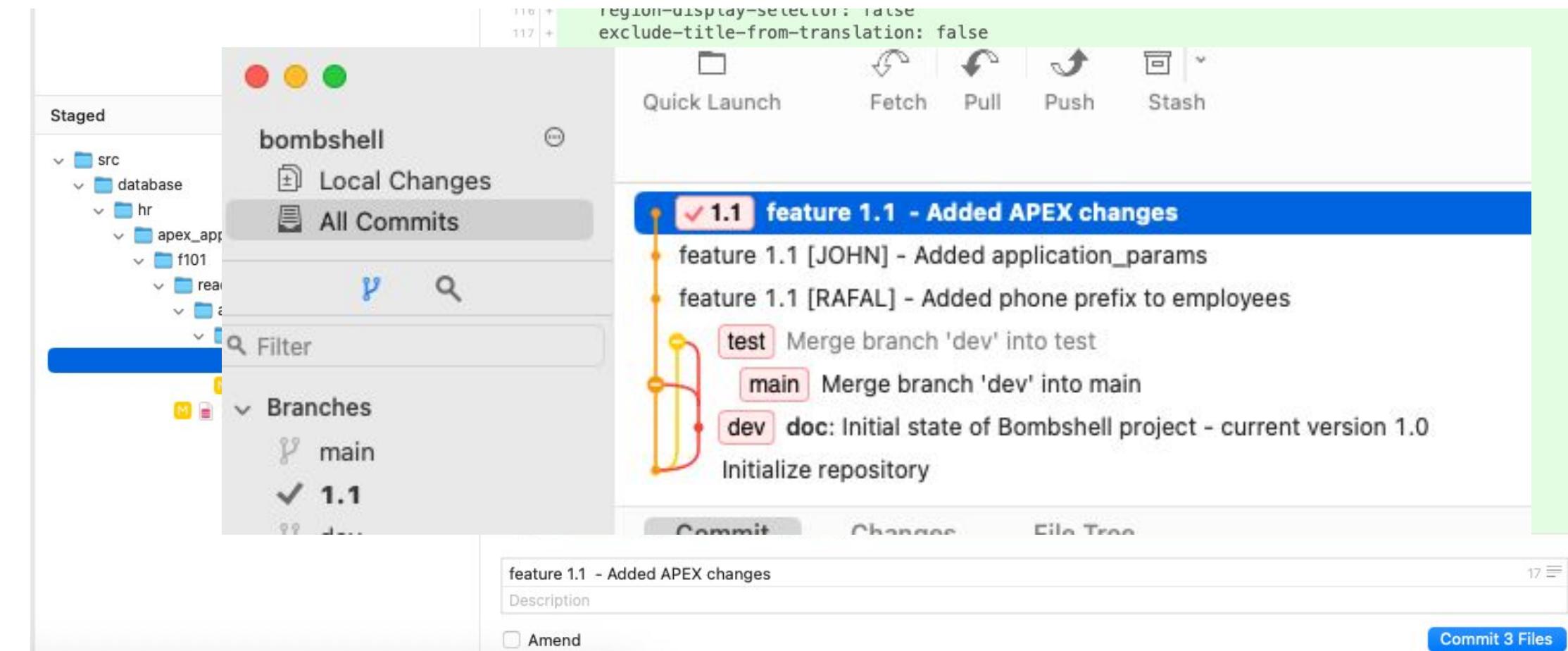
```
@@ -1,80 +80,646 @@ regions:  
1 80  
81 + # ===== Region: John's Report =====  
82 + id: 15709240491311201  
83 + identification:  
84 + name: John's Report  
85 + type: Interactive Report  
86 +  
87 + source:  
88 + location: Local Database  
89 + type: Table / View  
90 + table-owner: Parsing Schema  
91 + table-name: EMPLOYEES  
92 + include-rowid-column: false  
93 +  
94 + layout:  
95 + sequence: 10  
96 + parent-region: No Parent  
97 + slot: BODY  
98 + start-new-layout: false  
99 + start-new-row: true  
100 + column: Automatic  
101 + new-column: true  
102 + column-span: Automatic  
103 +  
104 + appearance:  
105 + template: Standard  
106 + template-options:  
107 + - '#DEFAULT#'  
108 + - t-Region--scrollBody  
109 + render-components: Above Content  
110 +  
111 + accessibility:  
112 + use-landmark: true  
113 + landmark-type: Template Default  
114 +  
115 + advanced:
```

ADMIN commits extra changes into GIT 1.1 branch

git commit



ADMIN



Project Stage

- Automatically **generates Liquibase changelogs/changesets** from all source and custom SQL files.
- Compares the current branch **(1.1)** with the target branch **(test)** to create changelogs that apply the necessary database updates.
- Organizes the output into a well-structured folders and files in a **/dist/** directory.



Project Stage

> project **stage -branch-name** test

Parameters:

- “-branch-name” - (optional) branch name for comparison.



The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. On the left is the Explorer sidebar, which displays a project structure under the 'BOMBSHELL' folder. The structure includes '.dbtools', 'dist', 'src', '.gitignore', and 'readme.md'. Below the Explorer is the Outline view, which is currently collapsed. The main workspace contains a terminal window with the following content:

```
16:38:24 SQL> project stage -branch-name test
```

The status bar at the bottom of the screen shows the following information:

- User: viins
- Time: 1:31
- Role: ADMIN
- Session: dev_admin

At the bottom right of the status bar, there are icons for search, file operations, and notifications.

Project Stage

- Review generated files and change order of custom changelogs if required



The screenshot shows a file explorer interface with the following structure:

- EXPLORER**:
 - BOMBSHELL**:
 - .dbtools
 - dist
 - releases
 - apex
 - next
 - changes/1.1
 - _custom
 - co
 - hr

The current file being edited is `stage.changelog.xml`, which contains the following XML code:

```
dist > releases > next > changes > 1.1 > stage.changelog.xml
1  <?xml version="1.0" encoding="UTF-8"?>
2  <databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3  xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
4  http://www.liquibase.org/xml/ns/dbchangelog-dbchangelog-4.3.xsd">
5
6  <include file="_custom/application_params_dml_1.sql" relativeToChangelogFile="true"/>
7  <include file="hr/tables/employees.sql" relativeToChangelogFile="true"/>
8  <include file="hr/views/v_emp_names_phones.sql" relativeToChangelogFile="true"/>
9  <include file="co/tables/application_params.sql" relativeToChangelogFile="true"/>
10 <include file="co/comments/application_params.sql" relativeToChangelogFile="true"/>
11 <include file="hr/comments/employees.sql" relativeToChangelogFile="true"/>
12
13 </databaseChangeLog>
```

A blue box highlights the line `<include file="_custom/application_params_dml_1.sql" relativeToChangelogFile="true"/>`. A green vertical bar is positioned to the left of the line numbers, and a dotted blue arrow points from the bottom right towards the highlighted line.

EXPLORER

- BOMBSHELL
 - .dbtools
 - dist
 - releases
 - apex
 - next
 - changes/1.1
 - _custom
 - co
 - hr

stage.changelog.xml M

```

<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
    xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
        http://www.liquibase.org/xml/ns/dbchangelog-4.3.xsd">
    <include file="hr/tables/employees.sql" relativeToChangelogFile="true"/>
    <include file="hr/views/v_emp_names_phones.sql" relativeToChangelogFile="true"/>
    <include file="co/tables/application_params.sql" relativeToChangelogFile="true"/>
    <include file="co/comments/application_params.sql" relativeToChangelogFile="true"/>
    <include file="hr/comments/employees.sql" relativeToChangelogFile="true"/>
    <include file="_custom/application_params_dml_1.sql" relativeToChangelogFile="true"/>
</databaseChangeLog>

```

release.changelog.xml M

main.changelog.xml M

Project Stage

- Review generated files and change order of custom changelogs if required

EXPLORER

- BOMBSHELL
 - .dbtools
 - dist
 - releases
 - apex
 - next
 - changes/1.1
 - _custom
 - co
 - hr

stage.changelog.xml M

```

dist > releases > next > changes > 1.1 > stage.changelog.xml
<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
    xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
        http://www.liquibase.org/xml/ns/dbchangelog-4.3.xsd">
    <include file="hr/tables/employees.sql" relativeToChangelogFile="true"/>
    <include file="hr/views/v_emp_names_phones.sql" relativeToChangelogFile="true"/>
    <include file="co/tables/application_params.sql" relativeToChangelogFile="true"/>
    <include file="co/comments/application_params.sql" relativeToChangelogFile="true"/>
    <include file="hr/comments/employees.sql" relativeToChangelogFile="true"/>
    <include file="_custom/application_params_dml_1.sql" relativeToChangelogFile="true"/>
</databaseChangeLog>

```

release.changelog.xml M

main.changelog.xml M



Project Stage

- Review generated files and proposed order of execution



The image shows a screenshot of a software interface, likely Liquibase, with the following details:

- EXPLORER** sidebar:
 - / BOMBSHELL
 - > .dbtools
 - < dist
 - < releases
 - > apex
 - < next
 - < changes/1.1
 - > _custom
 - > co
 - > hr
 - < stage.changelog.xml
 - < release.changelog.xml
 - < main.chanaeloa.xml
 - stage.changelog.xml** editor tab:

```
bomshell > dist > releases > 1.1 > changes > 1.1 > stage.changelog.xml
1  <?xml version="1.0" encoding="UTF-8"?>
2  <databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3  |           xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
4  |           xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
|           http://www.liquibase.org/xml/ns/dbchangelog-dbchangelog-4.3.xsd">
5
6  <include file="hr/tables/employees.sql" relativeToChangelogFile="true"/>
7  <include file="hr/views/v_emp_names_phones.sql" relativeToChangelogFile="true"/>
8  <include file="co/tables/application_params.sql" relativeToChangelogFile="true"/>
9  <include file="co/comments/application_params.sql" relativeToChangelogFile="true"/>
10 <include file="hr/comments/employees.sql" relativeToChangelogFile="true"/>
11 <include file="_custom/application_params_dml_1.sql" relativeToChangelogFile="true"/>
12 </databaseChangeLog>
```

Project Stage - commit [GIT branch 1.1]

The screenshot shows a Git commit interface with the following details:

- Unstaged:** A file named `stage.changelog.xml` is shown with two new lines added:

```
@@ -5,2 +5,7 @@
      http://www.liquibase.org/xml/ns/dbchangelog.xml
+<include file="hr/tables/employees.sql" relativeToChangelogEnd="true"/>
+<include file="hr/views/v_emp_names_phones.sql" relativeToChangelogEnd="true"/>
```
- Staged:** A commit titled "feature 1.1 - staged changes" is being prepared. The commit message is "feature 1.1 - staged changes".
- Commit Buttons:** "Commit 10 Files" and "Amend" buttons are visible.
- File Tree:** The commit includes 10 files:
 - `dist/releases/apex/apex.changelog.xml`
 - `dist/releases/apex/f101/f101.sql`
 - `dist/releases/apex/f101/f101.xml`
 - `dist/releases/main.changelog.xml`
 - `dist/releases/next/changes/1.1/co/comments/application_params.sql`
 - `dist/releases/next/changes/1.1/co/tables/application_params.sql`
 - `dist/releases/next/changes/1.1/hr/comments/employees.sql`
 - `dist/releases/next/changes/1.1/hr/tables/employees.sql`
 - `dist/releases/next/changes/1.1/hr/views/v_emp_names_phones.sql`
 - `dist/releases/next/changes/1.1/stage.changelog.xml`
- Author:** Grzegorczyk Rafal (rgrzegorczyk@pretius.com) on 22 May 2025 at 09:59:59 CEST.
- Refs:** Branch 1.1.
- SHA:** 5c993c570beb131f609cec9286c62d2152d4dd9d
- Parents:** 18770e2

ADMIN is written above a cartoon character icon in the bottom left corner.

5.1 Verify

Project Verify

- Test various aspects of your project (snapshots, changes, overall project)

```
> project verify -verbose
```

Parameters:

- “-verbose” - (optional) show extra details about current process.



Project Verify

- Test various aspects of your project (snapshots, changes, overall project).

```
11:53:46 SQL> project verify -verbose
----- Results Summary -----
Errors:  0
Warnings: 0
Info:   6
-----
Level      Group Name    Test Name          Message
-----      -----        -----          -----
INFO       settings     verifynonpublicsettings No unsupported internal settings found
INFO       init         verifyprojectname   The final project name will be "pmt_feedback"
INFO       examples    exampletest        a message
INFO       stage        stagechangelogcomplete Stage Changelog validation found no issues.
INFO       project     sqlclversion      SQLcl Version check passed
INFO       snapshot    verifysnapshot    Snapshot validation found no issues.
11:53:51 SQL>
```



6.Create release [1.1]

Project Release

- Transitions current work set into a released state and initiates a fresh set of work
 - The `release` command renames the contents of `dist/next` directory to the specified version number
 - Afterwards, it generates a new `next` folder inside `dist`
- Once the code is moved into a release, it must not be altered.

Questions? Ask below:



Project Release

```
> project release -version 1.1 -verbose
```

Parameters:

- “-version” - specify the version to label release
- “-verbose” - (optional) show extra details about current process.



The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. On the left, there's a sidebar with various icons: a file icon (EXPLORER), a magnifying glass (SEARCH), a circular arrow (REFRESH), a grid with a '1' (PROBLEMS), a bookmark (OUTLINE), a ship (OUTPUT), and a database (SQL HISTORY). The main area has a search bar at the top with the text "bombshell". Below it is a large terminal window displaying the command "10:21:19 SQL> project release -version 1.1 -verbose". At the bottom of the terminal, it says "viins | 1:37 | ADMIN | dev_admin". The status bar at the very bottom shows "1.1" and "Git Graph". A large watermark of a bomb with a lit fuse is overlaid across the center of the screen.



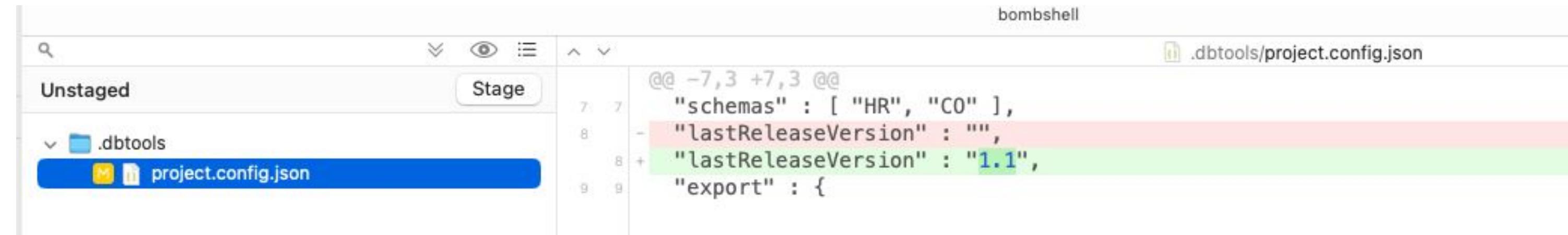
Several files and folders were created/changed:

```
10:21:19 SQL> project release -version 1.1 -verbose
Creating a release version 1.1 for the current body of work

Updated change:dist/releases/main.changelog.xml
Moved folder "dist/releases/next" to "dist/releases/1.1"
Created file: dist/releases/next
Created change:dist/releases/next/release.changelog.xml
Created change:dist/releases/next/release.changelog.xml
Created file: bombshell/.dbtools/project.config.json
Process completed successfully
10:21:28 SQL>
```



Version was changed in `project.config.json`



```
bombshell
diff --git a/.dbtools/project.config.json b/.dbtools/project.config.json
--- a/.dbtools/project.config.json
+++ b/.dbtools/project.config.json
@@ -7,3 +7,3 @@
 "schemas": [ "HR", "CO" ],
 - "lastReleaseVersion": "",
 + "lastReleaseVersion": "1.1",
 "export": {
```



Updated dist/releases/main.changelog.xml

```
dist/releases/main.changelog.xml
@@ -5,4 +5,5 @@
      http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-4.3.xsd">
- <include file="next/release.changelog.xml" relativeToChangelogFile="true"/>
+ <include file="1.1/release.changelog.xml" relativeToChangelogFile="true"/>
- <include file="apex/apex.changelog.xml" relativeToChangelogFile="true"/>
+ <include file="next/release.changelog.xml" relativeToChangelogFile="true"/>
</databaseChangeLog>
```



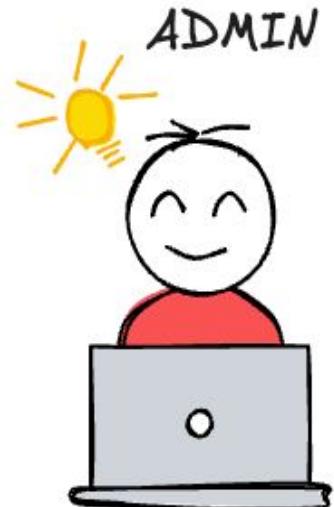
Created dist/releases/1.1/release.changelog.xml

The screenshot shows a diff tool interface comparing two XML files. The top file is 'dist/releases/main.changelog.xml' and the bottom file is 'dist/releases/1.1/release.changelog.xml'. A blue box highlights the title of the bottom file. A blue arrow points from the bottom file's title to the corresponding line in the top file.

```
diff --git a/dist/releases/main.changelog.xml b/dist/releases/1.1/release.changelog.xml
@@ -5,4 +5,5 @@
 http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-4.3.xsd">
-<include file="next/release.changelog.xml" relativeToChangelogFile="true"/>
+<include file="1.1/release.changelog.xml" relativeToChangelogFile="true"/>
-<include file="apex/apex.changelog.xml" relativeToChangelogFile="true"/>
+<include file="next/release.changelog.xml" relativeToChangelogFile="true"/>
</databaseChangeLog>
```



```
@@ -0,0 +1,8 @@
+<?xml version="1.0" encoding="UTF-8"?>
+<databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
+                     xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
+                     xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
+                                         http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-4.3.xsd">
+<!--CHANGES-->
+<include file="changes/1.1/stage.changelog.xml" relativeToChangelogFile="true"/>
+</databaseChangeLog>
```



Created dist/releases/1.1/changes/1.1/stage.changelog.xml

The terminal window shows the command to create the changelog file:

```
dist/releases/1.1/stage.changelog.xml
```

```
@@ -0,0 +1,8 @@
+<?xml version="1.0" encoding="UTF-8"?>
+<databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
+    xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
+    xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
+        http://www.liquibase.org/xml/ns/dbchangelog-dbchangelog-4.3.xsd">
+<!--CHANGES-->
+<include file="changes/1.1/stage.changelog.xml" relativeToChangelogFile="true"/>
+</databaseChangeLog>
```

A dotted arrow points from the terminal output to the file browser. The file browser shows the staged changes:

Unstaged

- .dbtools
- project.config.json
- dist
- releases
- 1.1
- changes
- 1.1
- _custom
- co
- hr

Stage

new file mode 100755

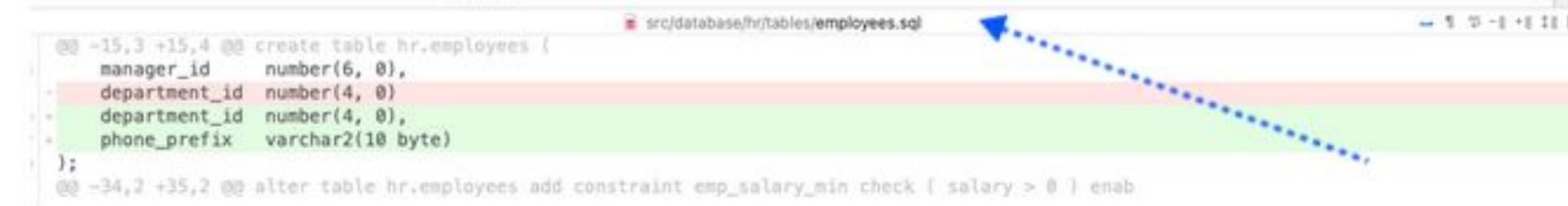
```
@@ -0,0 +1,12 @@
<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
    xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
        http://www.liquibase.org/xml/ns/dbchangelog-dbchangelog-4.3.xsd">
<include file="hr/tables/employees.sql" relativeToChangelogFile="true"/>
<include file="hr/views/v_emp_names_phones.sql" relativeToChangelogFile="true"/>
<include file="co/tables/application_params.sql" relativeToChangelogFile="true"/>
<include file="co/comments/application_params.sql" relativeToChangelogFile="true"/>
<include file="hr/comments/employees.sql" relativeToChangelogFile="true"/>
<include file="_custom/application_params_dml_1.sql" relativeToChangelogFile="true"/>
</databaseChangeLog>
```

ADMIN

A cartoon character icon is visible on the left.

Do you remember this slide?

RAFAL commits changes into GIT 1.1 branch



```
diff --git a/src/database/hr/tables/employees.sql b/src/database/hr/tables/employees.sql
@@ -15,3 +15,4 @@ create table hr.employees (
     manager_id    number(6, 0),
     department_id number(4, 0)
-    department_id number(4, 0),
-    phone_prefix  varchar2(10 byte)
+);
@@ -34,2 +35,2 @@ alter table hr.employees add constraint emp_salary_min check ( salary > 0 ) enab
```



ADMIN



SQLcl Project automatically created necessary DDL:

The screenshot shows the SQLcl IDE interface. On the left, the Explorer pane displays a project structure under 'BOMBSHELL' with folders like '.dbtools', 'dist', 'releases', '1.1', 'changes/1.1', '_custom', 'co', 'hr', 'comments', and 'tables'. A file named 'employees.sql' is selected in the Explorer. On the right, the main editor pane shows the contents of 'employees.sql'. The code is a Liquibase formatted SQL changeset. It includes a create table statement for 'hr.employees' with columns 'manager_id', 'department_id', 'department_id', and 'phone_prefix'. Below this, there is an alter table statement adding the 'phone_prefix' column. The code is annotated with comments: '-- liquibase formatted sql', '-- changeset HR:1747665513507 stripComments:false logica', '-- sqlcl_snapshot src/database/hr/tables/employees.sql:1', and '10' at the bottom.

```
src/database/hr/tables/employees.sql
@@ -15,3 +15,4 @@ create table hr.employees (
 15   manager_id      number(6, 0),
 16 - department_id  number(4, 0)
 16 + department_id  number(4, 0),
 17 + phone_prefix   varchar2(10 byte)

EXPLORER
BOMBSHELL
  .dbtools
  dist
    releases
      1.1
        changes/1.1
          _custom
          co
          hr
            comments
            tables
              employees.sql

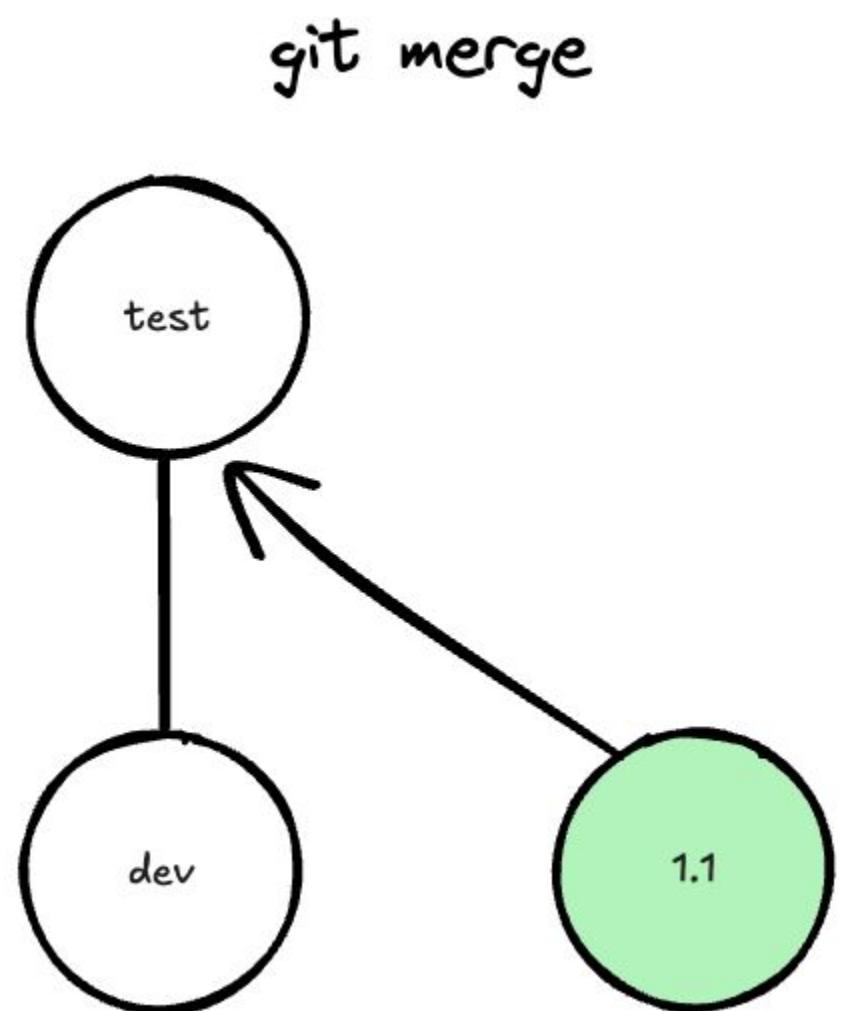
employees.sql R X
dist > releases > 1.1 > changes > 1.1 > hr > tables > employees.sql > ...
1 -- liquibase formatted sql
2 -- changeset HR:1747665513507 stripComments:false logica
3 -- sqlcl_snapshot src/database/hr/tables/employees.sql:1
4
5 alter table hr.employees add (
6   phone_prefix varchar2(10)
7 )
8 /
9
10
```

ADMIN

A cartoon character icon of a person with a sun-like head, wearing a red shirt, sitting at a laptop. The word 'ADMIN' is written above the character's head.

Merge 1.1 into test

- Merge 1.1 into test
- Merge 1.1 into dev as well



Merge 1.1 into test

The screenshot shows a Git commit interface with the following details:

- Staged:** .dbtools, dist/releases/1.1/changes/1.1/_custom/application_params_dml_1.sql, dist/releases/1.1/changes/1.1/co/comments/application_params.sql, dist/releases/1.1/changes/1.1/tables/application_params.sql, dist/releases/1.1/hr/comments/employees.sql, dist/releases/1.1/hr/tables/employees.sql, dist/releases/1.1/views/v_emp_names_phones.sql, dist/releases/1.1/stage.changelog.xml, dist/releases/1.1/release.changelog.xml, apex/f101/f101.sql, apex/f101/f101.xml, apex/apex.changelog.xml, next/changes/1.1/stage.changelog.xml, next/changes/1.1/release.changelog.xml, next/main.changelog.xml, utils/prechecks.sql.
- Merge branch '1.1' into test**
- Description:** (empty)
- Amend:**
- Commit 26 Files**

7. Generate artifact

Project Gen-Artifact

- Generates an artifact representing the current state of your project.

```
> sql -name test_admin
```

```
> project gen-artifact -name test -version 1.1 -verbose
```

Questions? Ask below:



Parameters (most used)

- "-name" - custom name (I named it with my target environment)
- "-version" - artifact version (use release version)



The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The title bar includes the pretius logo, a search bar with the text "bombshell", and several status icons. The left sidebar features the Explorer view, which is expanded to show the contents of a folder named "BOMBSHELL". Inside "BOMBSHELL", there are subfolders ".dbtools", "dist", and "src", along with files ".gitignore" and "readme.md". Below the Explorer is the Outline view, which is currently active. The main workspace contains a large, semi-transparent watermark of a bomb with a lit fuse. At the bottom of the screen is a terminal window. The tab bar at the top of the terminal window is labeled "TERMINAL" and shows the command "13:47:29 SQL> project gen-artifact -name test -version 1.1". The status bar at the bottom of the terminal window displays the user "viins", the session time "1:44", the profile "ADMIN", and the name "dev_admin". The bottom right corner of the status bar also shows a small icon of a person with a gear.

8.Deploy [1.1]

[DEV to TEST]

Project Deploy

- Deploys the artifact on a target database (TEST database, deployed from GIT branch 1.1)

```
> sql -name test_admin
```

```
> project deploy -file artifact/test-1.1.zip
```

Parameters

- “-file” - location of artifact file



The screenshot shows the pretius IDE interface. The top bar includes a search bar with the text "bombshell", a user icon, and system status indicators. The left sidebar features a file browser with the "BOMBSHELL" project expanded, showing subfolders ".dbtools", "artifact", "dist", "src", and ".gitignore". Below the file browser are several small icons: a document, a magnifying glass, a network, a play button, a folder with a question mark, a bookmark, a ship, and a database. The main workspace has tabs for "OUTPUT", "DEBUG CONSOLE", "TERMINAL", "SQL HISTORY", and "TASK MONITOR", with "TERMINAL" selected. The terminal window displays the command "16:13:43 SQL> project deploy -file artifact/test-1.1.zip". A message "Questions? Ask below:" is visible at the bottom right of the terminal area. The bottom status bar shows the session information "emacs | 1:42 | ADMIN | test_admin" and various system status icons.



Verify TEST after deployment



The image shows a screenshot of the SQL Developer interface. On the left, the connection tree shows a node for 'test_co' under 'Connections'. Under 'test_co', there are nodes for 'Customers', 'Priv', 'BOMBSHELL', 'DEVELOPMENT' (green dot), 'PRODUCTION' (red dot), 'TEST' (blue dot), 'test_admin', and 'Tables'. The 'APPLICATION_PARAMS' table is selected under the 'Tables' node. The main panel displays the 'APPLICATION_PARAMS' table with one row:

ID	NAME	VALUE
1	support_email	hello@pretius.com

The 'Data' tab is selected in the top navigation bar. Below the table are standard database operation buttons: Insert, Export, Delete Selected, Commit, Undo All, and a search/filter icon.

Verify TEST after deployment

SQL DEVELOPER

...

> SQL SNIPPETS

✓ CONNECTIONS

✓ Priv

✓ BOMBSHELL

✓ TEST

> test_admin

> test_co

✓ test_hr

✓ Tables

> COUNTRIES

> DATABASECHANGELOG

> DATABASECHANGELOGLOCK

> DATABASECHANGELOG_ACTIONS

> DEPARTMENTS

✓ EMPLOYEES

EMPLOYEE_ID
FIRST_NAME
LAST_NAME
EMAIL
PHONE_NUMBER
HIRE_DATE

EMPLOYEES X

test_hr > object > HR > TABLE > EMPLOYEES

Columns Data JSON Schema Constraints Grants
Triggers Flashback Dependencies Details Partitic
SQL

	COLUMN_NAME	DATA_TYPE
1	EMPLOYEE_ID	NUMBER(6,0)
2	FIRST_NAME	VARCHAR2(20 BYTE)
3	LAST_NAME	VARCHAR2(25 BYTE)
4	EMAIL	VARCHAR2(25 BYTE)
5	PHONE_NUMBER	VARCHAR2(20 BYTE)
6	HIRE_DATE	DATE
7	JOB_ID	VARCHAR2(10 BYTE)
8	SALARY	NUMBER(8,2)
9	COMMISSION_PCT	NUMBER(2,2)
10	MANAGER_ID	NUMBER(6,0)
11	DEPARTMENT_ID	NUMBER(4,0)
12	PHONE_PREFIX	VARCHAR2(10 BYTE)



Verify TEST after deployment

SQL DEVELOPER

... 

> SQL SNIPPETS

▽ CONNECTIONS

- ▽ Priv
- ▽ BOMBSHELL
- ▽ TEST 
- ▽ test_hr 
- ▽ Views

> **V_EMP_NAMES_PHONES** 

> Duality Views

> Indexes

> Packages

> Procedures

 ADMIN

V_EMP_NAMES_PHONES 

test_hr > object > HR > VIEW > V_EMP_NAMES_PHONES

Columns Data JSON Schema Grants Dependencies Details

Triggers SQL Errors

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT
1	FIRST_NAME	VARCHAR2(20)	Yes	(null)
2	LAST_NAME	VARCHAR2(25)	No	(null)
3	PHONE_NUMBER	VARCHAR2(20)	Yes	(null)
4	PHONE_PREFIX	VARCHAR2(10)	Yes	(null)

Liquibase - tracking table



ID	AUTHOR	FILENAME	DATEEXECUTED
1	HR	1.1/hr/tables/employees.sql	22/05/25 14:15:21.111
2	HR	1.1/hr/views/v_emp_names_phones.sql	22/05/25 14:15:23.336
3	CO	1.1/co/tables/application_params.sql	22/05/25 14:15:26.138
4	co	1.1/co/comments/application_params.sql	22/05/25 14:15:29.005
5	hr	1.1/hr/comments/employees.sql	22/05/25 14:15:34.786
6	Sqlcl	1.1/_custom/application_params_dml_1.sql	22/05/25 14:15:36.978
7	INSTALL_101	releases/apex/f101/f101.xml	22/05/25 14:16:06.526

We've just deployed
release 1.1 to TEST but...

What if production is on fire?



<<time>>

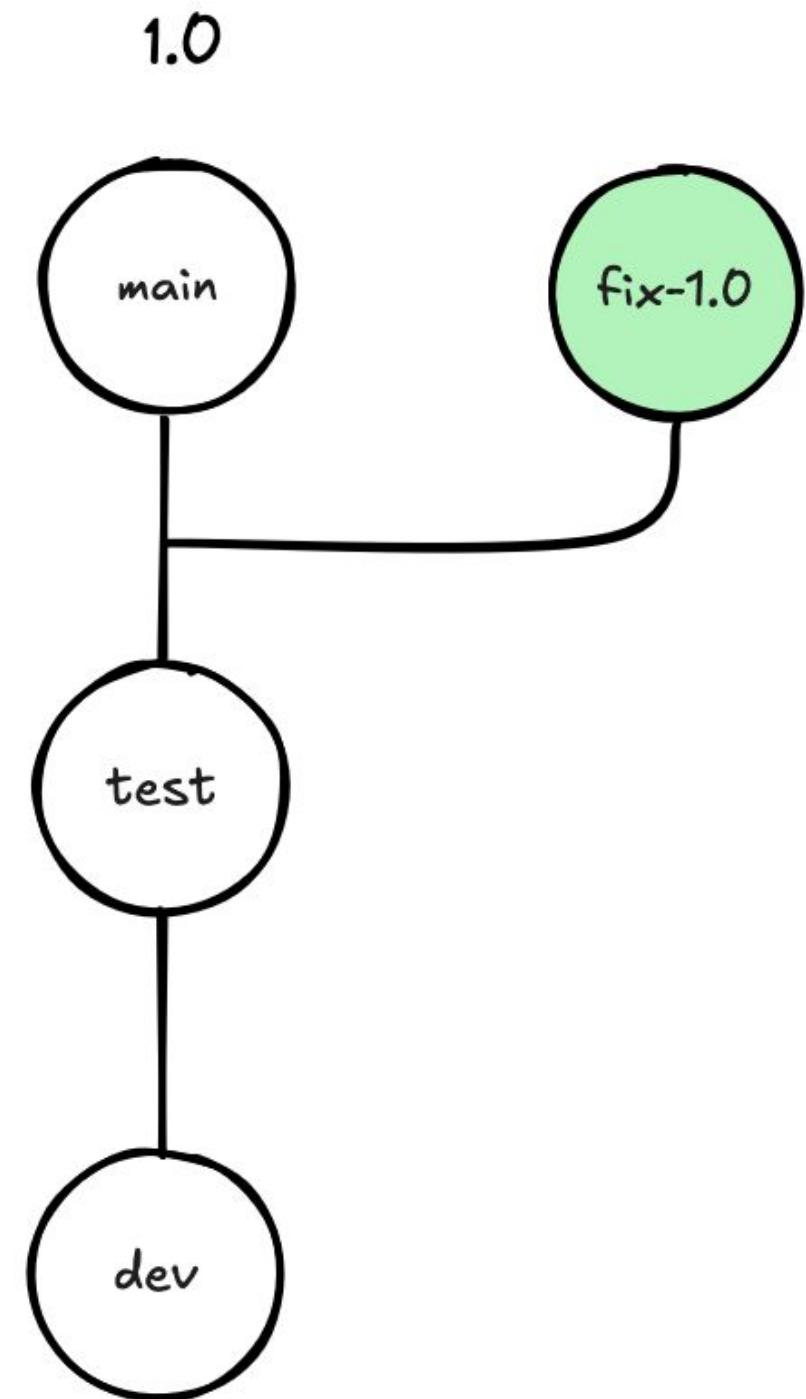
Issue on production

```
Error starting at line : 1 in command -
Insert into HR.JOBS (JOB_ID,JOB_TITLE,MIN_SALARY,MAX_SALARY)
values ('APEX','Oracle APEX Professional Certified APEX Low-Code Pretius Developer',20080,40000)
Error at Command Line : 2 Column : 16
Error report -
SQL Error: ORA-12899: value too large for column "HR"."JOBS"."JOB_TITLE" (actual: 66, maximum: 35)
```

Questions? Ask below:



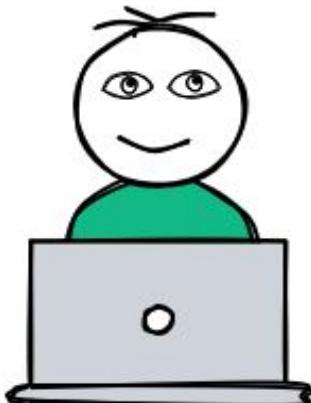
Create new GIT branch [fix-1.0] and switch to it



Prepare a fix on DEV environment

```
ALTER TABLE hr.jobs MODIFY job_title varchar2(100);
```

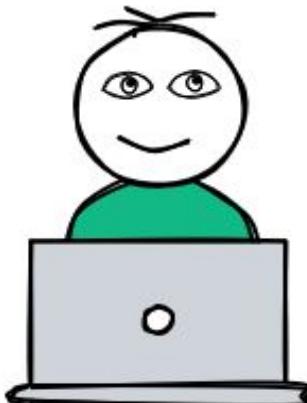
GEORGE



Export changes using SQLcl Project Export

```
> project export -objects HR.JOB$
```

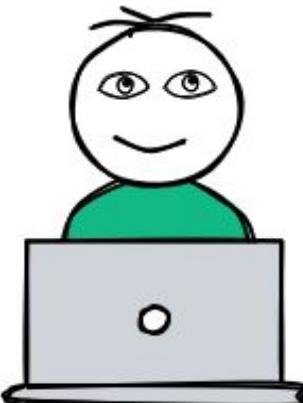
GEORGE



Export changes using SQLcl Project Export

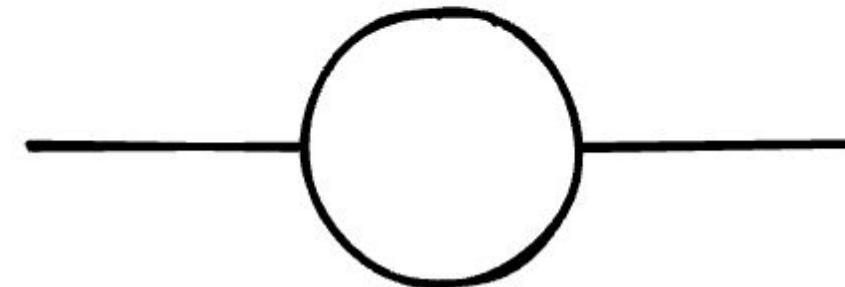
```
14:17:55 SQL> project export -objects HR.JOB$  
The current connection DEVELOPMENT_HIGH HR will be used for all operations  
*** TABLES ***  
*** COMMENTS ***  
-----  
COMMENT 1  
TABLE 1  
-----  
Exported 2 objects  
Elapsed 21 sec  
14:18:44 SQL>
```

GEORGE

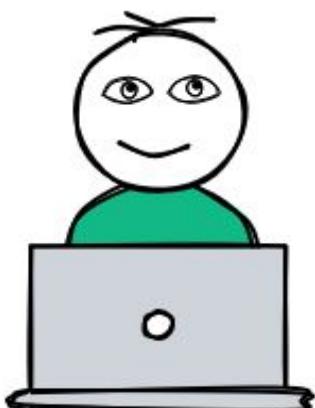


GEORGE commits changes into fix-1.0

git commit



GEORGE



Project Stage

```
> sql -name dev_admin  
> !git checkout fix-1.0  
> project stage -branch-name main
```

- Compares the current branch (**fix-1.0**) with the production branch (**main**) to create changelogs that apply the necessary database updates.
- Parameter `-branch-name` is optional in that case as **main** branch is default in `project.config`



Project Stage

```
18:41:23 SQL> project stage -branch-name main -verbose
```

Starting execution of stage command using the current branch

Stage is Comparing:

Old Branch refs/heads/main
New Branch refs/heads/fix-1.0

```
Created dir: dist/releases
Created dir: dist/utils
Created dir: dist/releases/next
Created dir: dist/releases/next/changes
Created dir: dist/releases/next/changes/fix-1.0
Created dir: dist/releases/next/changes/fix-1.0/_custom
Created change:dist/releases/next/changes/fix-1.0/stage.changelog.xml
Created change:dist/releases/main.changelog.xml
Created change:dist/releases/next/release.changelog.xml
Created file: dist/utils/recompile.sql
Created file: dist/utils/prechecks.sql
Updated file: dist/releases/next/changes/fix-1.0/hr/tables/jobs.sql
Updated change:dist/releases/next/changes/fix-1.0/stage.changelog.xml
Updated change:dist/releases/next/changes/fix-1.0/stage.changelog.xml
Updated change:dist/releases/next/release.changelog.xml
Updated change:dist/releases/next/release.changelog.xml
Created change:dist/releases/apex/apex.changelog.xml
Created change:dist/releases/apex/apex.changelog.xml
Created file: dist/releases/apex/f101/f101.sql
Created change:dist/releases/apex/f101/f101.xml
Updated change:dist/releases/apex/apex.changelog.xml
Updated change:dist/releases/main.changelog.xml
```

Completed executing stage command on branch: fix-1.0

Stage processesing completed, please review and commit your changes to repository



Project Stage

- Review generated files



The image shows a screenshot of a file explorer window titled "EXPLORER". The left pane displays a tree structure of project folders under "BOMBSHELL". The "dist" folder contains "releases" and "next". The "next" folder contains "changes/fix-1.0", which in turn contains "_custom" and "hr/tables". Inside "hr/tables", there is a file named "jobs.sql" and another file named "stage.changelog.xml". The right pane shows the contents of the "jobs.sql" file. The file path is listed as "dist > releases > next > changes > fix-1.0 > hr > tables > jobs.sql". The file content is a SQL script:

```
1 -- liquibase formatted sql
2 -- changeset HR:1748018521326 stripComment
3 -- sqlcl_snapshot src/database/hr/tables/j
4
5 alter table hr.jobs modify (
6   job_title varchar2(100)
7 )
8 /
9
10
```

A blue rectangular box highlights the line "alter table hr.jobs modify (".

Project Stage - commit [GIT branch fix-1.0]

Staged Unstage

- dist
 - releases
 - apex
 - next
 - changes
 - fix-1.0
 - hr
 - tables
 - jobs.sql
 - stage.changelog.xml
 - release.changelog.xml
 - main.changelog.xml
 - utils

fix-1.0 Staged changes 28

Description

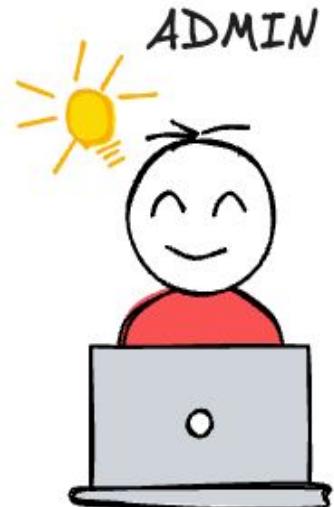
Amend

Commit 9 Files



Project Release

```
> project release -version fix-1.0 -verbose
```

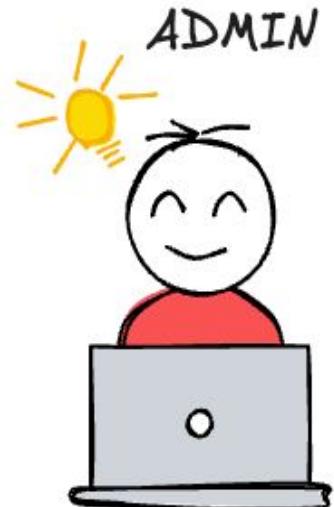


Several files and folders were created/changed:

▼ TERMINAL

```
18:54:25 SQL> project release -version fix-1.0 -verbose
Creating a release version fix-1.0 for the current body of work

Updated change:dist/releases/main.changelog.xml
Moved folder "dist/releases/next" to "dist/releases/fix-1.0"
Created file: dist/releases/next
Created change:dist/releases/next/release.changelog.xml
Created change:dist/releases/next/release.changelog.xml
Created file: bombshell/.dbtools/project.config.json
Process completed successfully
18:54:35 SQL>
```



Version was changed in `project.config.json`

```
📁 .dbtools/project.config.json
@@ -7,3 +7,3 @@
"schemas" : [ "HR", "CO" ],
- "lastReleaseVersion" : "",
+ "lastReleaseVersion" : "fix-1.0",
"export" : {
```



Updated dist/releases/main.changelog.xml

```
dist/releases/main.changelog.xml @@ -5,4 +5,5 @@
http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-4.3.xsd">
- <include file="next/release.changelog.xml" relativeToChangelogFile="true"/>
+ <include file="fix-1.0/release.changelog.xml" relativeToChangelogFile="true"/>
<include file="apex/apex.changelog.xml" relativeToChangelogFile="true"/>
+ <include file="next/release.changelog.xml" relativeToChangelogFile="true"/>
</databaseChangeLog>
No newline at end of file
```



Created dist/releases/fix-1.0/release.changelog.xml

The screenshot shows a diff tool comparing two XML files. The top window is titled "dist/releases/main.changelog.xml" and the bottom window is titled "dist/releases/fix-1.0/release.changelog.xml". A blue arrow points from the bottom window back up to the top window, indicating the flow of changes.

dist/releases/main.changelog.xml

```
@@ -5,4 +5,5 @@
      http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-4.3.xsd">
-<include file="next/release.changelog.xml" relativeToChangelogFile="true"/>
+<include file="fix-1.0/release.changelog.xml" relativeToChangelogFile="true"/>
 <include file="apex/apex.changelog.xml" relativeToChangelogFile="true"/>
+<include file="next/release.changelog.xml" relativeToChangelogFile="true"/>
</databaseChangeLog>
No newline at end of file
```

dist/releases/fix-1.0/release.changelog.xml

```
@@ -0,0 +1,8 @@
+<?xml version="1.0" encoding="UTF-8"?>
+<databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
+                     xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
+                     xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
+                                         http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-4.3.xsd">
+<!--CHANGES-->
+<include file="changes/fix-1.0/stage.changelog.xml" relativeToChangelogFile="true"/>
+</databaseChangeLog>
```



Created dist/releases/fix-1.0/changes/fix-1.0/stage.changelog.xml

```
dist/releases/fix-1.0/release.changelog.xml
```

```
@@ -0,0 +1,8 @@
+<?xml version="1.0" encoding="UTF-8"?>
+<databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
+    xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
+    xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
+        http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-4.3.xsd">
+<!--CHANGES-->
+<include file="changes/fix-1.0/stage.changelog.xml" relativeToChangelogFile="true"/>
+</databaseChangeLog>
```

```
stage.changelog.xml R X
```

```
dist > releases > fix-1.0 > changes > fix-1.0 > stage.changelog.xml
```

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3      xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
4          xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
5              http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-4.3.xsd">
6  <include file="hr/tables/jobs.sql" relativeToChangelogFile="true"/>
7  </databaseChangeLog>
```

ADMIN

SQLcl Project automatically created necessary DDL:



The image shows a screenshot of a development environment. On the left, there is a code editor window displaying a diff of a SQL script named `src/database/hr/tables/jobs.sql`. The diff highlights changes in lines 2, 3, and 4, where the `job_title` column is being modified from `varchar2(35 byte)` to `varchar2(100 byte)`. A dotted arrow points from this code editor to a file browser window on the right.

The file browser window, titled "EXPLORER", shows a project structure under the folder "BOMBSHELL". It includes sub-folders ".dbtools", "artifact", "dist", "releases", "fix-1.0", "changes/fix-1.0", "_custom", and "hr/tables". The "jobs.sql" file is selected in the browser.

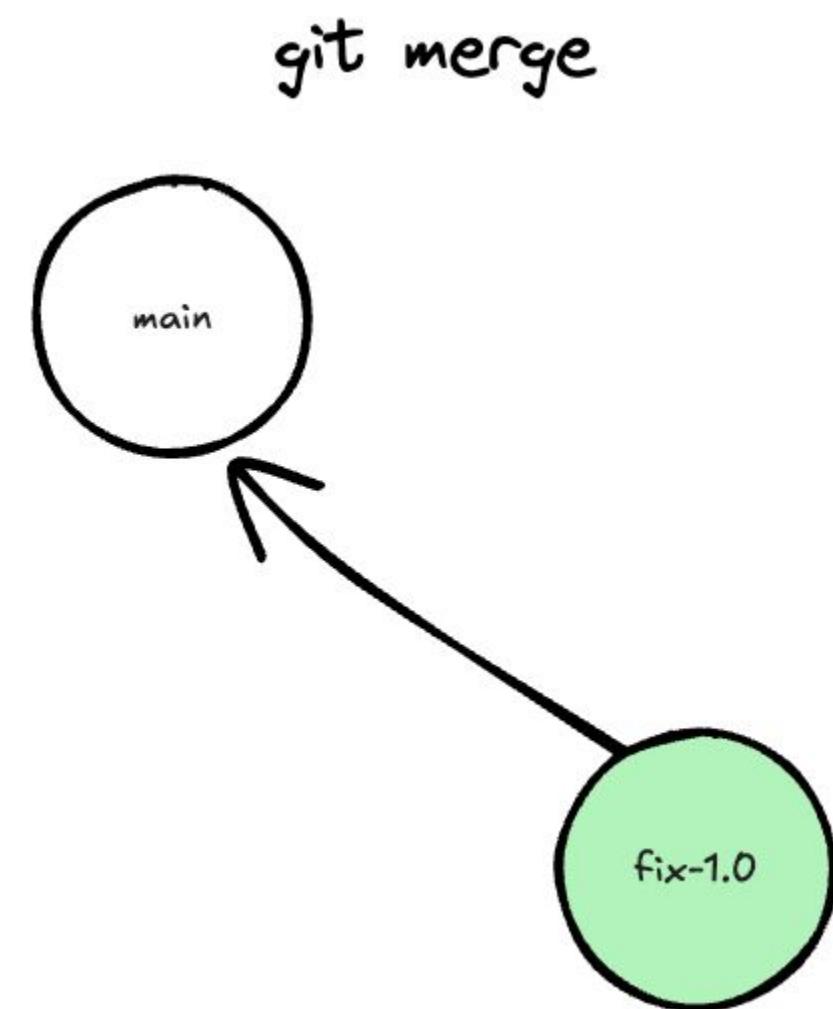
The rightmost window is a code editor showing the contents of the `jobs.sql` file. The code is formatted as Liquibase SQL. It includes a header indicating it is Liquibase-formatted SQL, followed by a changeset for HR:1748018521326, and a command to alter the `hr.jobs` table to change the `job_title` column to `varchar2(100)`.

```
diff --git a/src/database/hr/tables/jobs.sql b/src/database/hr/tables/jobs.sql
@@ -2,3 +2,3 @@ create table hr.jobs (
 2     job_id      varchar2(10 byte),
 3     job_title   varchar2(35 byte)
 4     job_title   varchar2(100 byte)
 5         constraint job_title_nn not null enable,
 6
@@ -19,2 +19,2 @@ alter table hr.jobs
 1
 2
 3
 4
 5
 6
 7
 8
 9
10
```

```
-- liquibase formatted sql
-- changeset HR:1748018521326 stripComments=true
-- sqlcl_snapshot src/database/hr/tables/jobs.sql
alter table hr.jobs modify (
    job_title varchar2(100)
)
/
```

Merge fix-1.0 into main [production]

- Merge fix-1.0 into main
- Merge fix-1.0 into dev, test



Project Gen-Artifact

- Generates an artifact representing the current state of your project.

```
> sql -name prod_admin  
> !git checkout main  
> project gen-artifact -name prod -version fix-1.0
```

```
19:25:39 SQL> project gen-artifact -name prod -version fix-1.0  
Your artifact has been generated prod-fix-1.0.zip  
19:25:43 SQL> █
```



Project Deploy

- Deploys the artifact on a target database

```
> sql -name prod_admin
```

```
> project deploy -file artifact/prod-fix-1.0.zip
```



The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The title bar displays the project name "fix-1.0". The Explorer sidebar on the left shows a folder structure for "BOMBSHELL" containing ".dbtools", "artifact", "dist", "src", ".gitignore", and "readme.md". The Terminal tab is active, showing the prompt "19:38:40 SQL>". The bottom status bar shows the command "viins | 1:0 | ADMIN | prod_admin". A large watermark of a bomb with a lit fuse is overlaid on the center of the screen. In the bottom-left corner, there is a cartoon illustration of a person sitting at a laptop with a sun icon above their head, labeled "ADMIN". The bottom navigation bar includes icons for file operations like "main", "Git Graph", and "bombshell".

Tips and tricks

Keep your DB users minimally privileged

- Do not use OVERprivileged users like SYS
- Use PROXY users instead:

```
CREATE USER JOHN IDENTIFIED BY password;
```

```
ALTER USER HR GRANT CONNECT THROUGH JOHN;
```

Questions? Ask below:

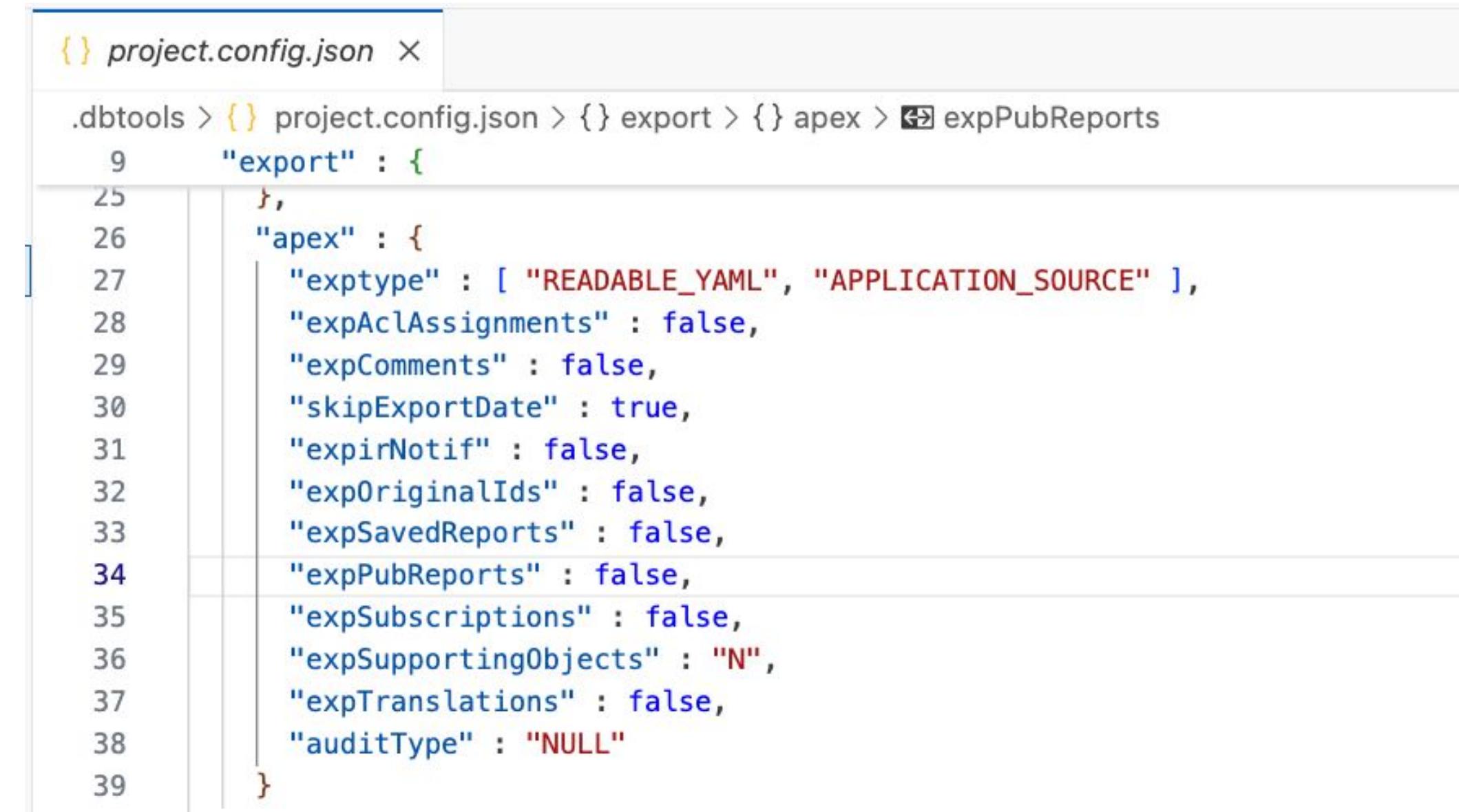
- Consider schema-level privileges (23 AI)

```
GRANT SELECT ANY TABLE ON SCHEMA HR TO JOHN;
```

```
GRANT EXECUTE ANY PROCEDURE ON SCHEMA HR TO JOHN;
```



Adjust your APEX export



The image shows a code editor window with the file name "project.config.json" at the top. The file path ".dbtools > project.config.json > export > apex > expPubReports" is visible above the code. The code itself is a JSON object with several properties:

```
{ "export": { "apex": { "exptype": [ "READABLE_YAML", "APPLICATION_SOURCE" ], "expAclAssignments": false, "expComments": false, "skipExportDate": true, "expirNotif": false, "expOriginalIds": false, "expSavedReports": false, "expPubReports": false, "expSubscriptions": false, "expSupportingObjects": "N", "expTranslations": false, "auditType": "NULL" } } }
```

The code is numbered from 9 to 39 on the left side.

Use default connectionName

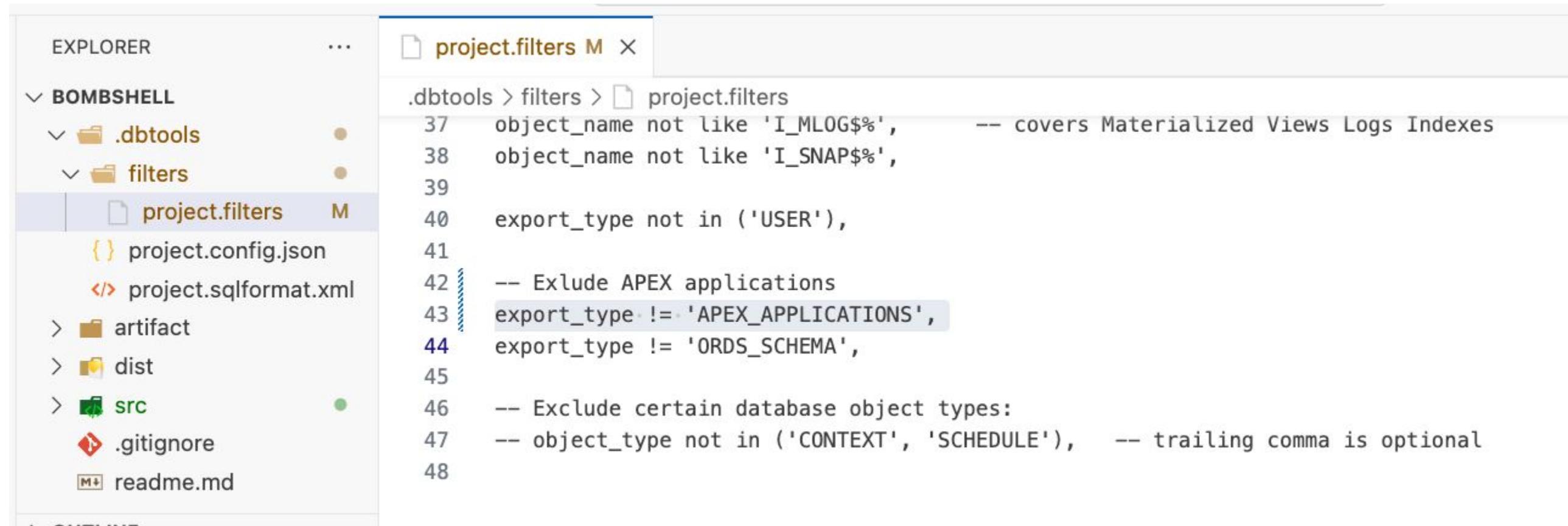
The screenshot shows the Oracle SQL Developer interface with the following details:

- EXPLORER** pane: A project named "BOMBSHELL" is expanded, showing a folder ".dbtools" which contains "filters", "project.config.json", and "project.sqlformat.xml".
- TERMINAL** pane: Displays the output of SQL commands:

```
09:39:35 09:31:28 SQL> show connection
SP2-0640: Not connected
09:39:40 09:31:28 SQL> project export -o APEX.101
Current connection is not set
Connecting using the named connection dev_admin from the project config
Connected.
```
- project.config.json** file content (shown in the code editor):

```
{
  "project": "bombshell",
  "sqlcl": {
    "connectionName": "dev_admin",
    "version": "25.1.1.0"
  }
}
```

Exclude APEX from export

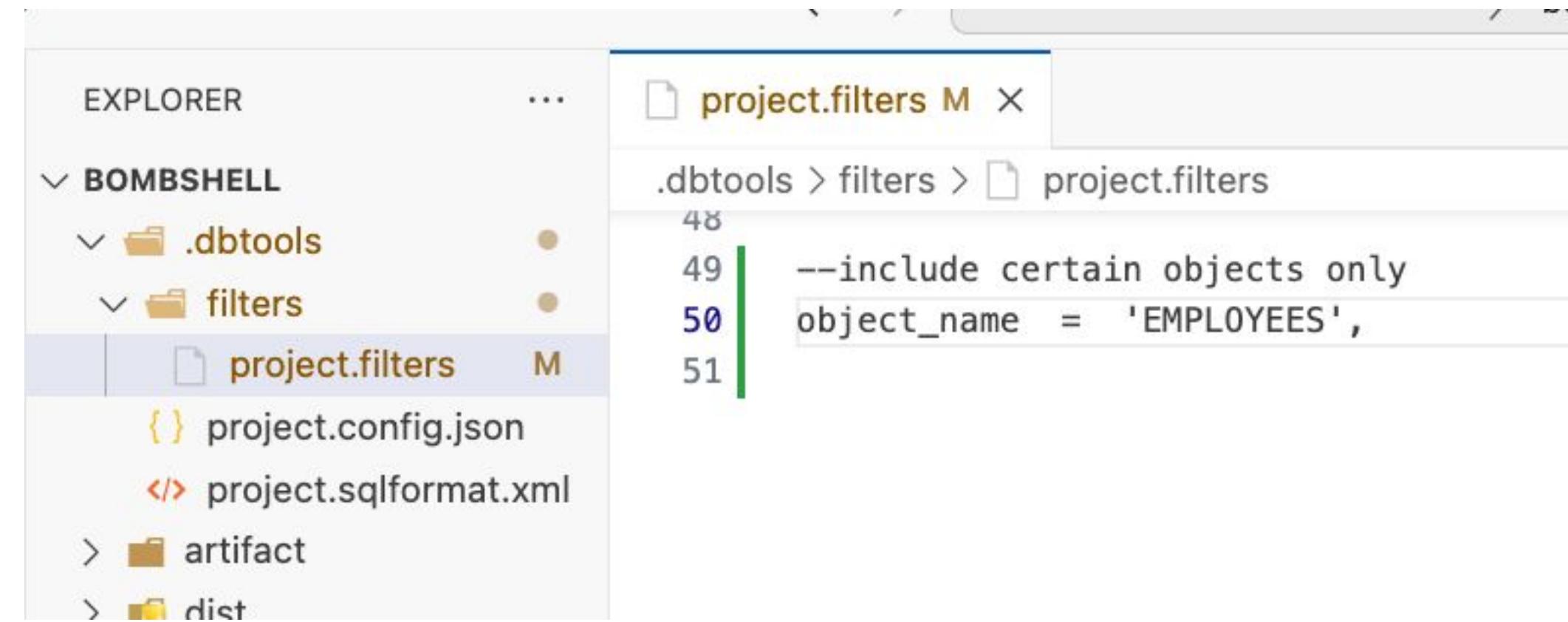


The image shows a screenshot of a code editor with an Explorer sidebar on the left. The Explorer sidebar lists several project folders and files under a 'BOMBSHELL' root. The 'filters' folder contains a file named 'project.filters'. The code editor's main area displays the contents of this file:

```
.dbtools > filters > project.filters
37 object_name not like 'I_MLOG%',      -- covers Materialized Views Logs Indexes
38 object_name not like 'I_SNAP$%',
39
40 export_type not in ('USER'),
41
42 -- Exclude APEX applications
43 export_type != 'APEX_APPLICATIONS',
44 export_type != 'ORDS_SCHEMA',
45
46 -- Exclude certain database object types:
47 -- object_type not in ('CONTEXT', 'SCHEDULE'), -- trailing comma is optional
48
```

The code uses SQL-like syntax with numbered lines (37, 38, 39, etc.) to filter database objects. It excludes materialized views, logs, and indexes in line 37, and apex applications and ords schemas in line 43.

Export certain objects only



```
project.filters M
.dbtools > filters > project.filters
48
49 --include certain objects only
50 object_name = 'EMPLOYEES',
51
```

Interested what's in the background of SQLcl Project?

- Run your commands with **-debug** parameter

```
> project export -debug
```

The screenshot shows the pretius IDE interface with the following details:

- Top Bar:** Includes standard OS window controls (red, yellow, green), a back/forward button, a search bar containing "bombshell", a pinned tab icon, and a status bar.
- Left Sidebar:** Features icons for Explorer, Search, Terminal, Database, and Settings. The Terminal icon has a blue circle with the number "2" indicating new activity.
- Central Area:** The Explorer panel shows a project structure under ".dbtools > filters > project.filters". The code editor shows the following SQL command:

```
47 -- object type not in ('CONTEXT', 'SCHEDULE'). -- trailing comma is optional
```
- Terminal Tab:** Active tab, showing the output of the command "project export -debug". The output includes:
 - DISPLAY ARGUMENTS -----
 - LIST: null
 - OBJECTS: null
 - SCHEMAS: null
 - THREADS: null
 - DEBUG: true
 - VERBOSE: false
 - END ARGUMENTS -----
 - List:false
 - Objects:[]
 - Schemas:[]
 - Threads:5
 - Debug: true
 - Verbose: false
 - Exporting database objects...
 - The current connection DEVELOPMENT_HIGH ADMIN will be used for all operations
 - /* Main dictionary query against ALL_OBJECTS: */
 - select * from (SELECT OWNER, OBJECT_NAME, SUBOBJECT_NAME, OBJECT_ID, DATA_OBJECT_ID,
 - OBJECT_TYPE, CREATED, LAST_DDL_TIME, TIMESTAMP, STATUS,
 - TEMPORARY, GENERATED, SECONDARY, NAMESPACE, EDITION_NAME,
 - SHARING, EDITIONABLE, ORACLE_MAINTAINED, APPLICATION, DEFAULT_COLLATION,
- Bottom Status Bar:** Shows file version (1.1*), Git Graph, and a status message "bombshell". It also displays line 48, column 1, spaces 4, UTF-8 encoding, and LF line endings.

ADMIN [DEV] [ACE] [CHICAGO]

Worksheet Query Builder

```
1 select * from (SELECT OWNER, OBJECT_NAME, SUBOBJECT_NAME, OBJECT_ID, DATA_OBJECT_ID,
2 OBJECT_TYPE, CREATED, LAST_DDL_TIME, TIMESTAMP, STATUS,
3 TEMPORARY, GENERATED, SECONDARY, NAMESPACE, EDITION_NAME,
4 SHARING, EDITIONABLE, ORACLE_MAINTAINED, APPLICATION, DEFAULT_COLLATION,
5 DUPLICATED, SHARDED, CREATED_APPID, CREATED_VSNID,
6 MODIFIED_APPID, MODIFIED_VSNID, 'ALL_OBJECTS' EXPORT_TYPE
7 FROM Dba_objects) Dba_objects where 1=1
8 filters --
9 and (not /*object_type in ('TABLE','VIEW','SEQUENCE') and*/ object_name like 'AQ$%') --<-- sqlcl internal.fixed.filters
10 and (export_type not in ('USER')) --<-- project.filters
11 and (not(Dba_objects.object_name = 'HTMLDB_PLAN_TABLE' and Dba_objects.object_type = 'TABLE')) --<-- sqlcl internal.fixed.filters
12 and (oracle_maintained = 'N') --<-- sqlcl internal.fixed.filters
13 and (not (object_type = 'TABLE' and object_name like 'MLOG$\_%' escape '\'
14 /* object_name in (select log_table from all_mview_logs)*/) ) --<-- sqlcl internal.fixed.filters
15 and (not (object_type = 'TRIGGER' and object_name ='DATABASECHANGELOG_ACTIONS_TRG')) --<-- project.filters
16 and (object_name not like 'I MLOG$%') --<-- project.filters
```

Script Output x | Query Result x

SQL | All Rows Fetched: 26 in 0.381 seconds

OWNER	OBJECT_NAME	SUBOBJECT_NAME	OBJECT_ID	DATA_OBJECT_ID	OBJECT_TYPE	CREATED	LAST_DDL_TIME	TIMESTAMP	STATUS	TEMPORARY	GENER
CO	APPLICATION_PARAMS	(null)	133896	133896	TABLE	2025-05-19 12:58:32	2025-05-19 12:58:37	2025-05-19 12:58:32	VALID	N	N
CO	CUSTOMERS	(null)	131985	131985	TABLE	2025-05-12 07:59:23	2025-05-12 08:00:54	2025-05-12 08:00:54	VALID	N	N
CO	INVENTORY	(null)	132002	132002	TABLE	2025-05-12 07:59:24	2025-05-12 08:00:54	2025-05-12 08:00:54	VALID	N	N
CO	ORDERS	(null)	131997	131997	TABLE	2025-05-12 07:59:24	2025-05-12 08:00:54	2025-05-12 08:00:54	VALID	N	N
CO	ORDER_ITEMS	(null)	132001	132001	TABLE	2025-05-12 07:59:24	2025-05-12 07:59:37	2025-05-12 07:59:24	VALID	N	N
CO	PRODUCTS	(null)	131991	131991	TABLE	2025-05-12 07:59:24	2025-05-12 08:00:53	2025-05-12 08:00:53	VALID	N	N
CO	SHIPMENTS	(null)	131999	131999	TABLE	2025-05-12 07:59:24	2025-05-12 08:00:54	2025-05-12 08:00:54	VALID	N	N
CO	STORES	(null)	131987	131987	TABLE	2025-05-12 07:59:23	2025-05-12 08:00:53	2025-05-12 08:00:53	VALID	N	N
HR	COUNTRIES	(null)	132028	132028	TABLE	2025-05-12 08:01:38	2025-05-12 08:01:47	2025-05-12 08:01:38	VALID	N	N
HR	DEPARTMENTS	(null)	132033	132033	TABLE	2025-05-12 08:01:39	2025-05-12 08:01:54	2025-05-12 08:01:39	VALID	N	N
HR	EMPLOYEES	(null)	132038	132038	TABLE	2025-05-12 08:01:40	2025-05-24 05:51:45	2025-05-23 12:12:48	VALID	N	N
HR	JOB_HISTORY	(null)	132036	132036	TABLE	2025-05-12 08:01:40	2025-05-23 12:17:52	2025-05-23 12:17:52	VALID	N	N
HR	JOB_HISTORY	(null)	132042	132042	TABLE	2025-05-12 08:01:41	2025-05-12 08:01:47	2025-05-12 08:01:41	VALID	N	N
HR	LOCATIONS	(null)	132030	132030	TABLE	2025-05-12 08:01:39	2025-05-12 08:01:45	2025-05-12 08:01:39	VALID	N	N
HR	REGIONS	(null)	132026	132026	TABLE	2025-05-12 08:01:38	2025-05-12 08:01:44	2025-05-12 08:01:38	VALID	N	N
HR	ADD_JOB_HISTORY	(null)	132058	(null)	PROCEDURE	2025-05-12 08:01:55	2025-05-12 08:01:55	2025-05-12 08:01:55	VALID	N	N
HR	SECURE_DML	(null)	132056	(null)	PROCEDURE	2025-05-12 08:01:54	2025-05-12 08:01:54	2025-05-12 08:01:54	VALID	N	N
HR	DEPARTMENTS_SEQ	(null)	132035	(null)	SEQUENCE	2025-05-12 08:01:40	2025-05-12 08:01:40	2025-05-12 08:01:40	VALID	N	N
HR	EMPLOYEES_SEQ	(null)	132041	(null)	SEQUENCE	2025-05-12 08:01:41	2025-05-12 08:01:41	2025-05-12 08:01:41	VALID	N	N
HR	LOCATIONS_SEQ	(null)	132032	(null)	SEQUENCE	2025-05-12 08:01:39	2025-05-12 08:01:39	2025-05-12 08:01:39	VALID	N	N
CO	CUSTOMER_ORDER_PRODUCTS	(null)	132004	(null)	VIEW	2025-05-12 07:59:25	2025-05-15 11:56:46	2025-05-12 07:59:25	VALID	N	N
CO	PRODUCT_ORDERS	(null)	132007	(null)	VIEW	2025-05-12 07:59:25	2025-05-15 11:56:52	2025-05-12 07:59:25	VALID	N	N
CO	PRODUCT_REVIEWS	(null)	132006	(null)	VIEW	2025-05-12 07:59:25	2025-05-12 07:59:40	2025-05-12 07:59:25	VALID	N	N
CO	STORE_ORDERS	(null)	132005	(null)	VIEW	2025-05-12 07:59:25	2025-05-15 11:56:59	2025-05-15 11:56:59	VALID	N	N
HR	EMP_DETAILS_VIEW	(null)	132044	(null)	VIEW	2025-05-12 08:01:42	2025-05-12 08:01:42	2025-05-12 08:01:42	INVALID	N	N
HR	V_EMP_NAMES_PHONES	(null)	133884	(null)	VIEW	2025-05-19 10:39:09	2025-05-19 10:39:09	2025-05-19 10:39:09	VALID	N	N

Want more about certain SQLcl command?

```
> help project [command][examples|syntax]
```

Want more about certain SQLcl command?

```
13:47:44 13:47:31 SQL> help project export  
Export database objects to your repository
```

Usage:

```
project|proj  export|ex {OPTIONS}
```

Options:

```
-list|-l
```

List all objects in alphabetical order

Outputs the list of objects that will be exported based on the current parameters

```
-objects|-o <objects>
```

Define a comma-separated list of objects to export, e.g. HR.COUNTRIES,HR.EMPLOYEES

```
-schemas|-s <schemas>
```

Specify a comma-separated list of schemas to export, e.g., HR

Want more about certain SQLcl command?

```
13:48:58 13:47:31 SQL> help project export examples
```

- Examples of custom export filters are in ".dbtools/filters/project.filters"
- ```
SQL> project export -debug -list -objects REGIONS,COUNTRIES -schemas HR,SCOTT -threads 4
```

```
13:49:02 13:47:31 SQL> █
```

```
13:49:38 13:47:31 SQL> help project stage syntax
```

```
project|proj stage|st [[add-custom|ac]] | [[-branch-name|-bn <branch-name>] [-debug|-de] [-verbose|-v]]
```

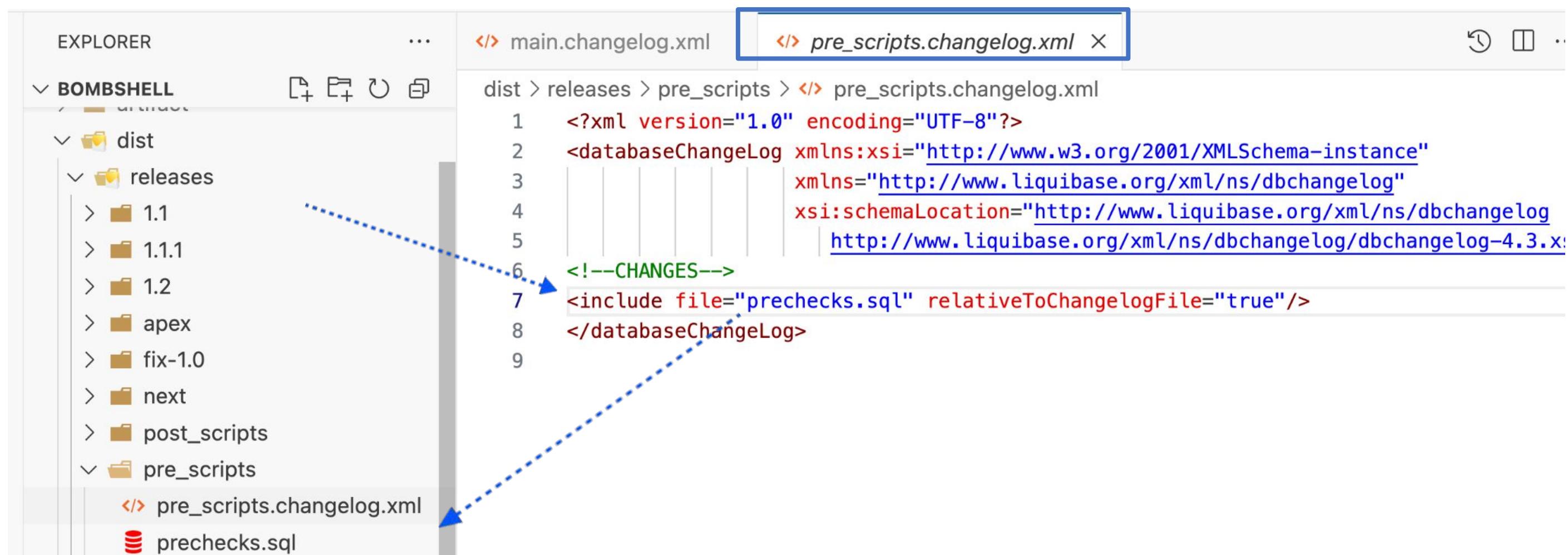
```
13:50:34 13:47:31 SQL> █
```

# Want to add repeatable PRE and POST scripts to your deployments?

- Modify main.changelog.xml

```
</> main.changelog.xml <...>
dist > releases > </> main.changelog.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3 | xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
4 | xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
5 | http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-4.3.xsd">
6 <include file="pre_scripts/pre_scripts.changelog.xml" relativeToChangelogFile="true"/>
7 <include file="fix-1.0/release.changelog.xml" relativeToChangelogFile="true"/>
8 <include file="1.1/release.changelog.xml" relativeToChangelogFile="true"/>
9 <include file="apex/apex.changelog.xml" relativeToChangelogFile="true"/>
10 <include file="1.1.1/release.changelog.xml" relativeToChangelogFile="true"/>
11 <include file="1.2/release_changelog.xml" relativeToChangelogFile="true"/>
12 <include file="post_scripts/post_scripts.changelog.xml" relativeToChangelogFile="true"/>
13
14 <include file="next/release.changelog.xml" relativeToChangelogFile="true"/>
15 </databaseChangeLog>
```

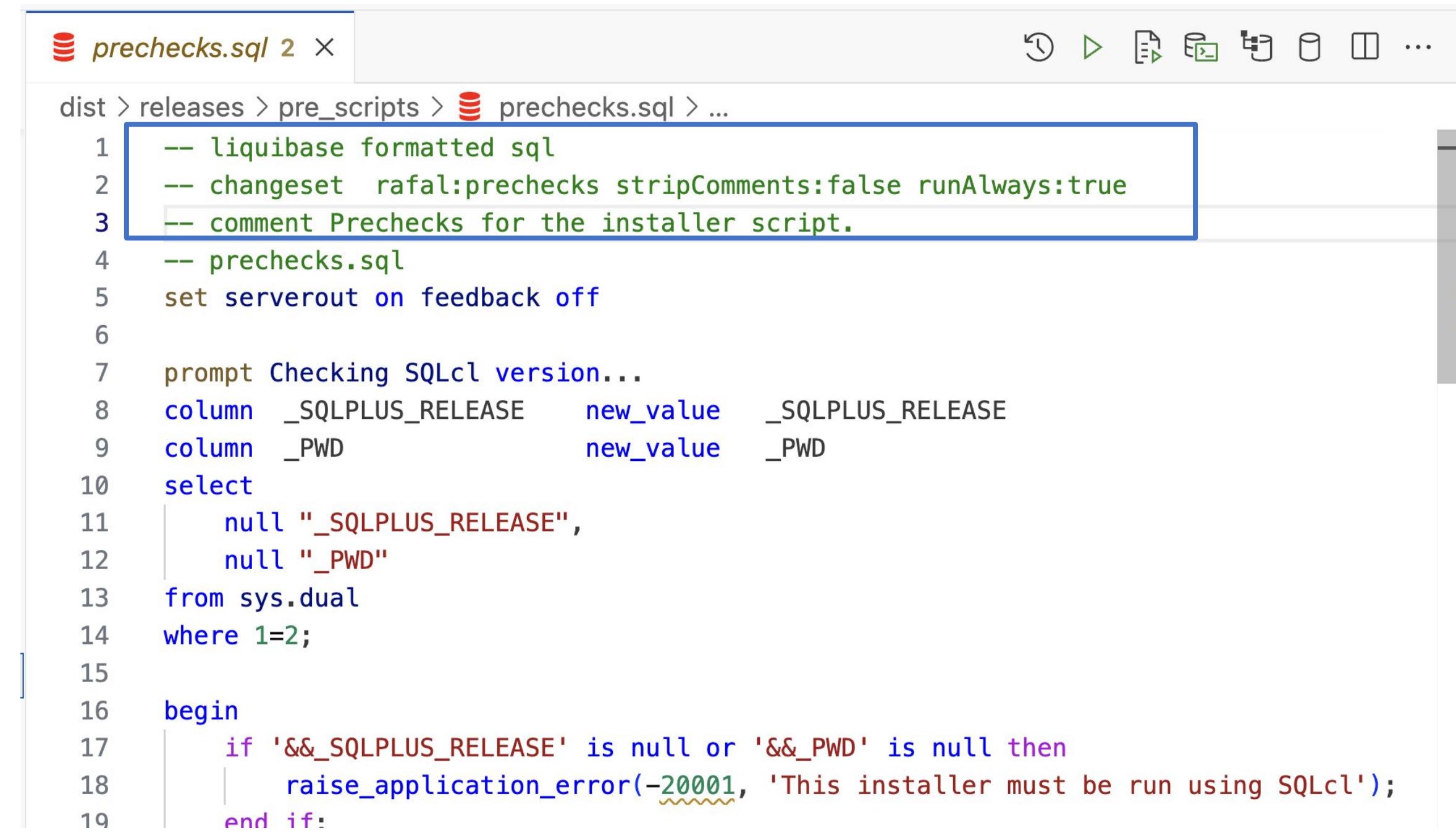
# Want to add repeatable PRE and POST scripts to your deployments?



The screenshot shows the Liquibase IDE interface. The left pane, labeled "EXPLORER", displays a project structure under "BOMBSHELL". It includes a "dist" folder containing "releases" (with subfolders 1.1, 1.1.1, 1.2), "apex", "fix-1.0", "next", "post\_scripts", and a "pre\_scripts" folder containing "pre\_scripts.changelog.xml" and "prechecks.sql". The right pane, labeled "main.changelog.xml" and "pre\_scripts.changelog.xml", shows the XML code for the changelog. The "pre\_scripts.changelog.xml" tab is active, highlighted with a blue border. The XML code is as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
 xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
 http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-4.3.xsd">
 <!--CHANGES-->
 <include file="prechecks.sql" relativeToChangelogFile="true"/>
</databaseChangeLog>
```

# Want to add repeatable PRE and POST scripts to your deployments?



The screenshot shows a code editor window with a tab labeled "prechecks.sql 2 X". The file path is shown as "dist > releases > pre\_scripts > prechecks.sql > ...". The code itself is a SQL script:

```
1 -- liquibase formatted sql
2 -- changeset rafal:prechecks stripComments:false runAlways:true
3 -- comment Prechecks for the installer script.
4 -- prechecks.sql
5 set serverout on feedback off
6
7 prompt Checking SQLcl version...
8 column _SQLPLUS_RELEASE new_value _SQLPLUS_RELEASE
9 column _PWD new_value _PWD
10 select
11 null "_SQLPLUS_RELEASE",
12 null "_PWD"
13 from sys.dual
14 where 1=2;
15
16 begin
17 if '&&_SQLPLUS_RELEASE' is null or '&&_PWD' is null then
18 raise_application_error(-20001, 'This installer must be run using SQLcl');
19 end if;
```

The first three lines of the script are highlighted with a blue box.

# Preview scripts instead of direct deployment? Easy

- Generate a copy of install.sql file and call it e.g. my\_preview\_script.sql
- Change update command to **update-sql**
- Provide a path to preview file



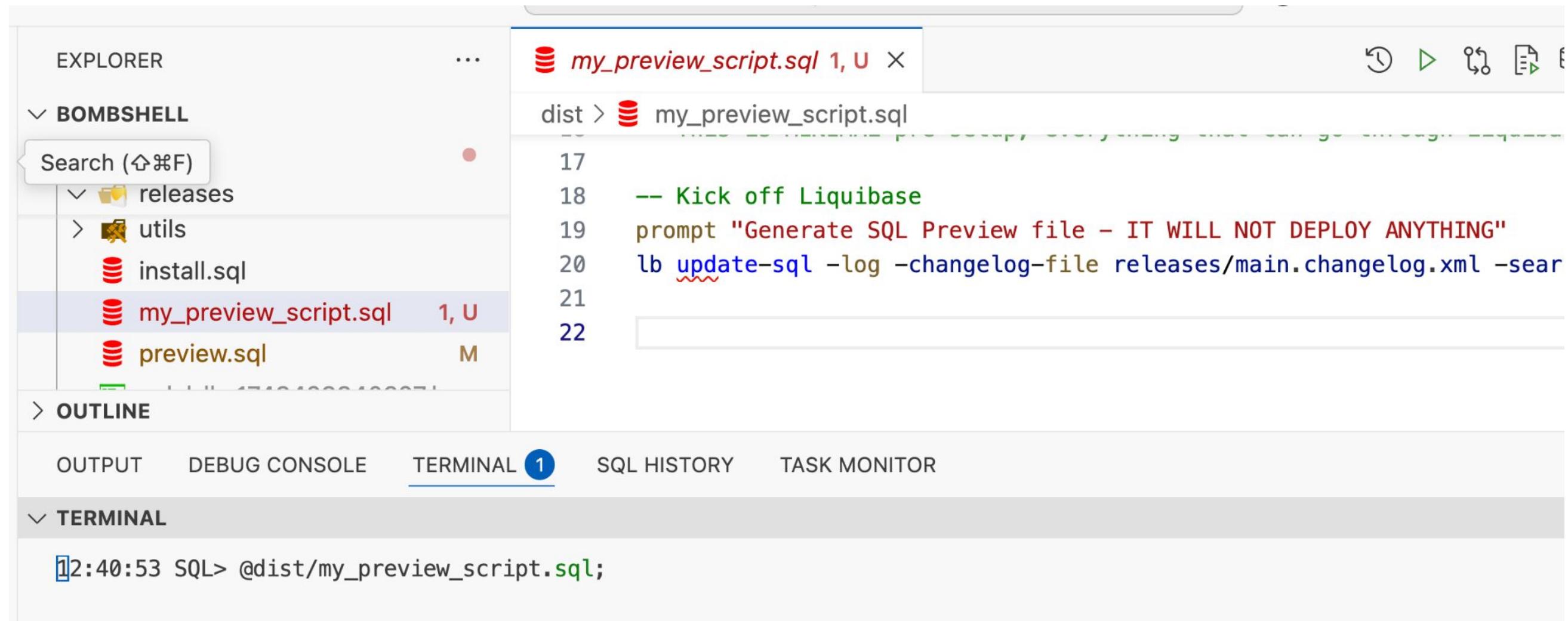
The screenshot shows a code editor window with a tab labeled "my\_preview\_script.sql 1, U X". The file path is "dist > my\_preview\_script.sql". The code content is as follows:

```
17
18 -- Kick off Liquibase
19 prompt "Generate SQL Preview file - IT WILL NOT DEPLOY ANYTHING"
20 lb update-sql -log -changelog-file releases/main.changelog.xml -search-path "." -liquibase-schema-name HR -output-file preview.sql
21
```

Two blue dashed arrows point upwards from the bottom of the slide towards the underlined "update-sql" command in the code. The first arrow is on the left, and the second is on the right.

# Preview scripts instead of direct deployment? Easy

- Connect to target env and execute @dist/my\_preview\_script.sql



The screenshot shows a code editor interface with the following details:

- EXPLORER** panel on the left:

  - BOMBSHELL** folder:
    - Search (F)
    - releases
    - utils
    - install.sql
    - my\_preview\_script.sql** (selected, 1, U)
    - preview.sql

- my\_preview\_script.sql** file content in the main editor:

```
17
18 -- Kick off Liquibase
19 prompt "Generate SQL Preview file - IT WILL NOT DEPLOY ANYTHING"
20 lb update-sql -log -changelog-file releases/main.changelog.xml -sear
21
22
```

- TERMINAL** tab at the bottom is active.
- Terminal output:

```
12:40:53 SQL> @dist/my_preview_script.sql;
```

```

dist > preview.sql > ...
146 -- sqlcl_snapshot src/database/co/tables/orders.sql:4084312c5833d82e2806b031b774a93f
147
148 alter table co.orders modify (
149 | order_status varchar2(20 char)
150);
151
152 update HR.DATABASECHANGELOG_ACTIONS set status = 'RAN' where id = '1748427658649' and
153
154 INSERT INTO HR.DATABASECHANGELOG (ID, AUTHOR, FILENAME, DATEEXECUTED, ORDEREXECUTED,
155
156 -- Changeset releases/post_scripts/recompile.sql::recompile::rafal
157 -- Logging Oracle Liquibase Extension actions to HR.DATABASECHANGELOG_ACTIONS
158 DECLARE
159 id varchar2(2000) := 'post_scripts/recompile';
160 status varchar2(2000) := 'PREPARING';
161 rawAction clob;
162 rawXml clob;
163 myrow varchar2(2000);
164 action clob := '';
165
166 begin
167 insert into HR.DATABASECHANGELOG (ID, AUTHOR, FILENAME, DATEEXECUTED, ORDEREXECUTED, STATUS)
168 values (id, 'rafal', 'post_scripts/recompile.sql', sysdate, 1, status);
169
170 for myrow in (select * from apex_collections where collection_name = 'LIQUIBASE_CHANGELOG') loop
171 rawAction := myrow.collection_value;
172 rawXml := myrow.collection_value;
173 if rawAction is not null then
174 execute immediate rawAction;
175 end if;
176 end loop;
177 end;
178
179 update HR.DATABASECHANGELOG_ACTIONS set status = 'RAN' where id = 'recompile' and status = 'PREPARING';
180
181 commit;
182
183 update HR.DATABASECHANGELOGLOCK SET LOCKED = 0, LOCKEDBY = NULL, LOCKGRANTED = NULL ;
184
185 end;
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413

```

```

dist > preview.sql > ...
1 -- Lock Database
2 UPDATE HR.DATABASECHANGELOGLOCK SET LOCKED = 1, LOCKEDBY = 'MacBook-Pro.local (192.168.1.11)' ;
3
4 --- ****
5 --- Update Database Script
6 --- ****
7 --- Change Log: releases/main.changelog.xml
8 --- Ran at: 5/28/25, 12:40 PM
9 --- Against: ADMIN@jdbc:oracle:thin:@TEST_LOW
10 --- Liquibase version: 4.30.0
11 --- ****
12
13 --- Changeset releases/pre_scripts/prechecks.sql::prechecks::rafal
14 --- Prechecks for the installer script.
15 --- Logging Oracle Liquibase Extension actions to HR.DATABASECHANGELOG_ACTIONS
16 DECLARE
17 id varchar2(2000) := 'prechecks';
18 status varchar2(2000) := 'PREPARING';
19 rawAction clob;
20 rawXml clob;
21 myrow varchar2(2000);
22 action clob := '';

```

## Modify install.sql

- If you deploy to more than one schema - you need to specify schema to store Liquibase tracking table
- The liquibase DATABASECHANGELOG tracking table (automatically created) is an object that holds all of the information about what and when was already deployed



```
bombshell
dist/install.sql

Unstaged Stage
dist/install.sql

@@ -19,3 +19,3 @@ set sqlblanklines on
prompt "Installing/updating schemas"
- lb update -log -changelog-file releases/main.changelog.xml -search-path "."
+ lb update -log -changelog-file releases/main.changelog.xml -search-path "." -liquibase-schema-name HR
```



Have you ever seen developers stepping  
on each others toe in shared database?

- Use LOKI
- free APEX app from Oracle



## User JOHN automatically “locks” his object

```
23:14:14 23:03:55 SQL> connect -name JOHN_HR ←
Connected.
23:14:51 23:03:55 SQL> CREATE OR REPLACE FUNCTION f_sample_function
 2 RETURN VARCHAR2
 3 IS
 4 BEGIN
 5 RETURN 'JOHN created this function!';
 6 END;
7* /

Function F_SAMPLE_FUNCTION compiled
```



RAFAL can't change this object until its locked by JOHN

```
23:14:56 23:03:55 SQL> connect -name RAFAL_HR ←
Connected.
23:15:16 23:03:55 SQL> CREATE OR REPLACE FUNCTION f_sample_function
 2 RETURN VARCHAR2
 3 IS
 4 BEGIN
 5 RETURN 'NO! RAFAL created this function!';
 6 END;
 7* /
CREATE OR REPLACE FUNCTION f_sample_function
*
ERROR at line 1:
ORA-04088: error during execution of trigger 'HR.LOKI_BEFORE_DDL_TGR'
ORA-00604: Error occurred at recursive SQL level 1. Check subsequent errors.
ORA-20001: Loki F_SAMPLE_FUNCTION is locked by JOHN
ORA-06512: at "LOKI.LOKI_LOCK", line 605
ORA-06512: at line 1
ORA-06512: at line 2
```

# LOKI app

Loki

john\_apexdev

Unlock All Transfer All Create

Search...

HR.F\_SAMPLE\_FUNCTION  
FUNCTION

JOHN  
2025/06/26 09:14:55 PM

Total 1

Status

Active (1)

User

JOHN (1)

Schema

HR (1)

The screenshot shows the Loki application interface. At the top, there's a navigation bar with the Loki logo, user information (john\_apexdev), and three buttons: 'Unlock All', 'Transfer All', and 'Create'. Below the navigation is a search bar with the placeholder 'Search...'. The main content area displays a single function card for 'HR.F\_SAMPLE\_FUNCTION' under the 'FUNCTION' category. To the right of the card, details are shown: 'JOHN' created '2025/06/26 09:14:55 PM', and a status indicator 'Active' with a green dot. Below the card, a summary 'Total 1' is displayed. On the left side, there are filter dropdowns for 'Status' (Active (1)), 'User' (JOHN (1)), and 'Schema' (HR (1)). Each filter has a corresponding icon and a three-dot menu button.

## LOKI app

The screenshot shows the Loki application interface. At the top, there is a header with the Loki logo (a green icon of a head with a horned helmet) and the word "Loki". To the right of the logo are three buttons: "Unlock All" (grey), "Transfer All" (grey), and "Create" (black). Below the header is a search bar with the placeholder "Search...". On the left side, there are two filter sections: "Status" with a checkbox for "Active (3)" and "User" with a checkbox for "JOHN (3)". The main content area displays a list of database objects:

| Object Type | Name                 | Owner | Last Modified          | Status |
|-------------|----------------------|-------|------------------------|--------|
| FUNCTION    | HR.F_SAMPLE_FUNCTION | JOHN  | 2025/06/26 09:14:55 PM | Active |
| TABLE       | HR.SAMPLE_TABLE      | JOHN  | 2025/06/26 09:42:32 PM | Active |
| PACKAGE     | HR.SAMPLE_PKG        | JOHN  | 2025/06/26 09:46:11 PM | Active |

The object "HR.SAMPLE\_TABLE" is highlighted with a green rectangular selection box. The entire screenshot has a decorative horizontal bar at the bottom consisting of various colored segments.

## LOKI app

The screenshot shows a list of database objects in the LOKI app:

- HR.F\_SAMPLE\_FUNCTION** FUNCTION
- HR.SAMPLE\_TABLE** TABLE

A context menu is open over the second item, showing options:

- Unlock (highlighted)
- Transfer All
- Create

At the top right of the screen, there is a timestamp: JOHN 2025/06/26 09:14:55 PM.

The screenshot shows a list of database objects in the LOKI app:

- HR.F\_SAMPLE\_FUNCTION** FUNCTION
- HR.SAMPLE\_TABLE** TABLE
- HR.SAMPLE\_PKG** PACKAGE

At the top right of the screen, there is a timestamp: JOHN 2025/06/26 09:14:55 PM.

A red arrow points to the "Unlock All" button in the toolbar, which is part of a larger group of buttons: "Unlock All", "Transfer All", and "Create".

## LOKI app

The screenshot shows the LOKI application interface. At the top, there are three buttons: 'Unlock All' (disabled), 'Transfer All' (disabled), and 'Create'. Below this, a list of three items is displayed:

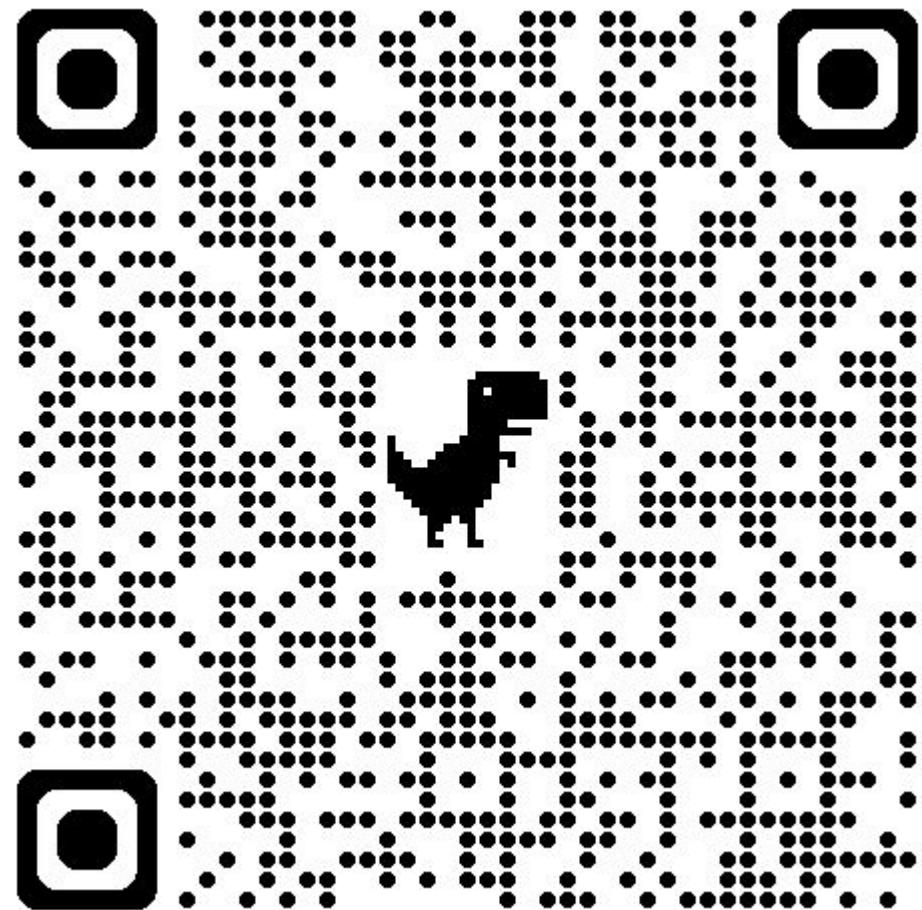
- HR.F\_SAMPLE\_FUNCTION** (FUNCTION)
- HR.SAMPLE\_TABLE** (TABLE)
- HR.SAMPLE\_PKG** (PACKAGE)

A context menu is open over the 'HR.SAMPLE\_PKG' item, listing the following options:

- Unlock
- Transfer
- Show DDL
- Show Details

Red arrows highlight the 'Transfer All' button at the top and the 'Transfer' option in the context menu.

Total 3



## LOKI app

- details
- step by step installation guide

<https://rafal.hashnode.dev/>



Loki

# Project commands (review)

# Project commands

## ❑ Initialize a new project

```
project init -name bombshell -schemas HR, CO
```

## ❑ Export all DB objects

```
project export
```

```
project export -objects HR.COUNTRIES, APEX.101
```

```
project stage add-custom -file-name my_file
```

## ❑ Stage (gen. changelogs/changesets)

```
project stage -branch-name test
```

## ❑ Release

```
project release -version 1.1
```

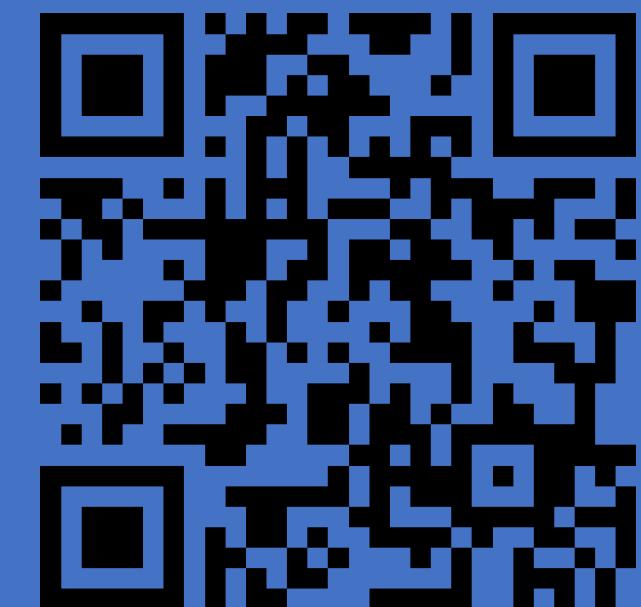
## ❑ Generate artifact (zip)

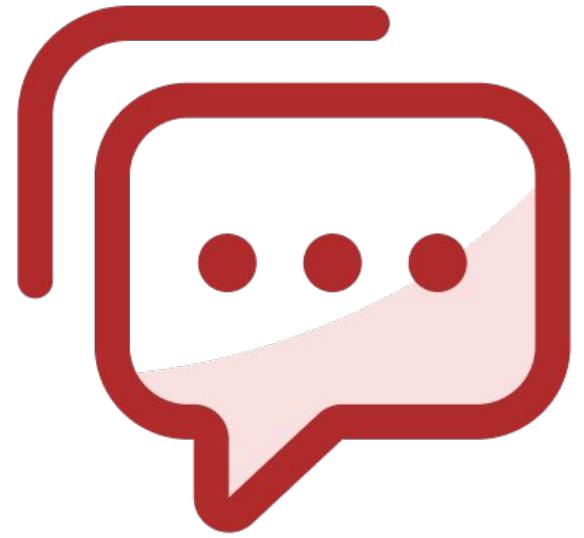
```
project gen-artifact -name test -version 1.1
```

## ❑ Deploy

```
project deploy -file artifact/test-1.1.zip
```

# Questions & Answers

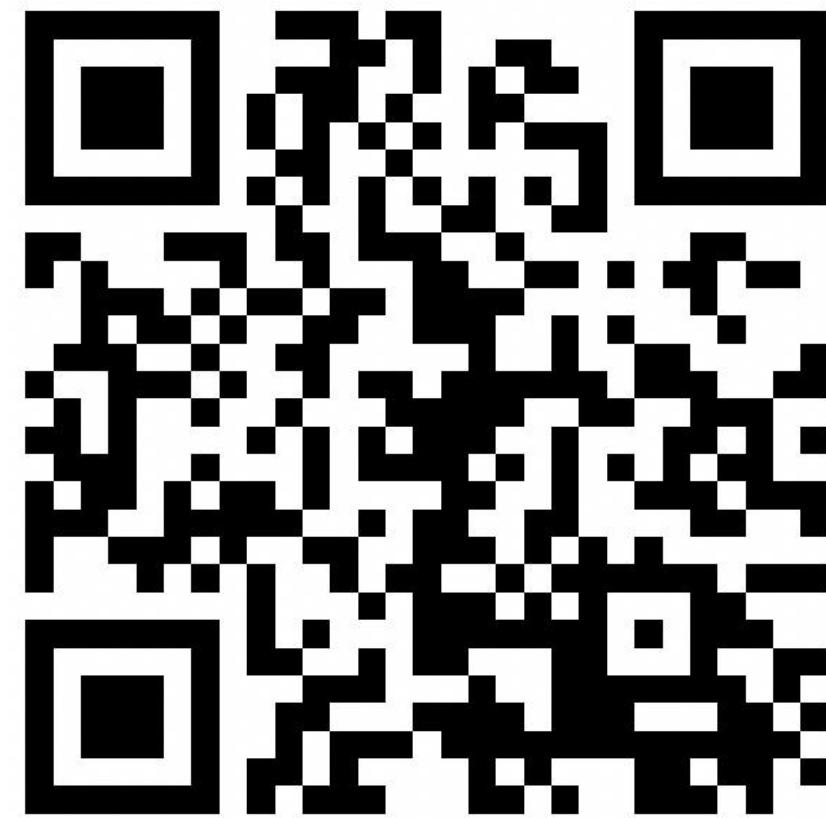




## Audience Q&A



Presenting with animations, GIFs or speaker notes? Enable our [Chrome extension](#)



**GitHub - rgrzegorczyk/conferences: A place  
for public content related to my  
presentations at various conferences**

github.com

<https://github.com/rgrzegorczyk/conferences>



Thank you!  
Feel free to reach me at  
the event or online!



Rafal Grzegorczyk  
[rgrzegorczyk@pretius.com](mailto:rgrzegorczyk@pretius.com)



You can find more at  
[www.pretius.com](http://www.pretius.com)



Partner



# Extra slides( hidden)

# project config

**View current configuration for the project:**

> project config **-list -verbose**

| SETTING NAME         | VALUE       | DEFAULT VALUE | DESCRIPTION                                 |
|----------------------|-------------|---------------|---------------------------------------------|
| project              | bombshell   |               | the project name                            |
| sqlcl.connectionName | dev_admin   |               | Connection name                             |
| sqlcl.version        | 25.1.1.0    |               | SQLcl version                               |
| schemas              | ["HR","CO"] |               | Array of schemas to include in this project |
| lastReleaseVersion   |             |               | The last version released                   |
| export.format.enable | true        | true          | Format code after export                    |

# project config

**View current configuration for the project:**

> project config **-list -verbose**

|                               |                                        |                                   |                                                                                                    |
|-------------------------------|----------------------------------------|-----------------------------------|----------------------------------------------------------------------------------------------------|
| export.apex.exptype           | ["READABLE_YAML","APPLICATION_SOURCE"] | READABLE_YAML, APPLICATION_SOURCE | READABLE_YAML and APPLICATION_SOURCE are the default. See -expType attribute for available options |
| export.apex.expAclAssignments | false                                  | false                             | Export ACL User Role Assignments                                                                   |
| export.apex.expComments       | false                                  | false                             | Export developer comments                                                                          |
| export.apex.skipExportDate    | true                                   | true                              | Skip export date in application export files                                                       |
| export.apex.expirNotif        | false                                  | false                             | Export all interactive report notifications                                                        |

# project config

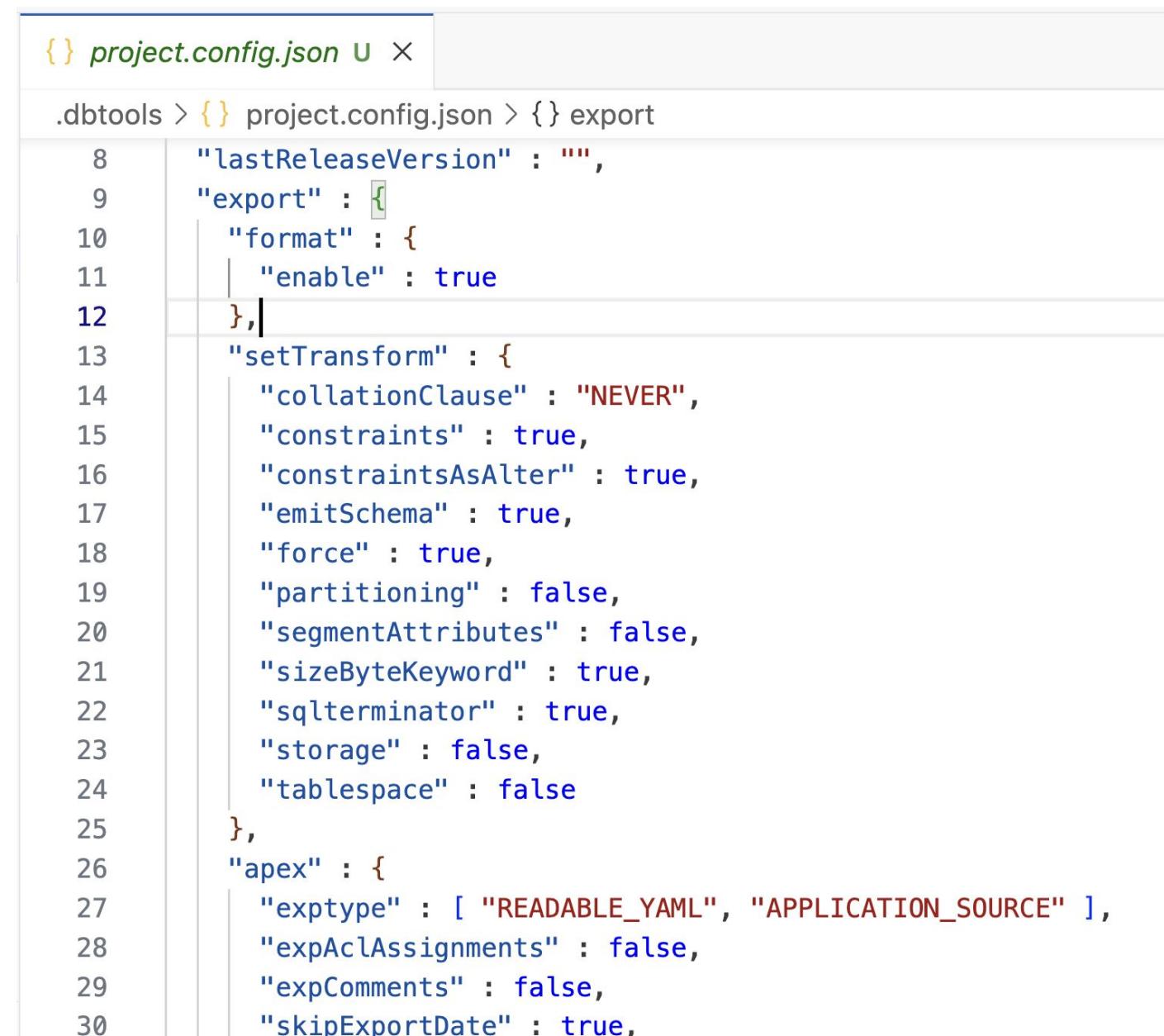
**View current configuration for the project:**

> project config **-list -verbose**

|                                             |       |       |                                                                                            |
|---------------------------------------------|-------|-------|--------------------------------------------------------------------------------------------|
| export.apex.expPubReports                   | false | false | Export all user saved public interactive reports                                           |
| export.apex.expSubscriptions                | false | false | Export references to subscribed components                                                 |
| <u>export.apex.expSupportingObje</u><br>cts | N     | N     | Pass (Y)es, (N)o or (I)nstall to override the default                                      |
| export.apex.expTranslations                 | false | false | Export the translation mappings and all text from the translation repository               |
| export.apex.auditType                       | NULL  | NULL  | Specifies the detail of audit information to include: DATE_S_ONLY, NAMES_AND_DATES or NULL |

# project config - adjust settings

## Manually:



The image shows a code editor window with the file `project.config.json` open. The file contains configuration settings for a database export. The code is written in JSON and includes comments indicating the purpose of various parameters.

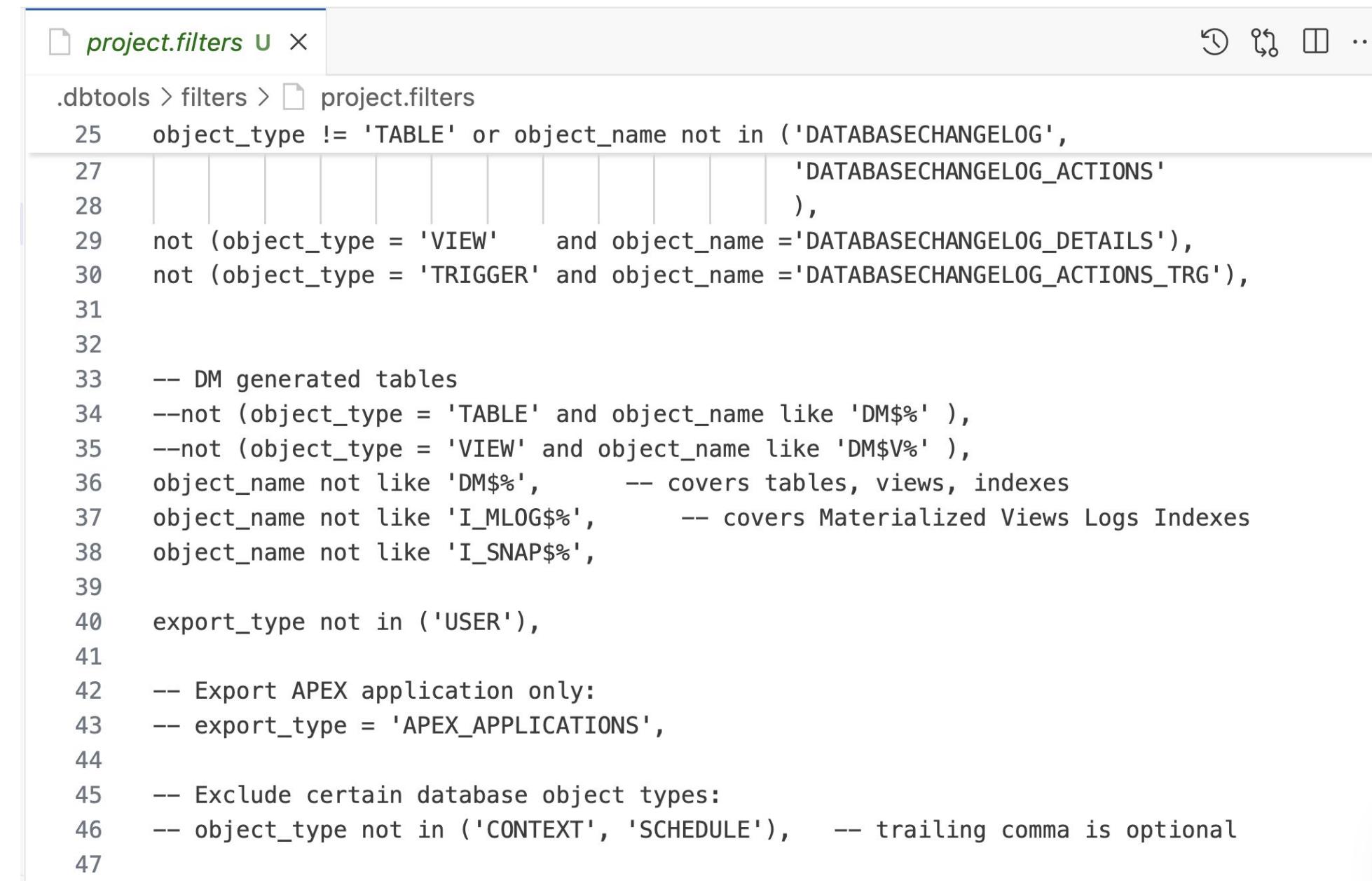
```
{ } project.config.json U X
.dbtools > { } project.config.json > {} export
 8 "lastReleaseVersion": "",
 9 "export": {
10 "format": {
11 "enable": true
12 },
13 "setTransform": {
14 "collationClause": "NEVER",
15 "constraints": true,
16 "constraintsAsAlter": true,
17 "emitSchema": true,
18 "force": true,
19 "partitioning": false,
20 "segmentAttributes": false,
21 "sizeByteKeyword": true,
22 "sqlterminator": true,
23 "storage": false,
24 "tablespace": false
25 },
26 "apex": {
27 "exptype": ["READABLE_YAML", "APPLICATION_SOURCE"],
28 "expAclAssignments": false,
29 "expComments": false,
30 "skipExportDate": true,
31 "skipExportFile": false
32 }
33 }
```

## project config - adjust settings

### Using command line:

```
> project config set -name sqlcl.version -type INTEGER -value 24 -verbose
```

## Edit project.filters to adjust your export:



The image shows a code editor window with the title bar "project.filters" and a toolbar with icons for save, undo, redo, and more. The file path ".dbtools > filters > project.filters" is visible above the code area. The code itself is a series of SQL-like statements with line numbers from 25 to 47. The code defines various filters for database objects based on type and name, including excludes for specific tables like DATABASECHANGELOG and DATABASECHANGELOG\_ACTIONS, and includes for DM generated tables and APEX applications.

```
25 object_type != 'TABLE' or object_name not in ('DATABASECHANGELOG',
26 'DATABASECHANGELOG_ACTIONS'
27),
28 not (object_type = 'VIEW' and object_name = 'DATABASECHANGELOG_DETAILS'),
29 not (object_type = 'TRIGGER' and object_name = 'DATABASECHANGELOG_ACTIONS_TRG'),
30
31
32
33 -- DM generated tables
34 --not (object_type = 'TABLE' and object_name like 'DM$%'),
35 --not (object_type = 'VIEW' and object_name like 'DM$V%'),
36 object_name not like 'DM$%', -- covers tables, views, indexes
37 object_name not like 'I_MLOG$%', -- covers Materialized Views Logs Indexes
38 object_name not like 'I_SNAP$',
39
40 export_type not in ('USER'),
41
42 -- Export APEX application only:
43 -- export_type = 'APEX_APPLICATIONS',
44
45 -- Exclude certain database object types:
46 -- object_type not in ('CONTEXT', 'SCHEDULE'), -- trailing comma is optional
47
```

The screenshot shows the Visual Studio Code (VS Code) interface with a terminal window open. The terminal is titled 'bash' and contains the following command and its output:

```
14:33:42 SQL> project export -objects HR.COUNTRIES,CO.CUSTOMERS,APEX.101
The current connection DEVELOPMENT_HIGH ADMIN will be used for all operations
\
```

The terminal window is located in the 'TERMINAL' tab of the interface. The left sidebar shows the 'EXPLORER' view with a folder icon, and the 'OUTLINE' view with a magnifying glass icon. There are also icons for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'SQL HISTORY', and 'TASK MONITOR'. The bottom status bar shows the user 'viins', the time '1:58', the session 'ADMIN', and the connection 'dev\_admin'. The bottom right corner of the interface includes standard VS Code navigation and search icons.



project export -objects -list

Preview your export:

> project **export -objects** HR.COUNTRIES, CO.CUSTOMERS **-list**

## Preview your export:

```
14:26:04 SQL> project export -objects HR.COUNTRIES,CO.CUSTOMERS -list
The current connection DEVELOPMENT_HIGH ADMIN will be used for all operations
```

```
*** TABLES ***
```

```
CO.CUSTOMERS
```

```
HR.COUNTRIES
```

```
*** INDEXES ***
```

```
CO.CUSTOMERS_NAME_I
```

```
*** COMMENTS ***
```

```
CO.CUSTOMERS
```

```
HR.COUNTRIES
```

```
HR.COUNTRIES
```

```
HR.COUNTRIES
```

```
HR.COUNTRIES
```

```
CO.CUSTOMERS
```

```
CO.CUSTOMERS
```

```
CO.CUSTOMERS
```

```
*** REF_CONSTRAINTS ***
```

```
HR.COUNTR_REG_FK
```

|                |   |
|----------------|---|
| INDEX          | 1 |
| REF_CONSTRAINT | 1 |
| TABLE          | 2 |

Listed 4 objects

# SQLcl will warn you about DROP statements

- HR.JOB.MAX\_SALARY column was dropped

```
10:56:09 SQL> project stage -branch-name test
```

```
Stage is Comparing:
```

|            |                  |
|------------|------------------|
| Old Branch | refs/heads/test  |
| New Branch | refs/heads/1.1.1 |

```
Stage processing completed, please review and commit your changes to repository
```

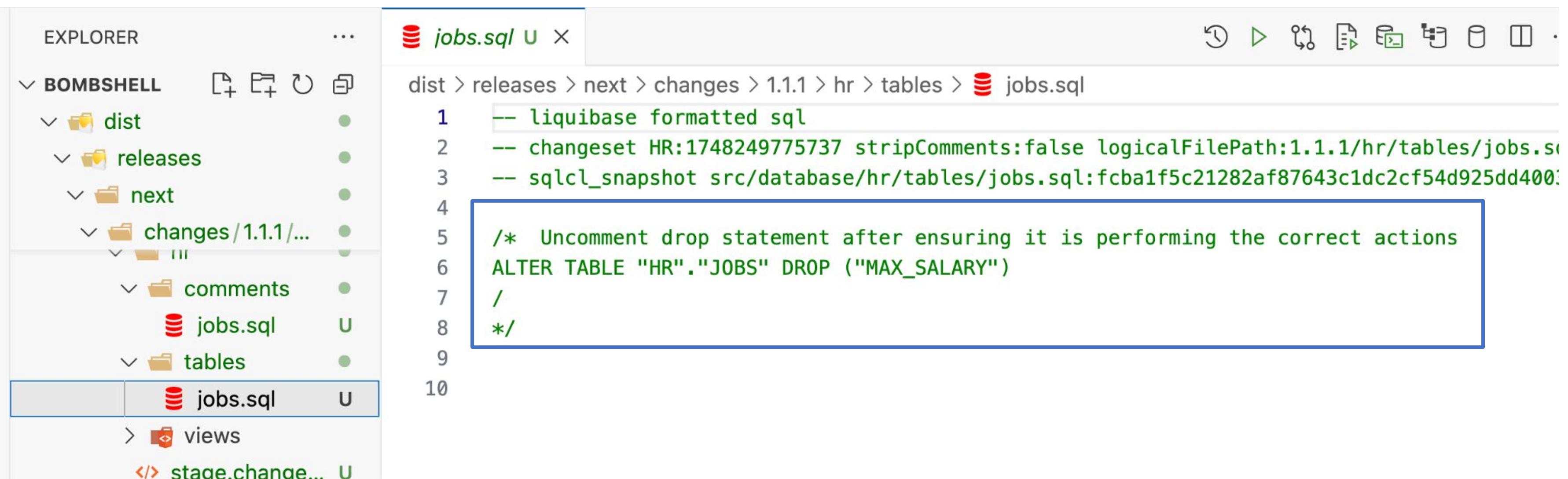
**WARN:** Your changes contain changelogs with one or more DROP statements.

These statements are commented out in the changeset and must be manually uncommented prior to execution.

This is done to ensure no destructive operations happen without notice.

*NOTE: In latest SQLcl 25.1 above warning is not shown. Bug is reported and will be fixed soon*

# SQLcl will warn you about DROP statements



The screenshot shows a code editor interface with an Explorer sidebar on the left and a main editor area on the right.

**EXPLORER** sidebar:

- BOMBSHELL
  - dist
  - releases
  - next
  - changes/1.1.1/
    - jobs.sql
  - comments
  - tables

**Editor Area:**

File path: dist > releases > next > changes > 1.1.1 > hr > tables > jobs.sql

```
1 -- liquibase formatted sql
2 -- changeset HR:1748249775737 stripComments:false logicalFilePath:1.1.1/hr/tables/jobs.s
3 -- sqlcl_snapshot src/database/hr/tables/jobs.sql:fcbaf5c21282af87643c1dc2cf54d925dd400:
4
5 /* Uncomment drop statement after ensuring it is performing the correct actions
6 ALTER TABLE "HR"."JOBS" DROP ("MAX_SALARY")
7 /
8 */
```