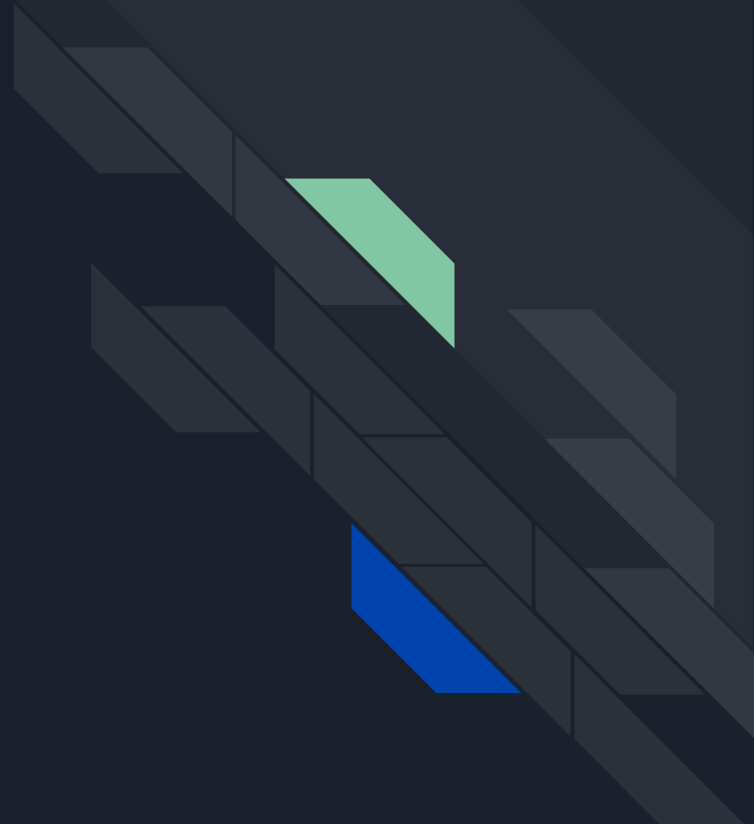


Advanced New ABAP

George Drakos

ABAPConf 5th December 2024





Agenda

- ❖ Introduction
- ❖ What is Advanced New ABAP?
- ❖ How this will benefit your daily tasks and you as a developer respectively
- ❖ Examples/Practical Use Cases
- ❖ Challenges and Considerations
- ❖ Action Time



About me



I'm a SAP Technical Consultant at Teya, a payment solutions company headquartered in London, though I work remotely from Greece. I have around 5 years of experience in SAP development, my expertise spans a wide range of technologies, including ABAP, SAP Fiori, and advanced programming models/frameworks like RAP and Fiori Elements, with a growing focus on CAP.

My linkedin Profile: [George Drakos](#)

Github Repository: [Repository](#)



Advanced New ABAP

More than new ABAP Syntax

- ❖ ‘Advanced New ABAP’ is all about enhancing the capabilities of the ABAP language itself
- ❖ For developers, it offers the opportunity to explore new features that make coding more efficient and powerful
- ❖ With the introduction of modern syntax complex tasks that once required extensive effort can now be simplified
- ❖ These improvements help streamline development, reduce code complexity, and enable developers to focus on delivering innovative solutions faster while leveraging ABAP’s full potential
- ❖ Simplifying complex tasks



Benefits for Daily Tasks

- ❖ Simplified Complex Operations: Enhanced capabilities make handling intricate scenarios easier.
- ❖ Better Code Maintenance: Cleaner, more modular code improves readability and debugging
- ❖ Streamlined Development: Modern syntax and tools reduce repetitive tasks, improving efficiency.



Practical Use Cases

Case 1 Fill Range Directly from Select Statement

```
SELECT @if_fsbp_const_range=>sign_include AS  
sign,  
  
    @if_fsbp_const_range=>option_equal AS option,  
  
    bukr AS low,  
  
    CAST( @space AS CHAR( 4 ) ) AS high  
  
FROM t001  
  
INTO TABLE @DATA(range_of_comp_codes)
```

Case 2 Value Operator using "LINES OF" Addition

```
DATA(t_1) = VALUE stringtab( ( `Line 1` )  
  
    ( `Line 2` ) ).  
  
DATA(t_2) = VALUE stringtab( ( `Line 3` )  
  
    ( `Line 4` ) ).  
  
DATA(t_3) = VALUE stringtab( ( LINES OF t_1 )  
  
    ( LINES OF t_2 FROM 2 )  
  
    ( `Line 5` ) ).
```



Challenges and Considerations

- ❖ Transitioning from classical to modern ABAP [Clean ABAP Repository](#) [Clean ABAP Book](#) [ABAP to the Future](#)
- ❖ Addressing compatibility and learning curves
- ❖ Advanced ABAP not only requires adopting the new syntax but also embracing a modern way of thinking, focusing on modular design and innovative problem-solving approaches.



Let's bring everything we've discussed to life
with some code!