Proudly supported by

# LAB MANUAL

# Lab Manual

# Lab 1- Creating a BTP ABAP Environment

## Create an SAP BTP ABAP Environment Trial User

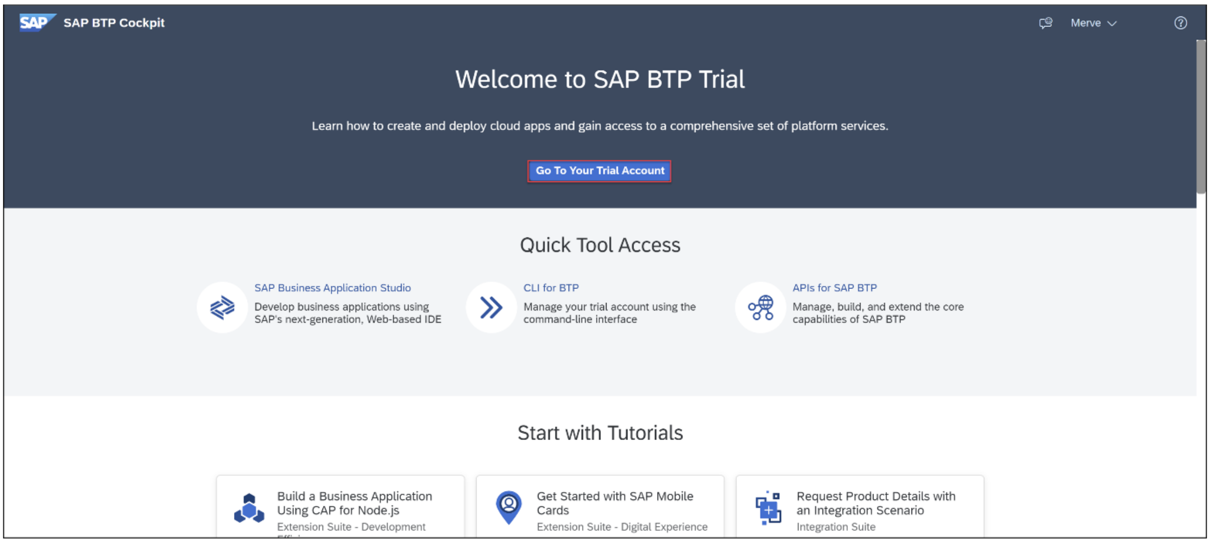
Create a trial user and ABAP cloud project with SAP BTP ABAP environment.

### We will learn

- How to create a trial user
- How to create an ABAP Cloud project

**Step 1:** In your web browser, open the https://cockpit.hanatrial.ondemand.com/.

Navigate to the trial global account by clicking Go to Your Trial Account.



Welcome to SAP BTP Trial

Learn how to create and deploy cloud apps and gain access to a comprehensive set of platform services.

Quick Tool Access

SAP Business Application Studio
Develop business applications using SAP's next-generation, Web-based IDE

CLI for BTP
Manage your trial account using the command-line interface

APIs for SAP BTP
Manage, build, and extend the core capabilities of SAP BTP

Start with Tutorials

Build a Business Application Using CAP for Node.js
Extension Suite - Development

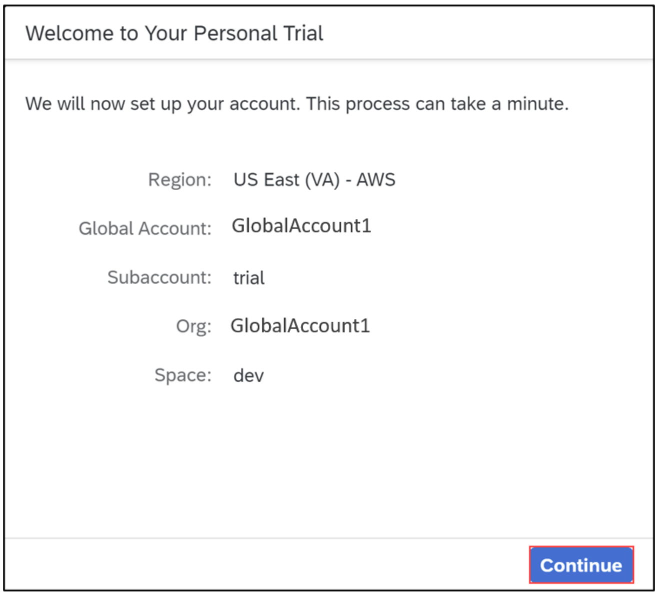
Get Started with SAP Mobile Cards
Extension Suite - Digital Experience

Request Product Details with an Integration Scenario
Integration Suite

If this is your first time accessing your trial account, you'll have to configure your account by choosing a region. Please select US East (VA) as a region. Your user profile will be set up for you automatically.

Wait till your account is set up and ready to go. Your global account, your subaccount, your organization, and your space are launched. This may take a couple of minutes.

Choose Continue.



Welcome to Your Personal Trial

We will now set up your account. This process can take a minute.

Region: US East (VA) - AWS

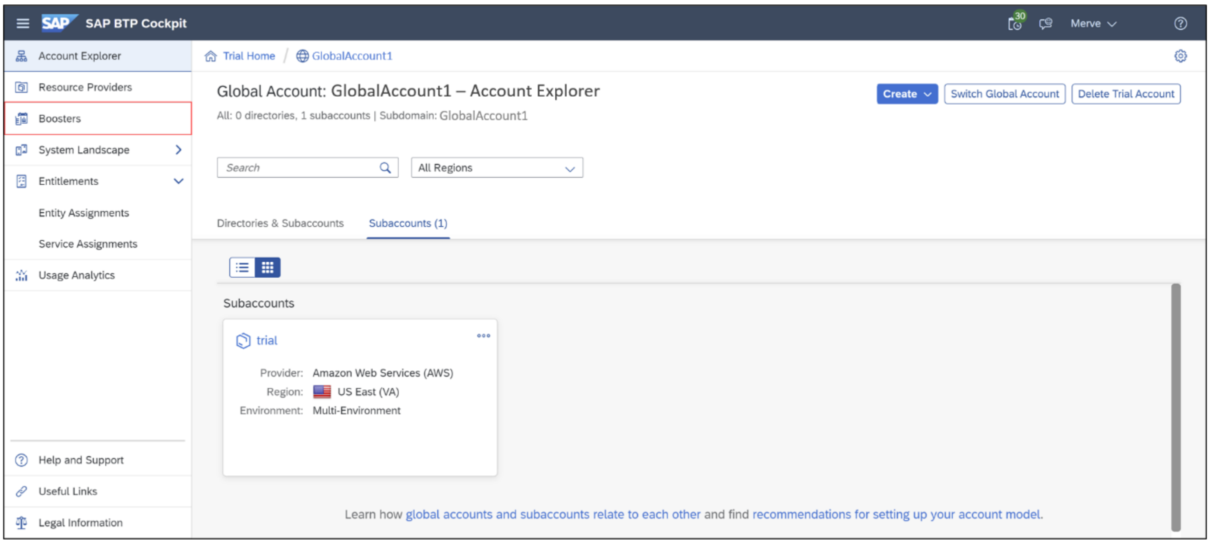
Global Account: GlobalAccount1

Subaccount: trial

Org: GlobalAccount1

Space: dev

From your global account page, choose Boosters on the left side.



Global Account: GlobalAccount1 – Account Explorer

All: 0 directories, 1 subaccounts | Subdomain: GlobalAccount1

Subaccounts

trial

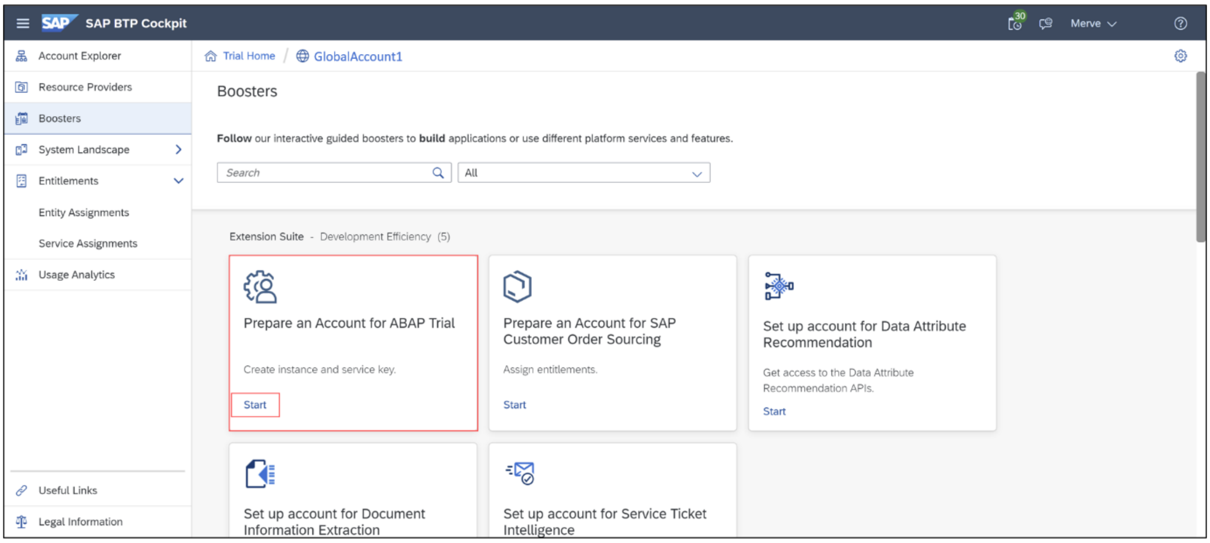
Provider: Amazon Web Services (AWS)

Region: US East (VA)

Environment: Multi-Environment

Learn how global accounts and subaccounts relate to each other and find recommendations for setting up your account model.

Search the Prepare an Account for ABAP Trial tile and press Start to start your booster. If you already created a service instance and service key, then please skip this step and move on with Step 2. Only one service instance can be created at a time.



SAP BTP Cockpit

- Account Explorer
- Resource Providers
- Boosters
- System Landscape
- Entitlements
  - Entity Assignments
  - Service Assignments
- Usage Analytics
- Useful Links
- Legal Information

Trial Home / GlobalAccount1

Boosters

**Follow** our interactive guided boosters to **build** applications or use different platform services and features.

Extension Suite - Development Efficiency (5)

Prepare an Account for ABAP Trial
Create instance and service key.
Start

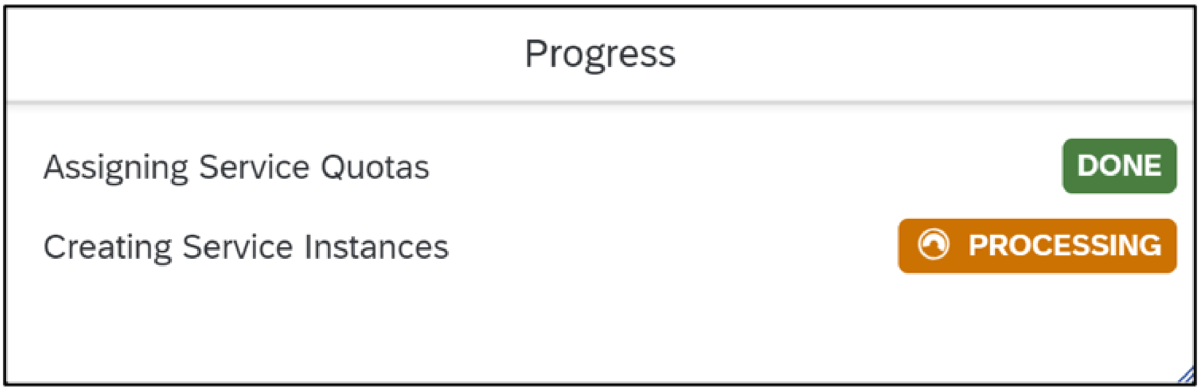
Prepare an Account for SAP Customer Order Sourcing
Assign entitlements.
Start

Set up account for Data Attribute Recommendation
Get access to the Data Attribute Recommendation APIs.
Start

Set up account for Document Information Extraction

Set up account for Service Ticket Intelligence

Now the service instance and service key will be created for the ABAP trial user. The service key can be found inside the service instance.

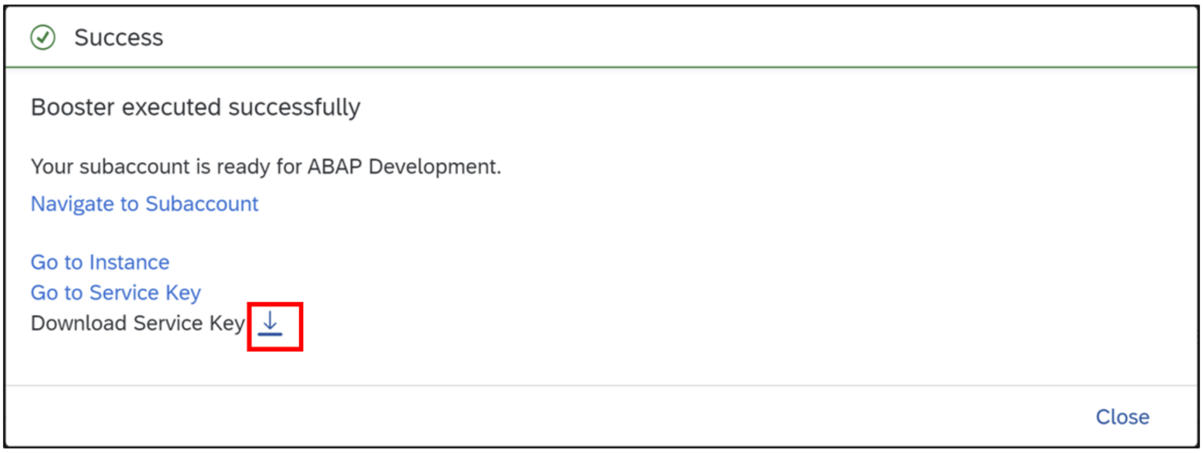


Progress

Assigning Service Quotas DONE

Creating Service Instances PROCESSING

The booster is now executed successfully. Download your service key for later use.



Success

Booster executed successfully

Your subaccount is ready for ABAP Development.

Navigate to Subaccount

Go to Instance

Go to Service Key

Download Service Key

**Step 2:** Open ABAP Development Tools

**Open Eclipse. Make sure you have installed ADT in your Eclipse.**

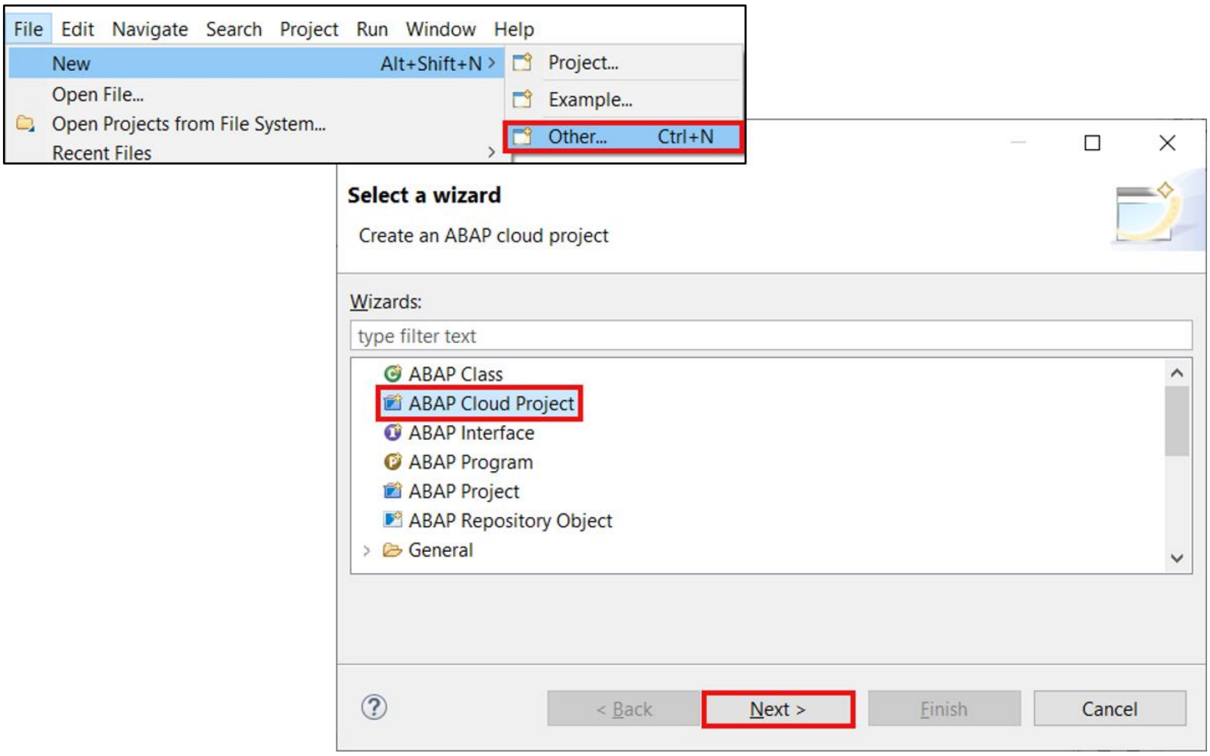
First, we need to install Eclipse IDE to perform the practical’s. We need to follow the following steps to configure Eclipse IDE on local system.

**To install the front-end component of ADT, proceed as follows:**

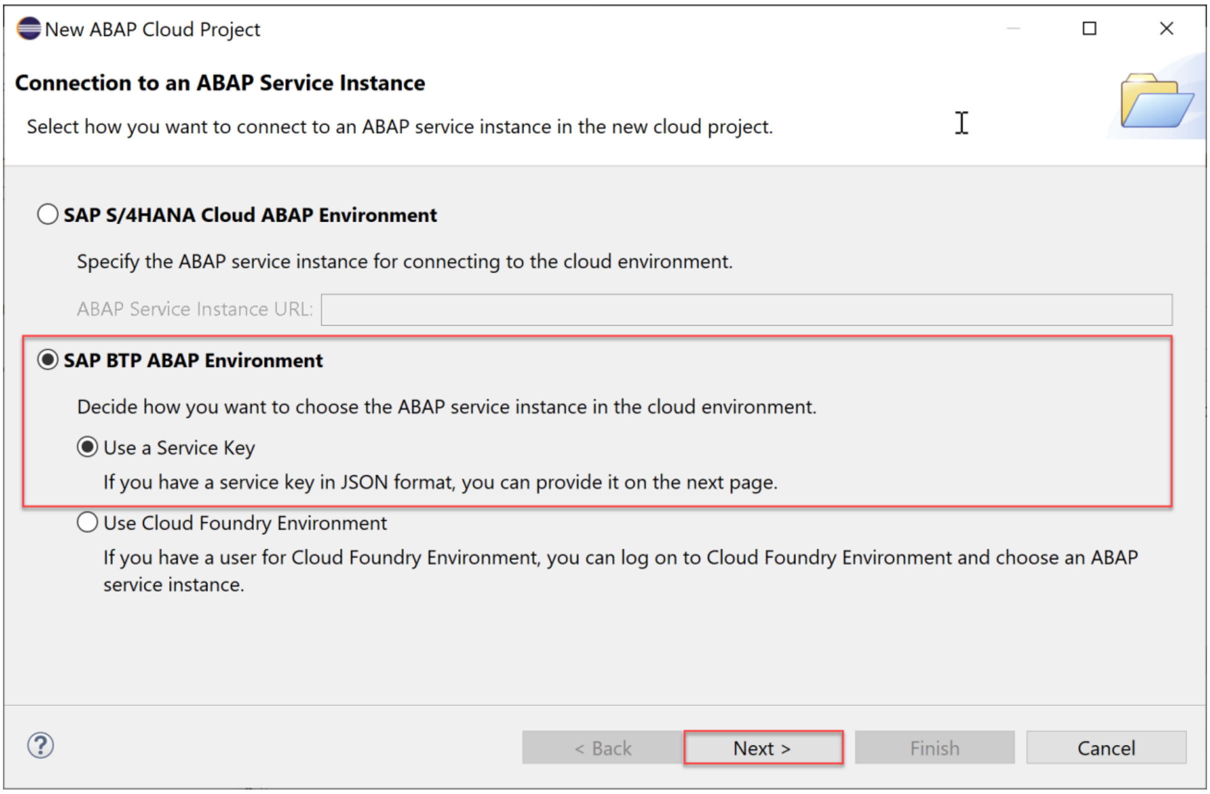
1. Get an installation of Eclipse 2022-09 (x86_64) (Eclipse IDE for Java Developers)
2. In Eclipse, choose in the menu bar Help > Install New Software...
3. Enter the URL https://tools.hana.ondemand.com/latest
4. Press Enter to display the available features.
5. Select ABAP Development Tools and choose next.
6. On the next wizard page, you get an overview of the features to be installed. Choose Next.
7. Confirm the license agreements and choose Finish to start the installation.

**Step 3:** Create ABAP cloud project

1. Select File > New > Other > ABAP Cloud Project and click Next >.



2. Select SAP BTP ABAP Environment > Use a Service Key and click Next >.



New ABAP Cloud Project

**Connection to an ABAP Service Instance**

Select how you want to connect to an ABAP service instance in the new cloud project.

○ **SAP S/4HANA Cloud ABAP Environment**

Specify the ABAP service instance for connecting to the cloud environment.

ABAP Service Instance URL:

● **SAP BTP ABAP Environment**

Decide how you want to choose the ABAP service instance in the cloud environment.

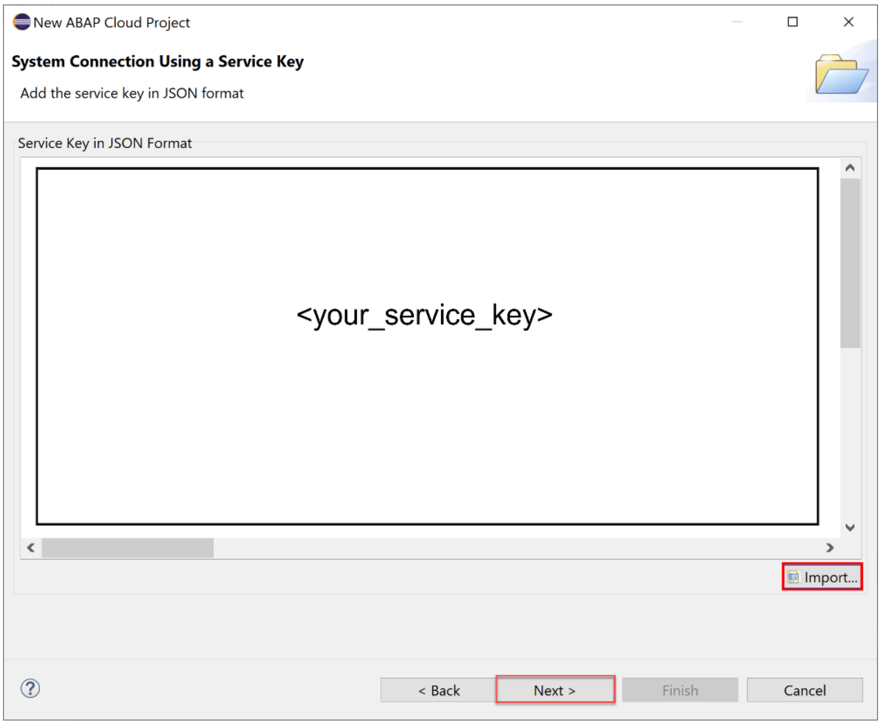
● Use a Service Key

If you have a service key in JSON format, you can provide it on the next page.

○ Use Cloud Foundry Environment

If you have a user for Cloud Foundry Environment, you can log on to Cloud Foundry Environment and choose an ABAP service instance.

3. Import your service key and click Next >.



New ABAP Cloud Project

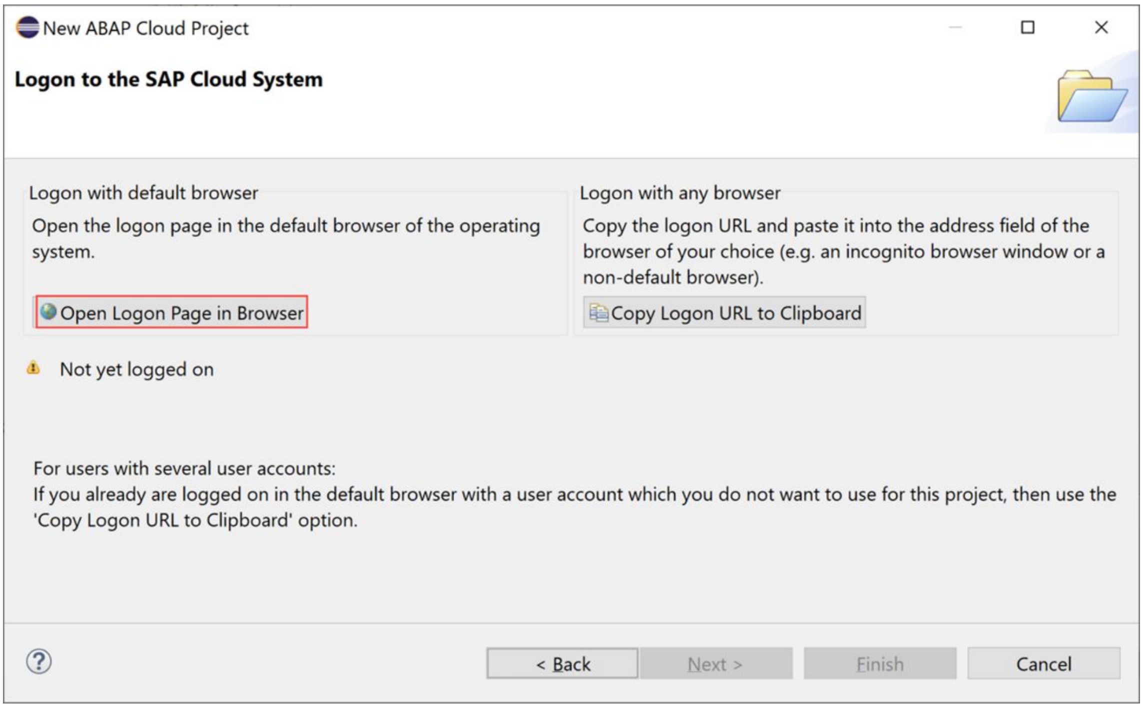
**System Connection Using a Service Key**

Add the service key in JSON format

Service Key in JSON Format

<your_service_key>

4. Click Open Logon Page in Browser.



New ABAP Cloud Project

**Logon to the SAP Cloud System**

Logon with default browser

Open the logon page in the default browser of the operating system.

Open Logon Page in Browser

Logon with any browser

Copy the logon URL and paste it into the address field of the browser of your choice (e.g. an incognito browser window or a non-default browser).

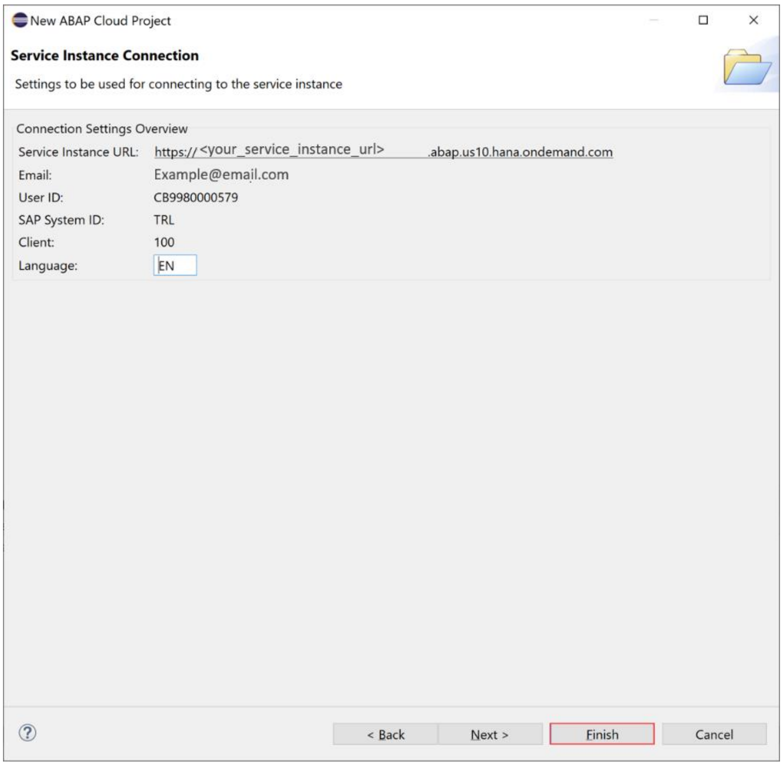
Copy Logon URL to Clipboard

Not yet logged on

For users with several user accounts:
If you already are logged on in the default browser with a user account which you do not want to use for this project, then use the 'Copy Logon URL to Clipboard' option.

< Back | Next > | Finish | Cancel

5. Now you've been authenticated automatically. Provide your credentials if requested. The credentials are the same you used to create your trial account on SAP BTP.



New ABAP Cloud Project

**Service Instance Connection**

Settings to be used for connecting to the service instance

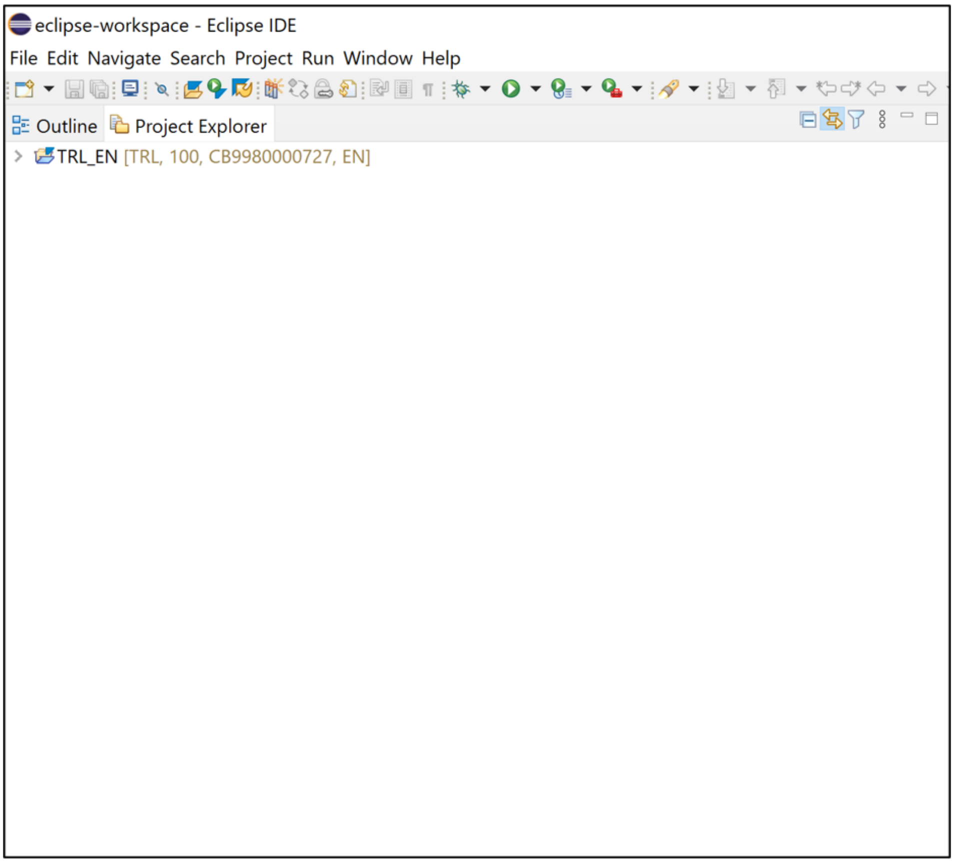
Connection Settings Overview

| | |
|---|---|
| Service Instance URL: | https://<your_service_instance_url> .abap.us10.hana.ondemand.com |
| Email: | Example@email.com |
| User ID: | CB9980000579 |
| SAP System ID: | TRL |
| Client: | 100 |
| Language: | EN |

< Back | Next > | Finish | Cancel

Click Finish.

6. Your trial system appears on the project explorer.



eclipse-workspace - Eclipse IDE

TRL_EN [TRL, 100, CB9980000727, EN]

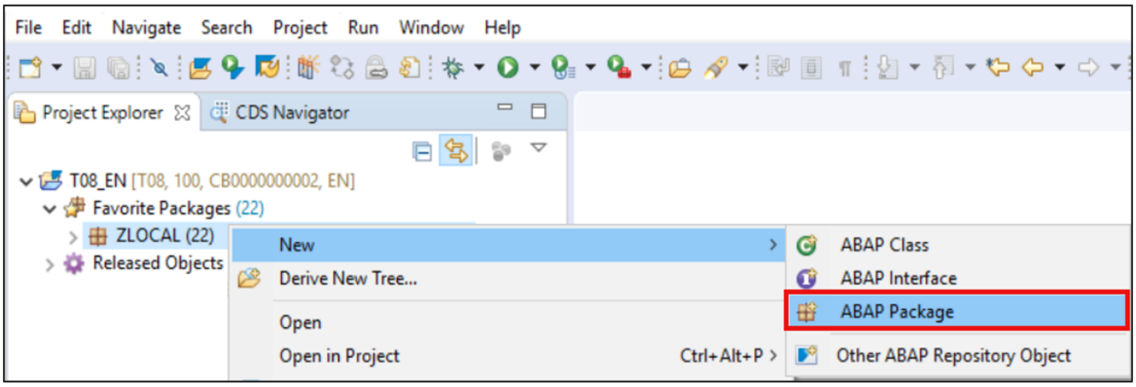
# Lab 2 - Creating an ABAP Package

**In this Lab we will Learn about ABAP Package Creation**

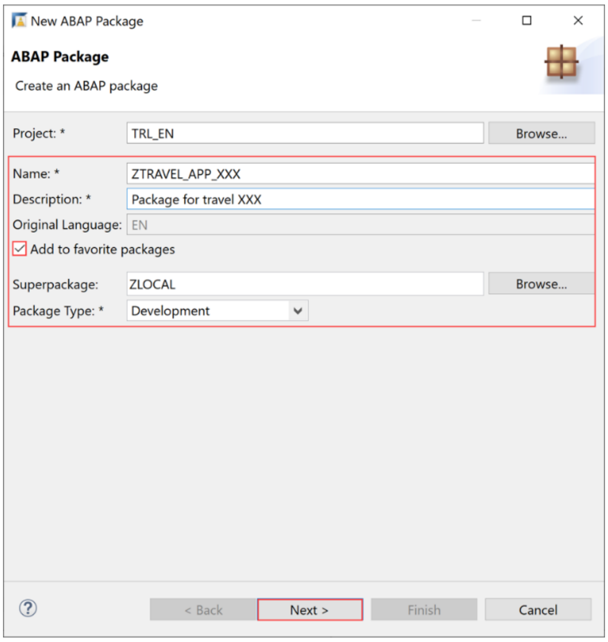
**Note: In this Lab, wherever XXX appears, use a number.**

1. Open ABAP Development Tools (ADT) and select your ABAP Cloud Project you created in Create a SAP BTP ABAP Environment Trial User.

   Right-click on ZLOCAL and select New > ABAP Package.



2. Create a new ABAP package:
   - Name: ZTRAVEL_APP_XXX
   - Description: Package for travel XXX
   - Superpackage: ZLOCAL
   - Check Add to favorite packages.



Project: * TRL_EN

Name: * ZTRAVEL_APP_XXX

Description: * Package for travel XXX

Original Language: EN

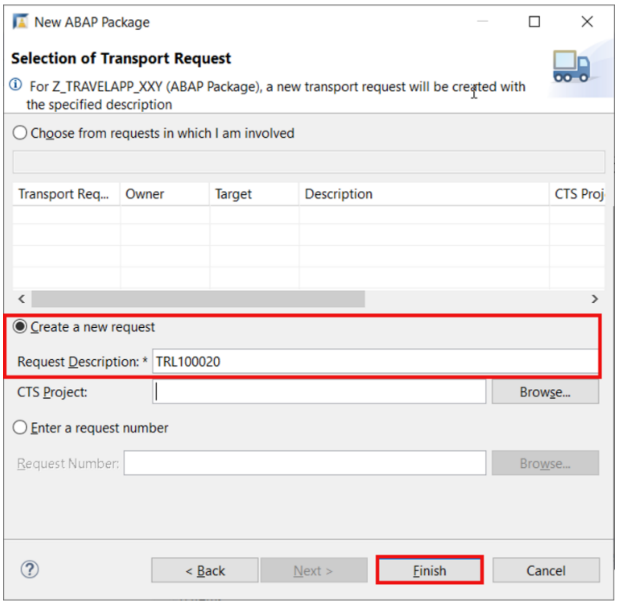
☑ Add to favorite packages

Superpackage: ZLOCAL

Package Type: * Development

Click Next >.

**3.** Select Create new request and enter a request description.



New ABAP Package

**Selection of Transport Request**

For Z_TRAVELAPP_XXY (ABAP Package), a new transport request will be created with the specified description

○ Choose from requests in which I am involved

| Transport Req... | Owner | Target | Description | CTS Proj |
|---|---|---|---|---|

● Create a new request

Request Description: * TRL100020

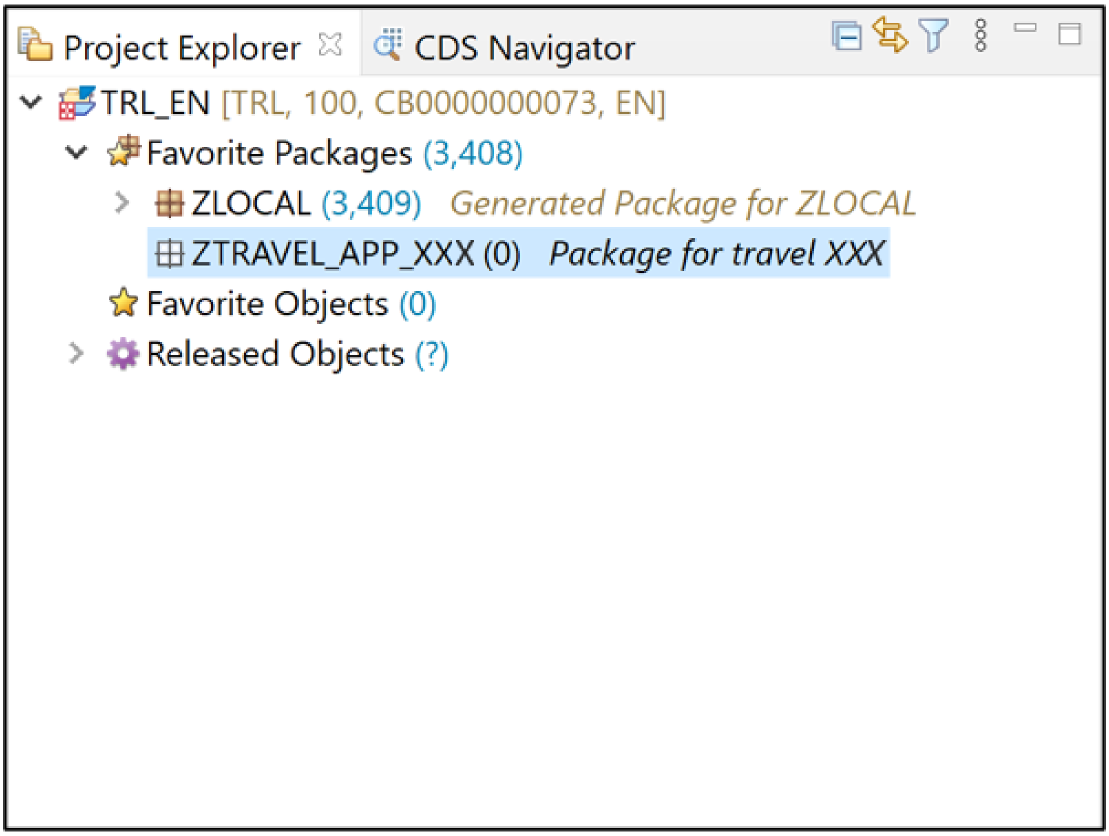
CTS Project:

○ Enter a request number

Request Number:

Click Finish.

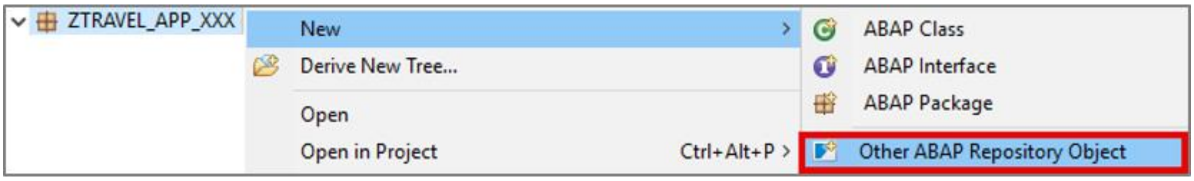
**4.** Now your package is added to favorite objects.



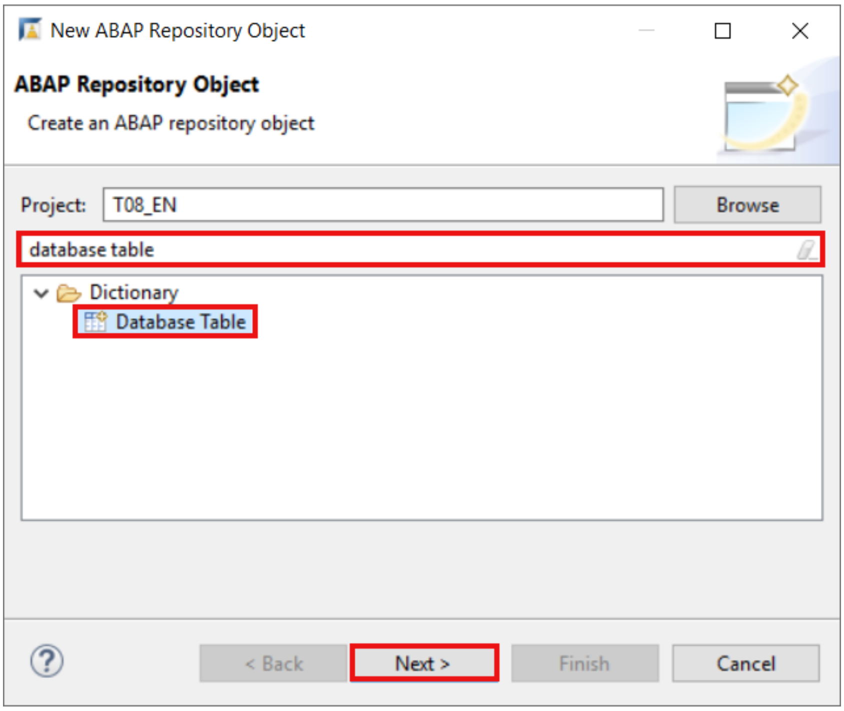
TRL_EN [TRL, 100, CB0000000073, EN]
- Favorite Packages (3,408)
  - ZLOCAL (3,409) *Generated Package for ZLOCAL*
  - ZTRAVEL_APP_XXX (0) *Package for travel XXX*
- Favorite Objects (0)
- Released Objects (?)

# Lab 3 - Creating a Database Table

**1.** Right-click on your package ZTRAVEL_APP_XXX, select New > Other ABAP Repository Object.



**2.** Search for database table, select it and click Next >.



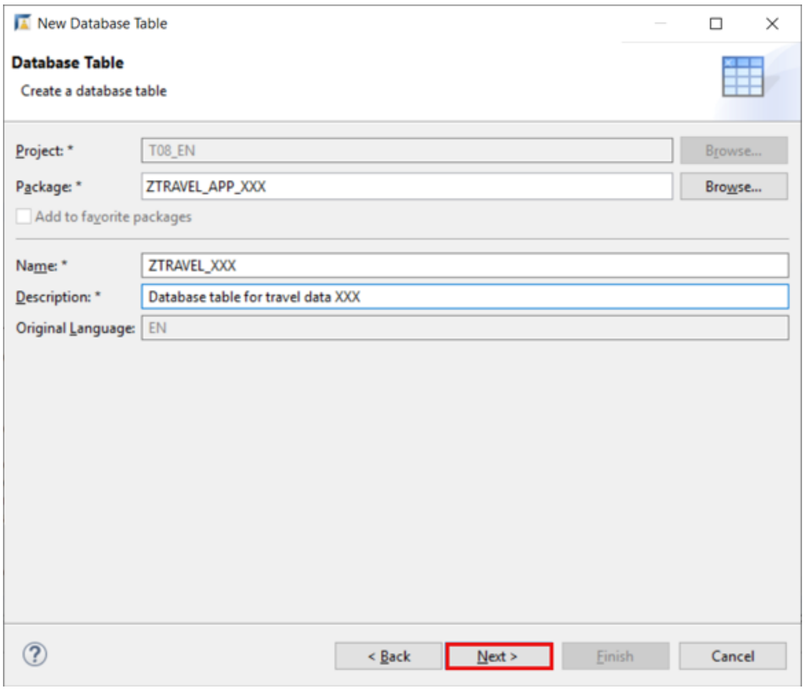
Project: T08_EN

database table

**3.** Create a new database table:

- Name: ZTRAVEL_XXX
- Description: Database table for travel data XXX

Click Next >.



New Database Table

**Database Table**

Create a database table

Project: * T08_EN

Package: * ZTRAVEL_APP_XXX

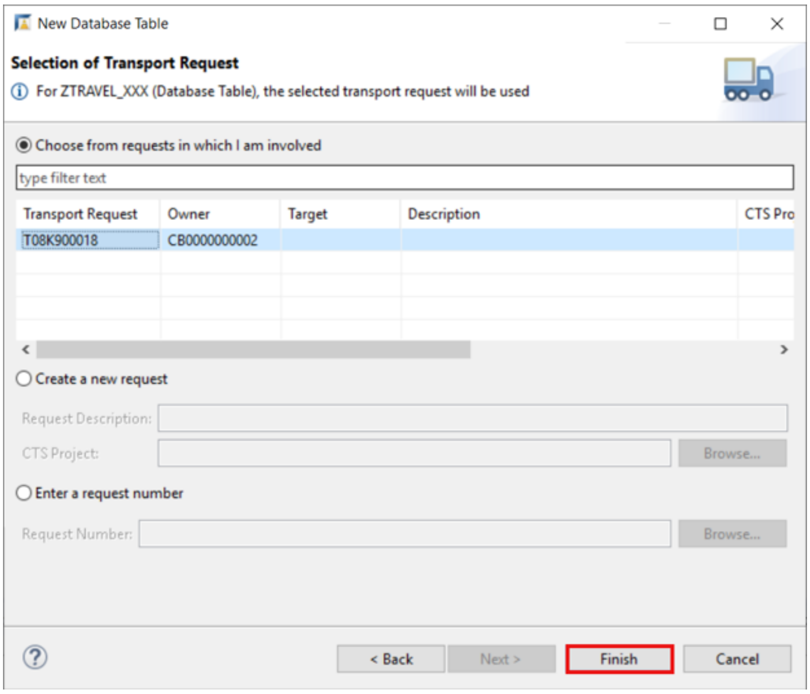
□ Add to favorite packages

Name: * ZTRAVEL_XXX

Description: * Database table for travel data XXX

Original Language: EN

4. Click Finish to create your transport request.



New Database Table

**Selection of Transport Request**

For ZTRAVEL_XXX (Database Table), the selected transport request will be used

● Choose from requests in which I am involved

type filter text

| Transport Request | Owner | Target | Description | CTS Pro |
|---|---|---|---|---|
| T08K900018 | CB0000000002 | | | |

○ Create a new request

Request Description:

CTS Project:

○ Enter a request number

Request Number:

**5.** Replace your code with following:

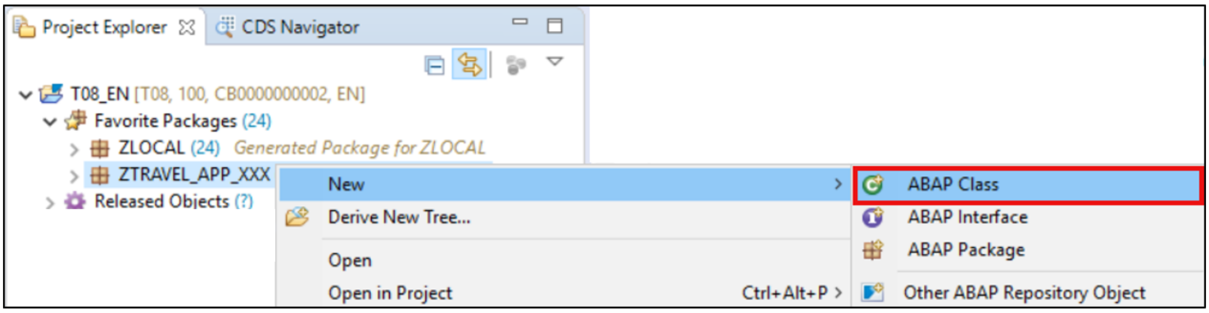
```
@EndUserText.label : 'Database table for travel data XXX'
@AbapCatalog.enhancementCategory : #NOT_EXTENSIBLE
@AbapCatalog.tableCategory : #TRANSPARENT
@AbapCatalog.deliveryClass : #A
@AbapCatalog.dataMaintenance : #RESTRICTED
define table ztravel_xxx {
  key client      : abap.clnt not null;
  key mykey       : sysuuid_x16 not null;
  travel_id       : /dmo/travel_id;
  agency_id       : /dmo/agency_id;
  customer_id     : /dmo/customer_id;
  begin_date      : /dmo/begin_date;
  end_date        : /dmo/end_date;
  @Semantics.amount.currencyCode : 'ztravel_xxx.currency_code'
  booking_fee     : /dmo/booking_fee;
  @Semantics.amount.currencyCode : 'ztravel_xxx.currency_code'
  total_price     : /dmo/total_price;
  currency_code   : /dmo/currency_code;
  description     : /dmo/description;
  overall_status  : /dmo/overall_status;
  created_by      : syuname;
  created_at      : timestampl;
  last_changed_by : syuname;
  last_changed_at : timestampl;
}
```

**6.** Save and Activate

- Save – Ctrl + S
- Activate – Ctrl+F3

# Lab 4 - Create an ABAP Class

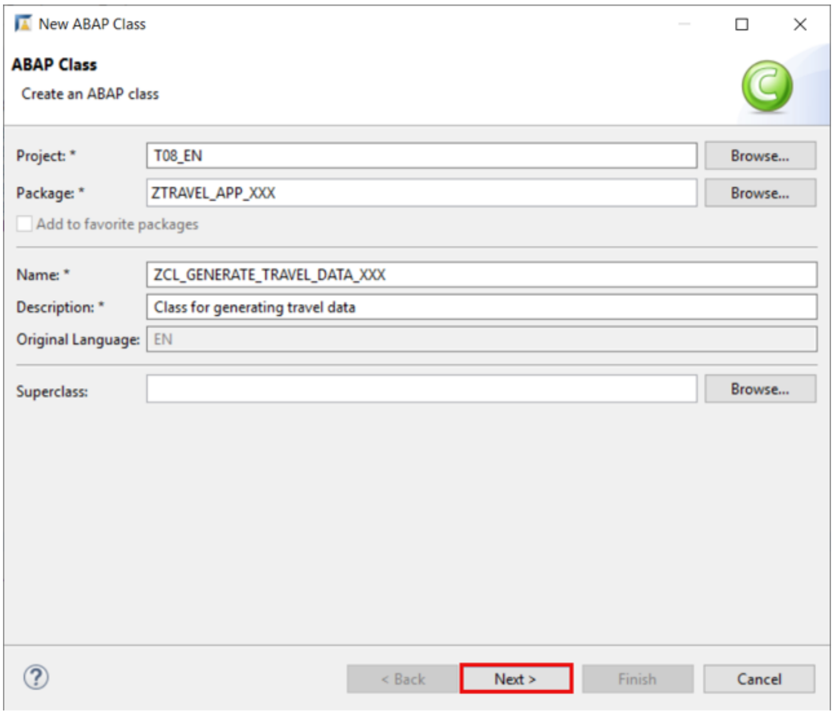
**1.** Right-click on your package ZTRAVEL_APP_XXX, select New > ABAP Class.



**2.** Create a new ABAP class:

- Name: ZCL_GENERATE_TRAVEL_DATA_XXX
- Description: Class for generating travel data

Click Next >.


New ABAP Class

ABAP Class

Create an ABAP class

Project: * T08_EN

Package: * ZTRAVEL_APP_XXX

□ Add to favorite packages

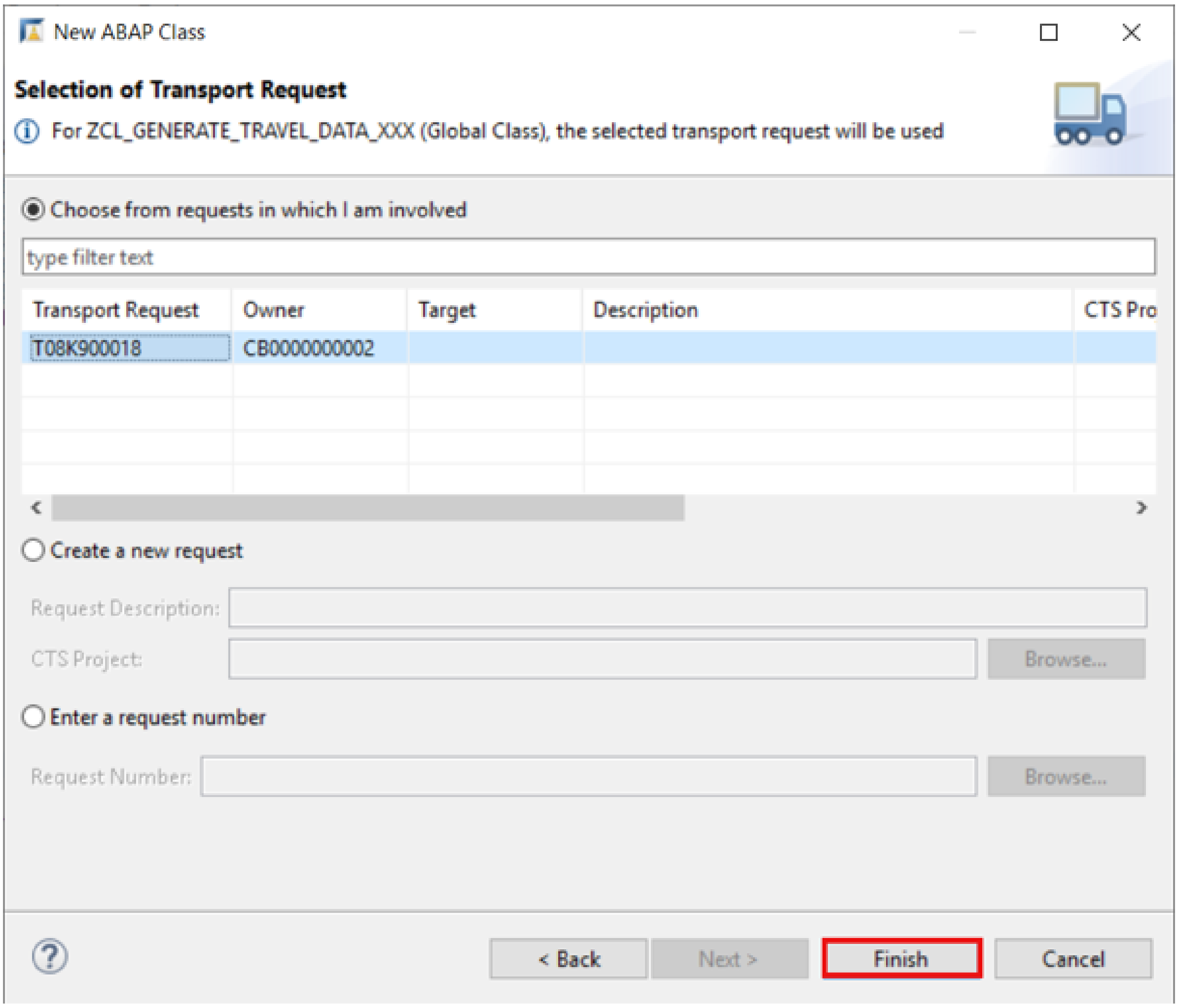
Name: * ZCL_GENERATE_TRAVEL_DATA_XXX

Description: * Class for generating travel data

Original Language: EN

Superclass:

**3.** Click Finish to create your transport request.



New ABAP Class

**Selection of Transport Request**

For ZCL_GENERATE_TRAVEL_DATA_XXX (Global Class), the selected transport request will be used

● Choose from requests in which I am involved

type filter text

| Transport Request | Owner | Target | Description | CTS Pro |
|---|---|---|---|---|
| T08K900018 | CB0000000002 | | | |

○ Create a new request

Request Description:

CTS Project:

○ Enter a request number

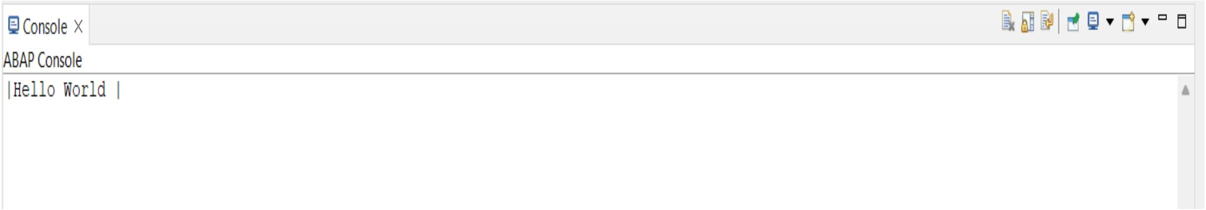
Request Number:

# Lab 5 - WAP to print Hello World

**Create a New Class "zcl_generate_travel_data_yyy" and paste the following code.**

```
CLASS zcl_generate_travel_data_yyy DEFINITION
  PUBLIC
  FINAL
  CREATE PUBLIC .

  PUBLIC SECTION.
    INTERFACES if_oo_adt_classrun.
  PROTECTED SECTION.
  PRIVATE SECTION.
ENDCLASS.



CLASS zcl_generate_travel_data_yyy IMPLEMENTATION.
    METHOD if_oo_adt_classrun~main.


*    output the result as a console message.

    out->write( '|Hello World |').
    ENDMETHOD.


ENDCLASS.
```

**Output:**

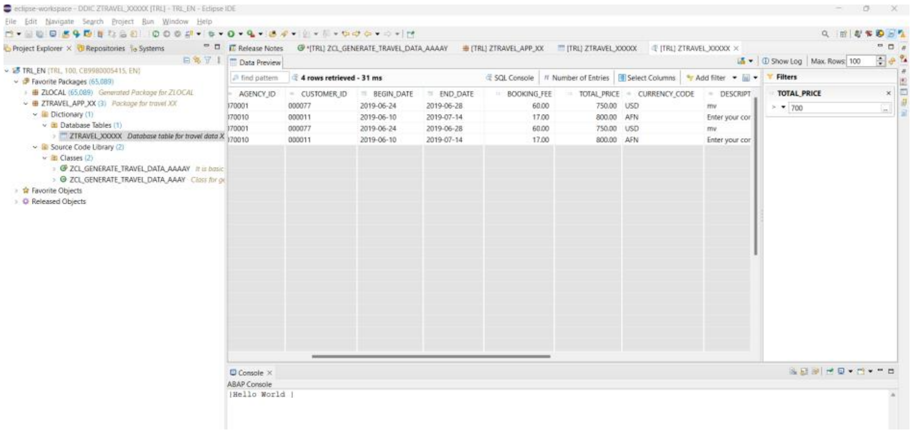


```
|Hello World |
```

# Lab 6 - Perform Query Operation on the Output Table

## 1. How to extract the records having Total Price > 700

Click on “Add Filter”. Click on Total Price and write the condition



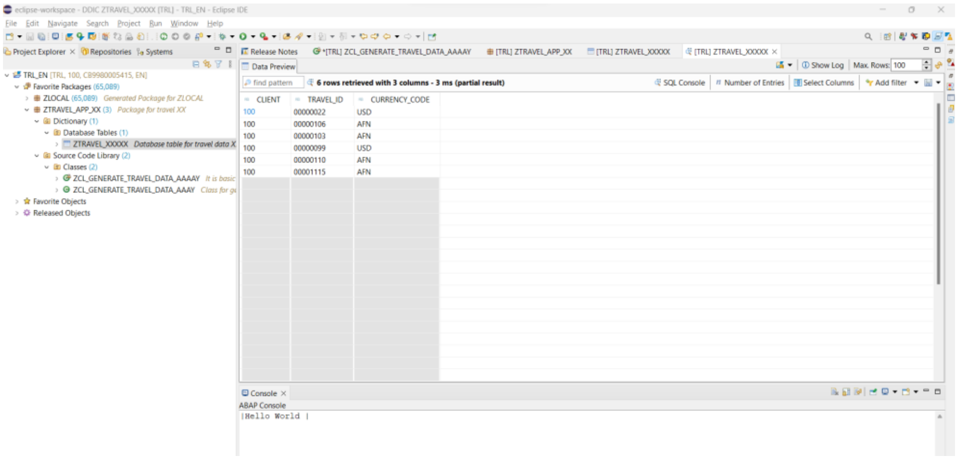
4 rows retrieved - 31 ms

Filters: TOTAL_PRICE > 700

| AGENCY_ID | CUSTOMER_ID | BEGIN_DATE | END_DATE | BOOKING_FEE | TOTAL_PRICE | CURRENCY_CODE | DESCRIPT |
|---|---|---|---|---|---|---|---|
| 170001 | 000077 | 2019-06-24 | 2019-06-28 | 60.00 | 750.00 | USD | mv |
| 170010 | 000011 | 2019-06-10 | 2019-07-14 | 17.00 | 800.00 | AFN | Enter your cor |
| 170001 | 000077 | 2019-06-24 | 2019-06-28 | 60.00 | 750.00 | USD | mv |
| 170010 | 000011 | 2019-06-10 | 2019-07-14 | 17.00 | 800.00 | AFN | Enter your cor |

```
Hello World
```

## 2. Click on Select Columns and Select the columns you want to display

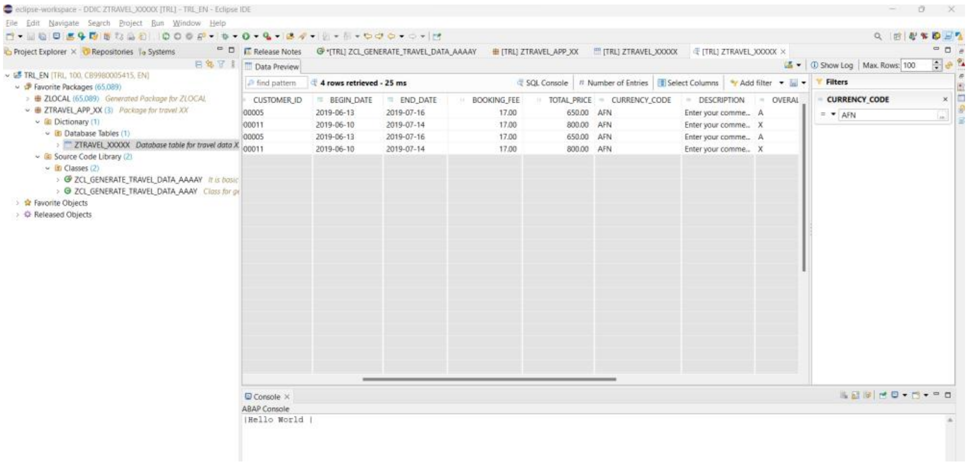


6 rows retrieved with 3 columns - 3 ms (partial result)

| CLIENT | TRAVEL_ID | CURRENCY_CODE |
|---|---|---|
| 100 | 00000022 | USD |
| 100 | 00000106 | AFN |
| 100 | 00000103 | AFN |
| 100 | 00000099 | USD |
| 100 | 00000110 | AFN |
| 100 | 00001115 | AFN |

```
Hello World
```

## 3. Filter the Country Code = ‘AFN’ from the data



4 rows retrieved - 25 ms

Filters: CURRENCY_CODE = AFN

| CUSTOMER_ID | BEGIN_DATE | END_DATE | BOOKING_FEE | TOTAL_PRICE | CURRENCY_CODE | DESCRIPTION | OVERAL |
|---|---|---|---|---|---|---|---|
| 00005 | 2019-06-13 | 2019-07-16 | 17.00 | 650.00 | AFN | Enter your comme... | A |
| 00011 | 2019-06-10 | 2019-07-14 | 17.00 | 800.00 | AFN | Enter your comme... | X |
| 00005 | 2019-06-13 | 2019-07-16 | 17.00 | 650.00 | AFN | Enter your comme... | A |
| 00011 | 2019-06-10 | 2019-07-14 | 17.00 | 800.00 | AFN | Enter your comme... | X |

```
Hello World
```

# Lab 7 - Perform various ABAP data types of operation

**Numeric Operations:**

**Addition:**

```
DATA lv_num1 TYPE i VALUE 10.
DATA lv_num2 TYPE i VALUE 5.
DATA lv_sum TYPE i.

lv_sum = lv_num1 + lv_num2.
WRITE: 'Sum:', lv_sum.
```

**Subtraction:**

```
DATA lv_num1 TYPE i VALUE 10.
DATA lv_num2 TYPE i VALUE 5.
DATA lv_diff TYPE i.

lv_diff = lv_num1 - lv_num2.
WRITE: 'Difference:', lv_diff.
```

**Multiplication:**

```
DATA lv_num1 TYPE i VALUE 10.
DATA lv_num2 TYPE i VALUE 5.
DATA lv_product TYPE i.

lv_product = lv_num1 * lv_num2.
WRITE: 'Product:', lv_product.
```

**Division:**

```
DATA lv_num1 TYPE f VALUE 10.0.
DATA lv_num2 TYPE f VALUE 5.0.
DATA lv_quotient TYPE f.

lv_quotient = lv_num1 / lv_num2.
WRITE: 'Quotient:', lv_quotient.
```

**2-Character/String Operations:**

**Concatenation:**

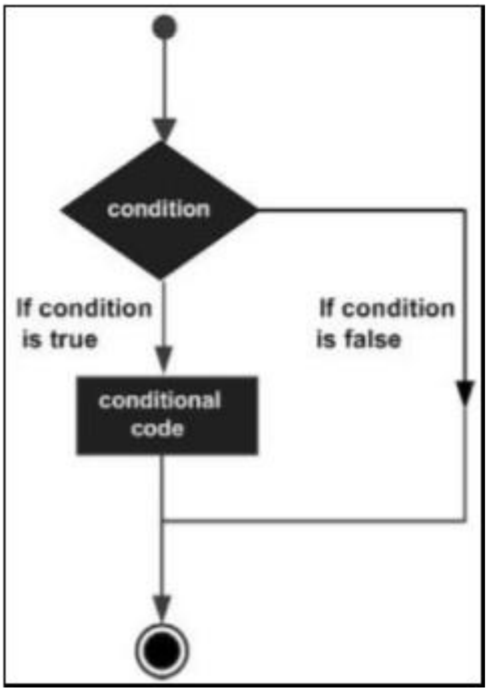
```
DATA lv_str1 TYPE string VALUE 'Hello'.
DATA lv_str2 TYPE string VALUE 'World'.
DATA lv_result TYPE string.

CONCATENATE lv_str1 lv_str2 INTO lv_result.
WRITE: 'Result:', lv_result.
```

**String Length:**

```
DATA lv_str TYPE string VALUE 'Hello World'.
DATA lv_length TYPE i.

lv_length = STRLEN( lv_str ).
WRITE: 'Length:', lv_length.
```

## ABAP programming language for decision-making statements.

the general form of a typical decision-making structure found in most of the programming languages −



| S.No. | Statement & Description |
|---|---|
| 1 | IF Statement<br>An IF statement consists of a logical expression followed by one or more statements. |
| 2 | IF.. Else Statement<br>An IF statement can be followed by an optional ELSE statement that executes when the expression is false. |
| 3 | Nested IF Statement<br>You may use one IF or ELSEIF statement inside another IF or ELSEIF statement. |
| 4 | CASE Control Statement<br>CASE statement is used when we need to compare two or more fields or variables. |

**Example**

```
DATA lv_age TYPE i.

lv_age = 25.


IF lv_age >= 18.

  WRITE: 'You are an adult.'.
```

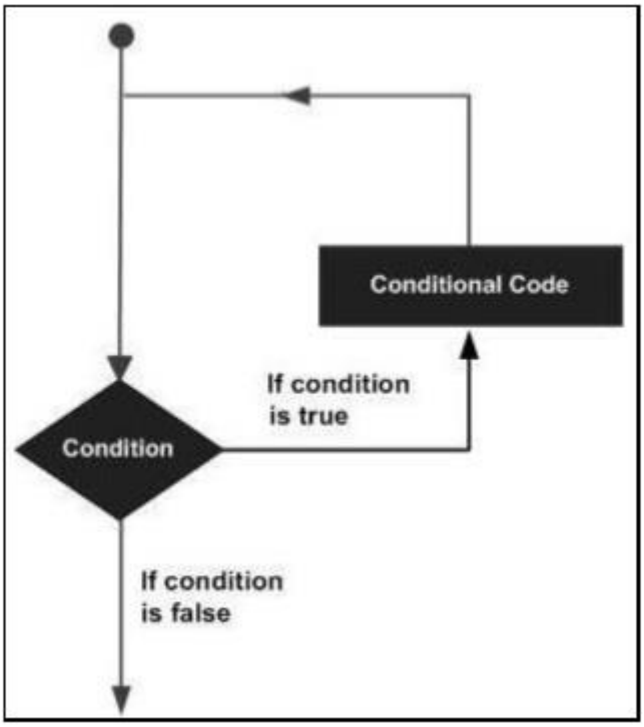
```
ELSE.
  WRITE: 'You are not an adult.'.
ENDIF.
```

In this example, we have a variable lv_age assigned a value of 25. The IF statement checks whether lv_age is greater than or equal to 18. If the condition is true, it executes the statements within the IF block and displays "You are an adult." Otherwise, it executes the statements within the ELSE block and displays "You are not an adult."

**The output will be:** ---- You are an adult.

## Lab 8 - Loops in ABAP, For loop, While Loop

Programming languages provide various control structures that allow for more complicated execution paths. A **loop statement** allows us to execute a statement or group of statements multiple times and following is the general form of a loop statement in most of the programming languages.



**Code For Loop Example:**

```
DATA: lv_counter TYPE i.

lv_counter = 1.


DO 5 TIMES.

  WRITE: 'Counter:', lv_counter.

  lv_counter = lv_counter + 1.

ENDDO.
```

**Explanation:** In this example, we have a variable lv_counter initialized with a value of 1. The DO loop is executed 5 times. Within the loop, the current value of lv_counter is displayed using the WRITE statement. After each iteration, the value of lv_counter is incremented by 1 using the assignment statement lv_counter = lv_counter + 1.

**The output will be:**

```
Counter: 1

Counter: 2

Counter: 3

Counter: 4

Counter: 5
```

**While Loop Example:**

```
DATA: lv_counter TYPE i.

lv_counter = 0.


WHILE lv_counter < 5.

  WRITE: 'Counter:', lv_counter.

  lv_counter = lv_counter + 1.

ENDWHILE.
```

**Explanation:** In this example, the WHILE loop is executed as long as lv_counter is less than 5. The loop starts with lv_counter initialized to 0, and within the loop, the current value of lv_counter is displayed using the WRITE statement. After each iteration, the value of lv_counter is incremented by 1 using the assignment statement lv_counter = lv_counter + 1.

**The output will be:**

```
Counter: 0

Counter: 1

Counter: 2

Counter: 3

Counter: 4
```

# Lab 9 - Insert Data Table Entries and print them on the console

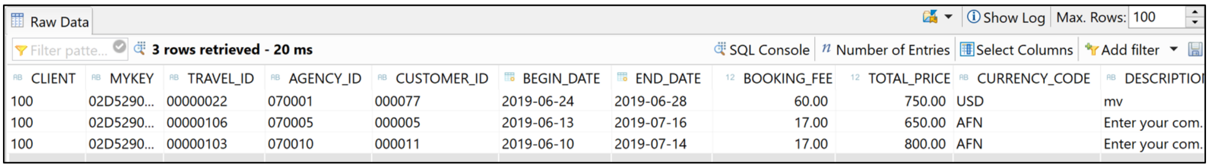
**Open the class first created “zcl_generate_travel_data_xxx” and paste the following code.**

```
CLASS zcl_generate_travel_data_xxx DEFINITION
  PUBLIC
  FINAL
  CREATE PUBLIC .
  PUBLIC SECTION.
    INTERFACES if_oo_adt_classrun.
  PROTECTED SECTION.
  PRIVATE SECTION.
ENDCLASS.

CLASS zcl_generate_travel_data_xxx IMPLEMENTATION.
  METHOD if_oo_adt_classrun~main.

    DATA itab TYPE TABLE OF ztravel_xxx.
*   fill internal travel table (itab)
    itab = VALUE #(
      ( mykey = '02D5290E594C1EDA93815057FD946624' travel_id = '00000022'
agency_id = '070001' customer_id = '000077' begin_date = '20190624' end_date
= '20190628' booking_fee = '60.00' total_price =  '750.00' currency_code =
'USD'
        description = 'mv' overall_status = 'A' created_by = 'MUSTERMANN'
created_at = '20190612133945.5960060' last_changed_by = 'MUSTERFRAU'
last_changed_at = '20190702105400.3647680' )
      ( mykey = '02D5290E594C1EDA93815C50CD7AE62A' travel_id = '00000106'
agency_id = '070005' customer_id = '000005' begin_date = '20190613' end_date
= '20190716' booking_fee = '17.00' total_price = '650.00' currency_code =
'AFN'
        description = 'Enter your comments here' overall_status = 'A' cre-
ated_by = 'MUSTERMANN' created_at = '20190613111129.2391370' last_changed_by
= 'MUSTERMANN' last_changed_at = '20190711140753.1472620' )
      ( mykey = '02D5290E594C1EDA93858EED2DA2EB0B' travel_id = '00000103'
agency_id = '070010' customer_id = '000011' begin_date = '20190610' end_date
= '20190714' booking_fee = '17.00' total_price = '800.00' currency_code =
'AFN'
        description = 'Enter your comments here' overall_status = 'X' cre-
ated_by = 'MUSTERFRAU' created_at = '20190613105654.4296640' last_changed_by
= 'MUSTERFRAU' last_changed_at = '20190613111041.2251330' )
    ).

*   delete existing entries in the database table
    DELETE FROM ztravel_xxx.

*   insert the new table entries
    INSERT ztravel_xxx FROM TABLE @itab.

*   output the result as a console message
    out->write( |{ sy-dbcnt } travel entries inserted successfully!| ).
  ENDMETHOD.
ENDCLASS.
```

1. Save, activate and click F9 to run your ABAP class.

- Save – Ctrl + S
- Activate – Ctrl + F3
- Run – F9

Check your result. Therefore, open your database table ZTRAVEL_XXX and press F8 to see your data. Now the dictionary tables are filled with data.



3 rows retrieved - 20 ms

| CLIENT | MYKEY | TRAVEL_ID | AGENCY_ID | CUSTOMER_ID | BEGIN_DATE | END_DATE | BOOKING_FEE | TOTAL_PRICE | CURRENCY_CODE | DESCRIPTIOI |
|---|---|---|---|---|---|---|---|---|---|---|
| 100 | 02D5290... | 00000022 | 070001 | 000077 | 2019-06-24 | 2019-06-28 | 60.00 | 750.00 | USD | mv |
| 100 | 02D5290... | 00000106 | 070005 | 000005 | 2019-06-13 | 2019-07-16 | 17.00 | 650.00 | AFN | Enter your com. |
| 100 | 02D5290... | 00000103 | 070010 | 000011 | 2019-06-10 | 2019-07-14 | 17.00 | 800.00 | AFN | Enter your com. |

# Reference

1. https://www.sap.com/india/about/company/what-is-sap.html
2. https://www.investopedia.com/terms/e/erp.asp
3. https://www.g2.com/categories/erp-systems
4. https://learning.sap.com/learning-journey/get-started-with-abap-programming-on-sap-btp/understanding-the-basic-features-of-abap_c0e5346f-a136-4b9f-a167-903
5. https://www.sap.com/documents/2018/08/7a62516c-157d-0010-87a3-c30de2ffd8ff.html
6. 29.https://open.sap.com/courses/mm4h2
7. 30.https://www.udemy.com/topic/sap-mm/
8. 31.https://blogs.sap.com/2008/05/09/sap-project-system-a-ready-reference-part-1
9. 32.https://training.sap.com/content/sap-training-hana
10. 33.https://learning.sap-press.com/abap
11. 34.https://data-flair.training/blogs/sap-hana-vs-sap-s-4-hana/
12. 35.https://axxis-consulting.com/what-is-sap-c4hana/