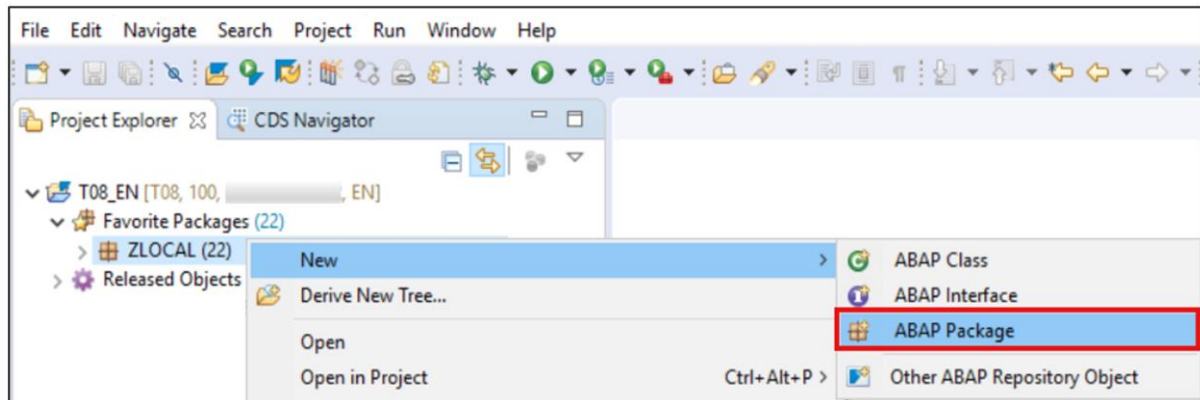


## Exercise 01 Create Table Persistence and Generate Data

### Step 1 Create ABAP Package

Open ABAP Development Tools (ADT) and select your ABAP Cloud Project you created in [Create an SAP BTP ABAP Environment Trial User](#).

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



1. Create a new ABAP package:

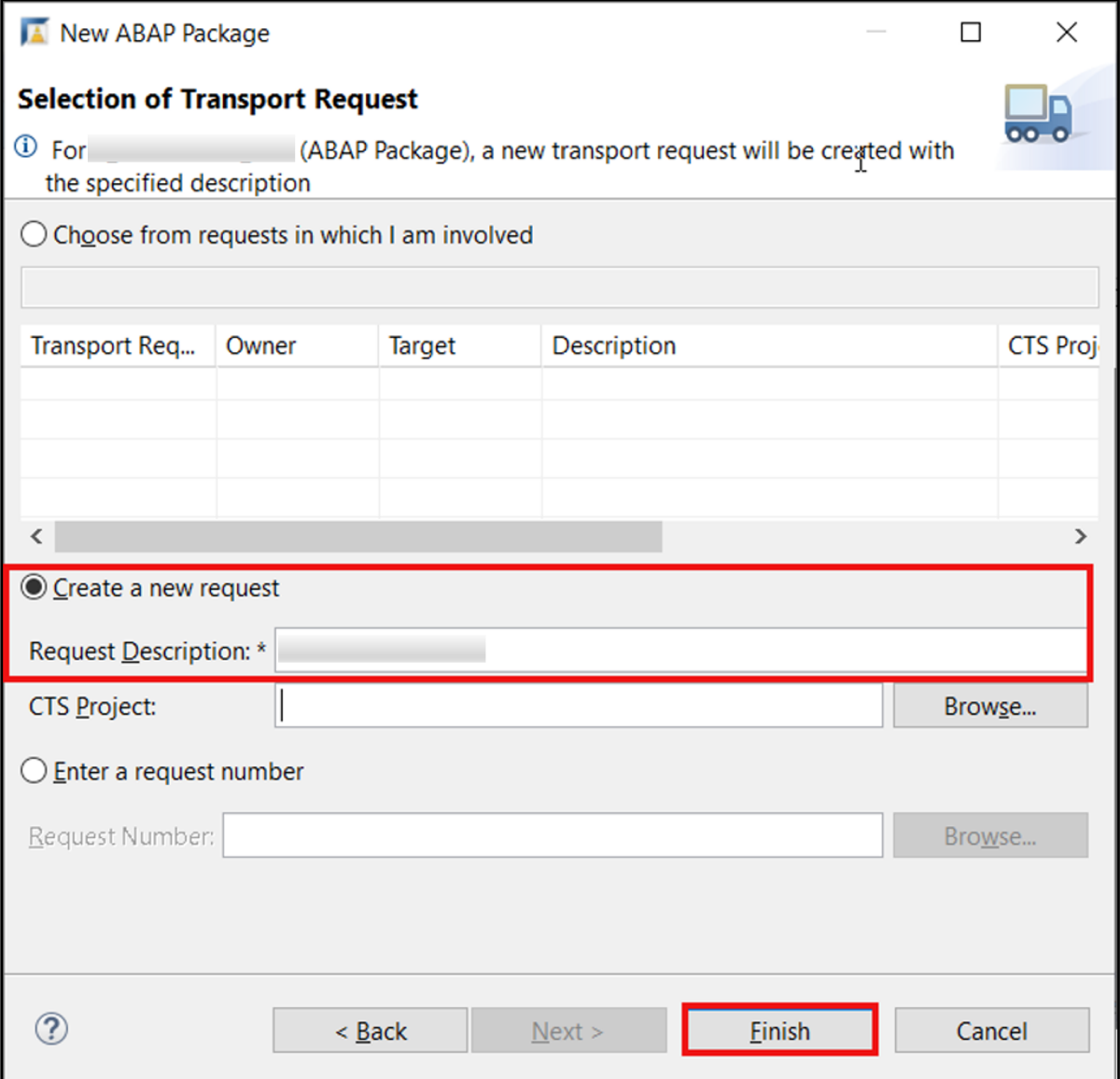
- Name:
- Description:
- Superpackage:
- Check Add to favorite packages.

Use your number instead of .

The screenshot shows the 'New ABAP Package' dialog box. The title bar says 'New ABAP Package'. The main title is 'ABAP Package' with a subtitle 'Create an ABAP package'. The 'Project: \*' field contains 'TRL\_EN'. The 'Name: \*' field contains 'Z230511\_APP\_XXX'. The 'Description: \*' field contains 'Package for travel XXX'. The 'Original Language' field contains 'EN'. The 'Add to favorite packages' checkbox is checked. The 'Superpackage' field contains 'Y-----ABAP\_DEMO\_EXERCISE'. The 'Package Type: \*' dropdown is set to 'Development'. The 'Next >' button is highlighted with a red box.

Click Next >.

2. Select Create new request and enter a request description.



**New ABAP Package**

**Selection of Transport Request**

For (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: \*

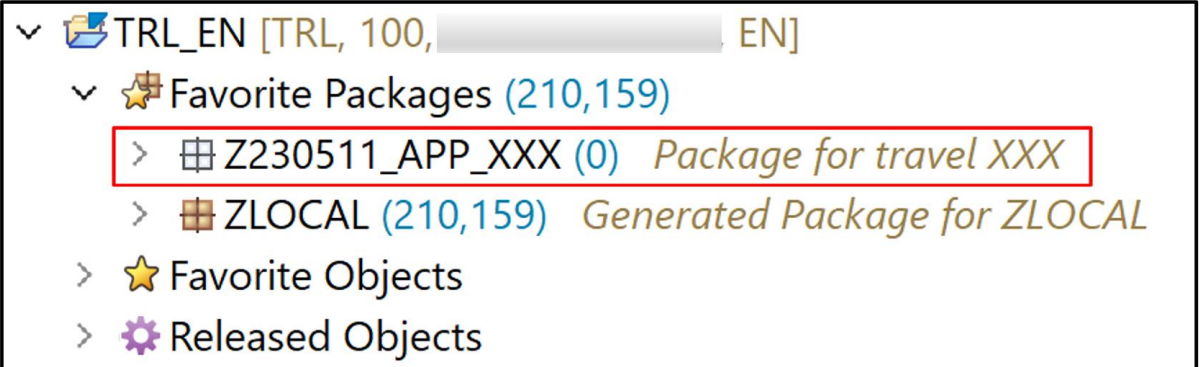
CTS Project: Browse...

☐ Enter a request number

Request Number: Browse...

Click Finish.

3. Now your package is added to favorites objects.

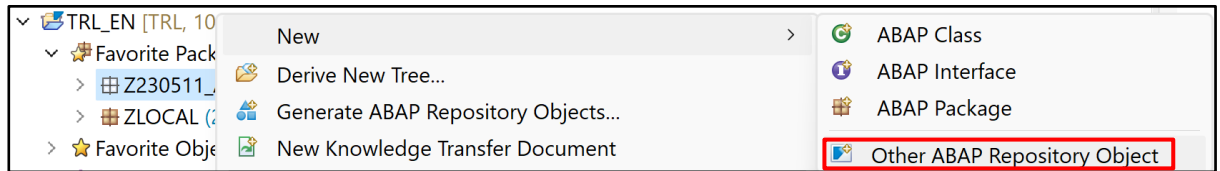


TRL\_EN [TRL, 100, EN]

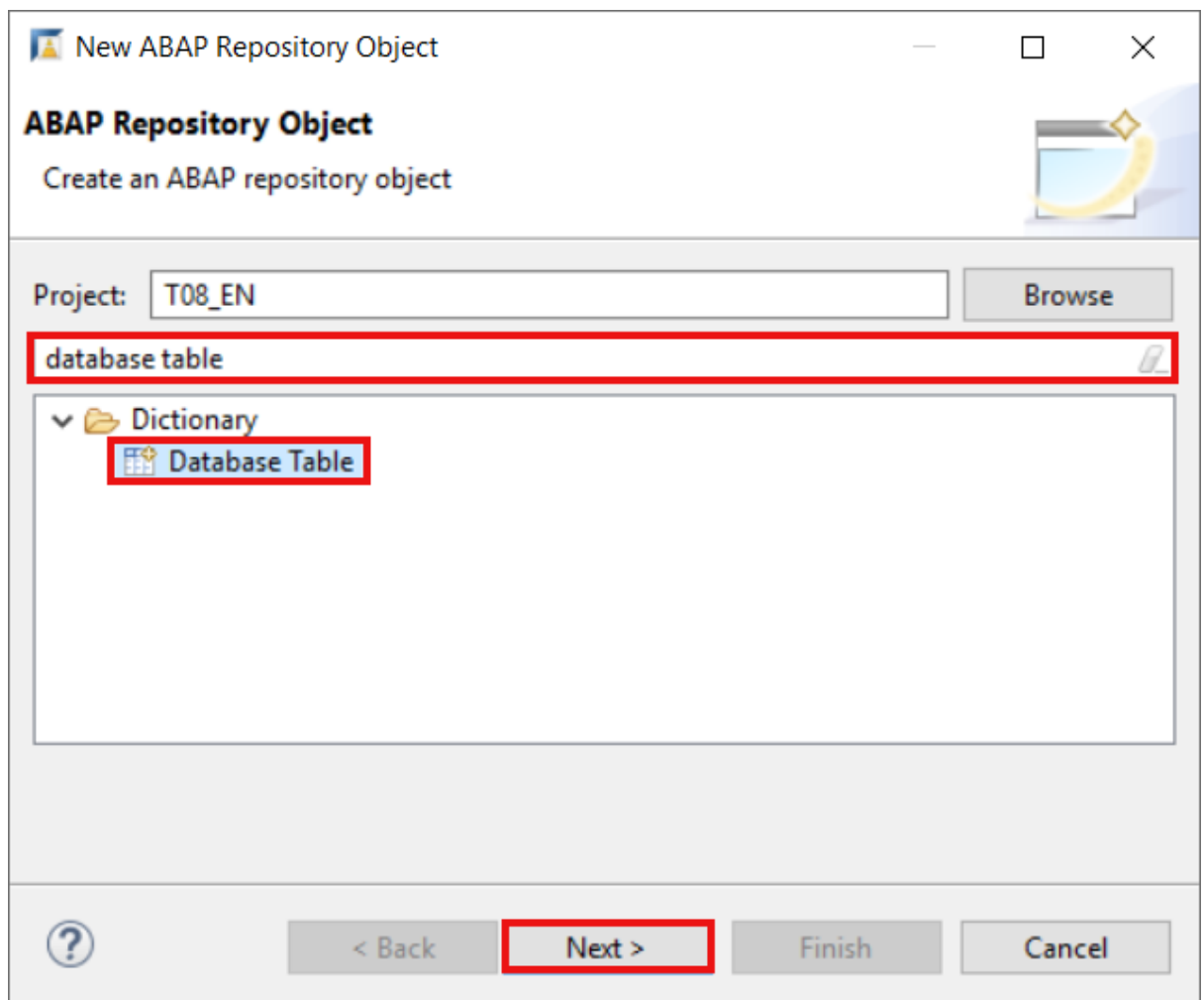
- Favorite Packages (210,159)
  - Z230511\_APP\_XXX (0) Package for travel XXX**
  - ZLOCAL (210,159) Generated Package for ZLOCAL
- Favorite Objects
- Released Objects

## Step 2 Create database table

1. Right-click on your package `Z230511_APP_XXX`, select New > Other ABAP Repository Object.



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



3. Create a new database table:

- Name: ZTAB\_230511\_XXX
- Description: Database table for travel data XXX

Use your number instead of XXX.


The screenshot shows the 'New Database Table' dialog box. The title bar reads 'New Database Table'. Below the title bar, there is a section titled 'Database Table' with a subtitle 'Create a database table' and a small icon of a table. The main area contains several input fields: 'Project: \*' with the value 'TRL\_EN' and a 'Browse...' button; 'Package: \*' with the value 'Z230511\_APP\_XXX' and a 'Browse...' button; an unchecked checkbox 'Add to favorite packages'; 'Name: \*' with the value 'ZTAB\_230511\_XXX'; 'Description: \*' with the value 'Database table for travel data XXX'; and 'Original Language:' with the value 'EN'. At the bottom, there is a navigation bar with a help icon, a '< Back' button, a 'Next >' button (highlighted with a red rectangle), a 'Finish' button, and a 'Cancel' button.

Click Next >.

4. Click Finish to create your transport request.

New Database Table

### Select Transport Request

 For ZTAB\_230511\_XXX (TABL), the selected transport request will be used

☒ Choose from requests in which I am involved [Configure Columns](#)

type filter text

Transport Requ...	Owner	Target	Description	CTS Proje

☐ Create a new request

Request Description: \*

CTS Project:  [Browse...](#)

☐ Enter a request number

Request Number:  [Browse...](#)

[?](#) [< Back](#) [Next >](#) **Finish** [Cancel](#)

5. Replace your code with following:

Use your number instead of xxx.

ABAP

[Copy](#)

```
@EndUserText.label : 'Database table for travel data xxx'

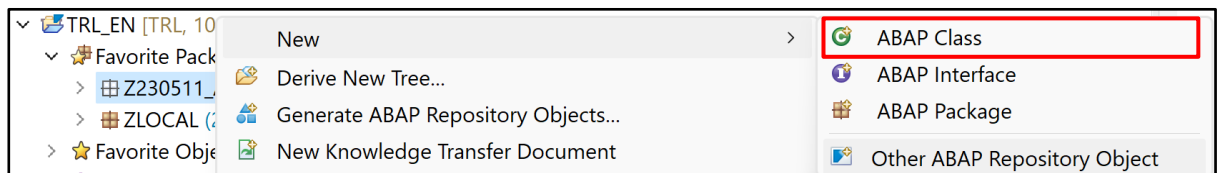
@AbapCatalog.enhancementCategory : #NOT_EXTENSIBLE
@AbapCatalog.tableCategory : #TRANSPARENT
@AbapCatalog.deliveryClass : #A
@AbapCatalog.dataMaintenance : #RESTRICTED

define table ztab_230511_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 : 'ztab_230511_xxx.currency_code'
    booking_fee     : /dmo/booking_fee;
    @Semantics.amount.currencyCode : 'ztab_230511_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      : timestamp1;
    last_changed_by : syuname;
    last_changed_at : timestamp1;
}
```

6. Save and activate.

### Step 3 Create ABAP class

1. Right-click on your package `Z230511_APP_XXX`, select New > ABAP Class.



2. Create a new ABAP class:

- o Name: `ZCL_230511_GENERATE_DATA_XXX`
- o Description: `Class for generating travel data`

Use your number instead of `XXX`.

A screenshot of the 'New ABAP Class' dialog box. The dialog box contains fields for 'Project', 'Package', 'Name', 'Description', 'Original Language', 'Superclass', and 'Interfaces'. The 'Name' field is filled with 'ZCL\_230511\_GENERATE\_DATA\_XXX' and the 'Description' field is filled with 'Class for generating travel data'. The 'Original Language' is set to 'EN'. The 'Next >' button at the bottom is highlighted with a red rectangle.


Click Next >.



3. Click Finish to create your transport request.

New ABAP Class

### Select Transport Request

 For ZCL\_230511\_GENERATE\_DATA\_XXX (CLAS), the selected transport request will be used

☒ Choose from requests in which I am involved [Configure Columns](#)

type filter text

Transport Requ...	Owner	Target	Description	CTS Proje

☐ Create a new request

Request Description: \*

CTS Project:  [Browse...](#)

☐ Enter a request number

Request Number:  [Browse...](#)

[?](#) [< Back](#) [Next >](#) **Finish** [Cancel](#)

4. Replace your code with following:

Use your number instead of xxx.

ABAP

Copy

```
CLASS zcl_230511_generate_data_xxx DEFINITION
```

```
  PUBLIC
```

```
  FINAL
```

```
  CREATE PUBLIC .
```

```
  PUBLIC SECTION.
```

```
    INTERFACES if_oo_adt_classrun.
```

```
  PROTECTED SECTION.
```

```
  PRIVATE SECTION.
```

```
ENDCLASS.
```

```
CLASS zcl_230511_generate_data_xxx IMPLEMENTATION.
```

```
  METHOD if_oo_adt_classrun~main.
```

```
    DATA itab TYPE TABLE OF ztab_230511_xxx.
```

```
    *   fill internal travel table (itab)
```

```
    itab = VALUE #(
```

```
      ( mykey = '02D5290E594C1EDA93815057FD946624' travel_id = '0000022' 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' created_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' created_by =  
      'MUSTERFRAU' created_at = '20190613105654.4296640' last_changed_by = 'MUSTERFRAU'  
      last_changed_at = '20190613111041.2251330' )
```

```
      ( mykey = '02D5290E594C1EDA93858EED2DA2EB24' travel_id = '00000107' agency_id  
      = '070020' customer_id = '000311' begin_date = '20200210' end_date = '20200314'  
      booking_fee = '22.00' total_price = '913.00' currency_code = 'JPY'
```

```

        description = 'Travel for Kioto' overall_status = 'X' created_by = sy-uname
created_at = '20200211105654.4296640' last_changed_by = 'MUSTERFRAU'
last_changed_at = '20210613111041.2251330' )

    ).

*   delete existing entries in the database table
    DELETE FROM ztab_230511_xxx.

*   insert the new table entries
    INSERT ztab_230511_xxx FROM TABLE @itab.


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

ENDMETHOD.
ENDCLASS.

```

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

**Ctrl+S**  
Save  
Save the current contents

**Ctrl + F3**  
or   
Activate

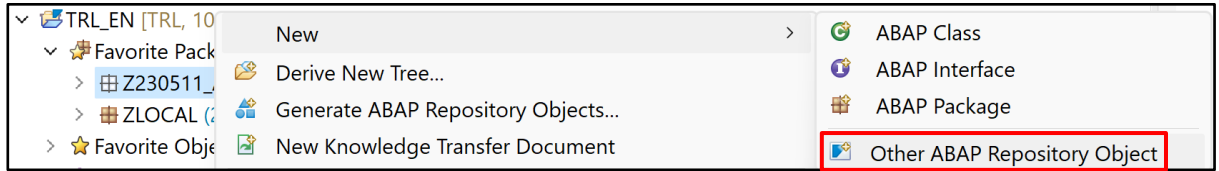
**F9**  
Run

6. Check your result. Therefore open your database table **ZTAB\_230511\_XXX** and press **F8** to see your data. Now the dictionary tables are filled with data.

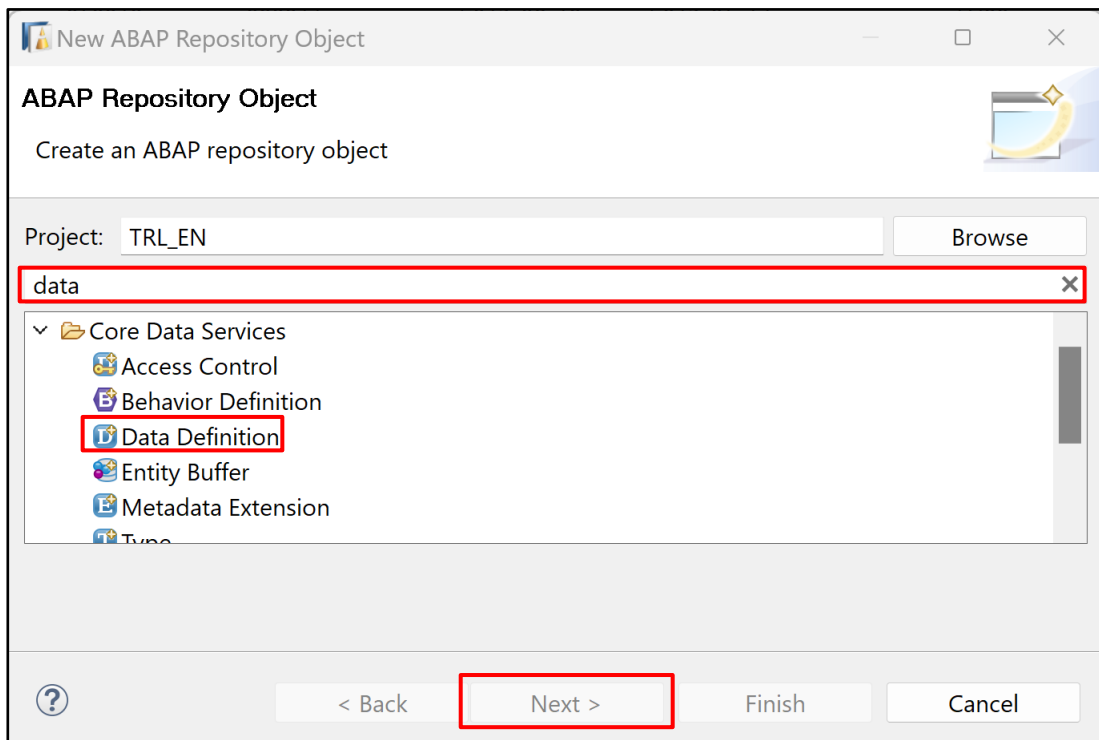
Data Preview												
find pattern 4 rows retrieved - 17 ms												
CLIENT	MYKEY	TRAVEL_ID	AGENCY_ID	CUSTOMER_ID	BEGIN_DATE	END_DATE	BOOKING_FEE	TOTAL_PRICE	CURRENCY_CODE	DESCRIPTION	OVERALL	
100	02D5290...	00000022	070001	000077	2019-06-24	2019-06-28	60.00	750.00	USD	mv	A	
100	02D5290...	00000106	070005	000005	2019-06-13	2019-07-16	17.00	650.00	AFN	Enter your com...	A	
100	02D5290...	00000103	070010	000011	2019-06-10	2019-07-14	17.00	800.00	AFN	Enter your com...	X	
100	02D5290...	00000107	070020	000311	2020-02-10	2020-03-14	22.00	913.00	JPY	Travel for Kioto	X	

## Step 4 Create CDS view

1. Right-click on your package `Z230511_APP_XXX`, select New > Other ABAP Repository Object.



2. Search for data definition, select it and click Next >.



3. Create a new data definition object:

- Name: ZCDS\_230511\_XXX
- Description: CDS for travel data XXX

Use your number instead of XXX.

**New Data Definition**

**Data Definition**  
Create a data definition

Project: \* TRL\_EN Browse...

Package: \* Z230511\_APP\_XXX Browse...

☐ Add to favorite packages

Name: \* ZCDS\_230511\_XXX

Description: \* CDS for travel data XXX

Original Language: EN

Referenced Object: Browse...

? < Back **Next >** Finish Cancel

Click Next >.

4. Click Finish to create your transport request.

The screenshot shows a software window titled "New Data Definition". Inside, the "Select Transport Request" tab is active. An information icon and text state: "For ZCDS\_230511\_XXX (DDL), the selected transport request will be used". A truck icon is in the top right. Below, a radio button labeled "Choose from requests in which I am involved" is selected. To its right is a "Configure Columns" button. A search bar labeled "type filter text" is present. Below the search bar is a table with columns: "Transport Requ...", "Owner", "Target", "Description", and "CTS Proje". The table is currently empty. Below the table, there are two unselected radio button options: "Create a new request" and "Enter a request number". Under "Create a new request", there are input fields for "Request Description: \*" and "CTS Project:", each with a "Browse..." button. Under "Enter a request number", there is an input field for "Request Number:" with a "Browse..." button. At the bottom of the window, there is a navigation bar with a help icon, "< Back", "Next >", "Finish" (highlighted with a red rectangle), and "Cancel".

New Data Definition

### Select Transport Request

For ZCDS\_230511\_XXX (DDL), the selected transport request will be used

☒ Choose from requests in which I am involved Configure Columns

type filter text

Transport Requ...	Owner	Target	Description	CTS Proje

☐ Create a new request

Request Description: \*

CTS Project:

☐ Enter a request number

Request Number:

5. Replace your code with following:

Use your number instead of xxx.

ABAP

[Copy](#)

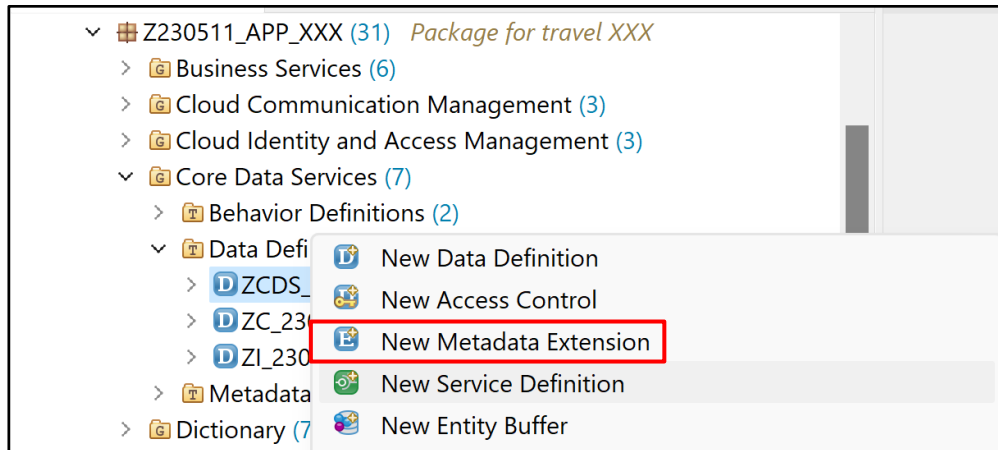
```
@AbapCatalog.sqlViewName: 'ZV_230511_XXX'
@AbapCatalog.compiler.compareFilter: true
@AccessControl.authorizationCheck: #NOT_REQUIRED
@AbapCatalog.viewEnhancementCategory: [#NONE]
@EndUserText.label: 'CDS for travel data XXX'
@Metadata.allowExtensions: true

define view ZCDS_230511_XXX
as select from ztab_230511_xxx
{
  key mykey,
    travel_id,
    agency_id,
    customer_id,
    begin_date,
    end_date,
    @Semantics.amount.currencyCode : 'currency_code'
    booking_fee,
    @Semantics.amount.currencyCode : 'currency_code'
    total_price,
    @Semantics.currencyCode: true
    currency_code,
    description,
    overall_status,
    created_by,
    created_at,
    last_changed_by,
    last_changed_at
}
```

6. Save and activate.

## Step 5 Create Metadata Service Definition

1. Navigate through your package `z230511_XXX`, Right-click on your CDS `ZCDS_230511_XXX`, select New Metadata Extension.





## 2. Create a new metadata extension object:

- Name:
- Description:

Use your number instead of .

**New Metadata Extension**

**Metadata Extension**  
Create a metadata extension

Project: \*

Package: \*

☐ Add to favorite packages

Name: \*

Description: \*

Original Language:


Extended Entity:


Click Next >.

3. Click Finish to create your transport request.


New Metadata Extension

Select Transport Request



 For ZME\_230511\_XXX (DDLX), the selected transport request will be used

☒ Choose from requests in which I am involved

 Configure Columns

Transport Requ...	Owner	Target	Description	CTS Proje


☐ Create a new request

Request Description: \*

CTS Project:

☐ Enter a request number

Request Number:



#### 4. Replace your code with following:

Use your number instead of xxx.

ABAP

[Copy](#)

```
@Metadata.layer: #CUSTOMER
@UI: {
  headerInfo: {
    typeName: 'Travel xxx',
    typeNamePlural: 'Travels xxx'
  }
}
@Search.searchable: true
annotate view ZCDS_230511_xxx
with
{
  @UI: { lineItem: [ { position: 10 ,
                      label: 'Unique Key' } ],
        hidden: true }
  mykey;
  @UI.lineItem: [ { position: 20 ,
                    label: 'Travel ID' } ]
  @UI.selectionField: [{position: 10 }]
  travel_id;
  @UI.lineItem: [ { position: 30 ,
                    label: 'Agency ID' } ]
  agency_id;
  @UI.lineItem: [ { position: 40 ,
                    label: 'Customer ID' } ]
  customer_id;
  @UI.lineItem: [ { position: 50 ,
                    label: 'Begin date' } ]
  begin_date;
  @UI.lineItem: [ { position: 60 ,
                    label: 'End date' } ]
  end_date;
  @UI.lineItem: [ { position: 70 ,
                    label: 'Booking fee' } ]
  booking_fee;
  @UI.lineItem: [ { position: 80 ,
                    label: 'Total price' } ]
  total_price;
  @UI.lineItem: [ { position: 90 ,
                    label: 'Currency' } ]
  currency_code;
  @UI.lineItem: [ { position: 100 ,
                    label: 'Description' } ]
  @Search: { defaultSearchElement: true,
              fuzzinessThreshold: 0.8,
              ranking: #HIGH }
  description;
  @UI.lineItem: [ { position: 110 ,
```

```

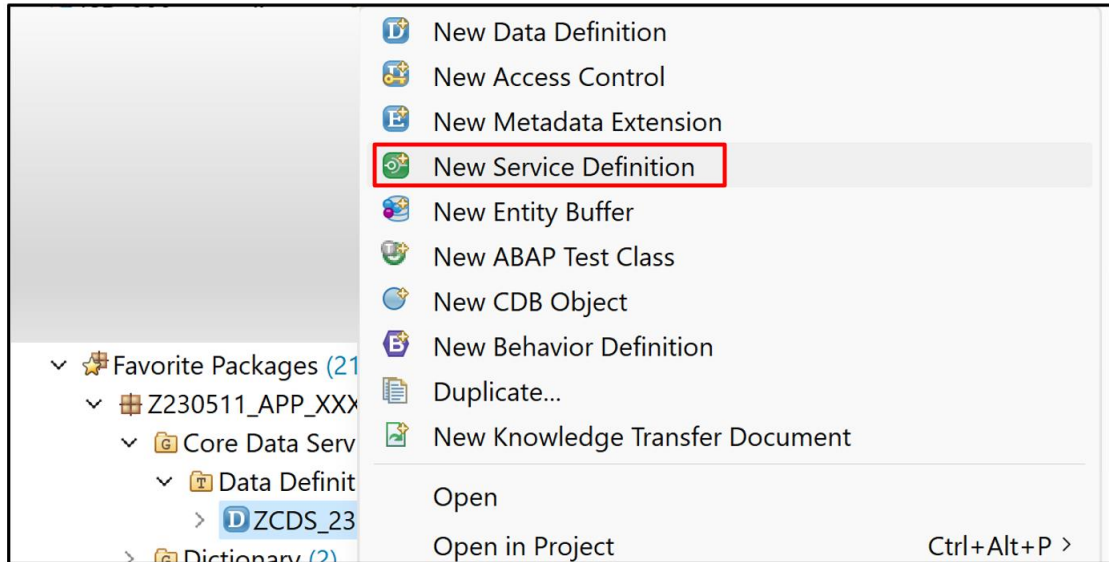
        label: 'Status' } ]
@UI.selectionField: [{position: 20 }]
overall_status;
@UI.lineItem: [ { position: 120 ,
                  label: 'Created by' } ]
created_by;
@UI.lineItem: [ { position: 130 ,
                  label: 'Created at' } ]
created_at;
@UI: { lineItem: [ { position: 140 ,
                    label: 'Last changed by' } ],
      hidden: true }
last_changed_by;
@UI: { lineItem: [ { position: 150 ,
                    label: 'Last changed by' } ],
      hidden: true }
last_changed_at;
}

```

5. Save and activate.

## Step 6 Create Service Definition

6. Navigate through your package `Z230511_XXX`, Right-click on your CDS `ZCDS_230511_XXX`, select New Service Definition.



7. Create a new service definition object:

- Name:
- Description:

Use your number instead of .

New Service Definition

**Service Definition**

Create a service definition

Project: \*

Package: \*

☐ Add to favorite packages

Name: \*

Description: \*

Original Language:

Source Type:


Referenced Object:


Click Next >.

8. Click Finish to create your transport request.

New Service Definition

### Select Transport Request

 For ZSD\_230511\_XXX (SRVD), the selected transport request will be used

☒ Choose from requests in which I am involved  Configure Columns

type filter text

Transport Requ...	Owner	Target	Description	CTS Proje


☐ Create a new request

Request Description: \*

CTS Project:

☐ Enter a request number

Request Number:



9. Replace your code with following:

Use your number instead of xxx.

ABAP

[Copy](#)

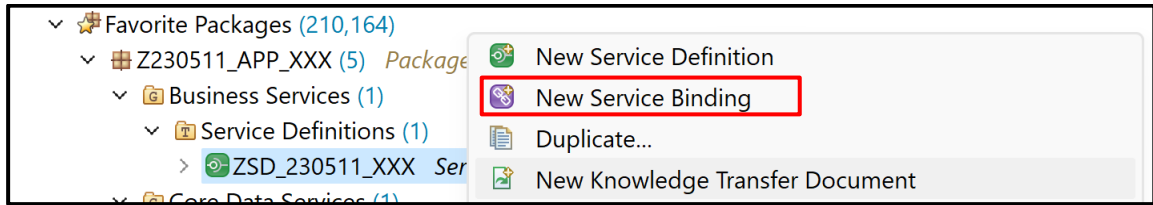
```
@EndUserText.label: 'Service Definition for travel data xxx'
define service ZSD_230511_XXX
{
  Expose ZCDS_230511_XXX as Travel;
}
```

10. Save and activate.



## Step 6 Create Service Binding

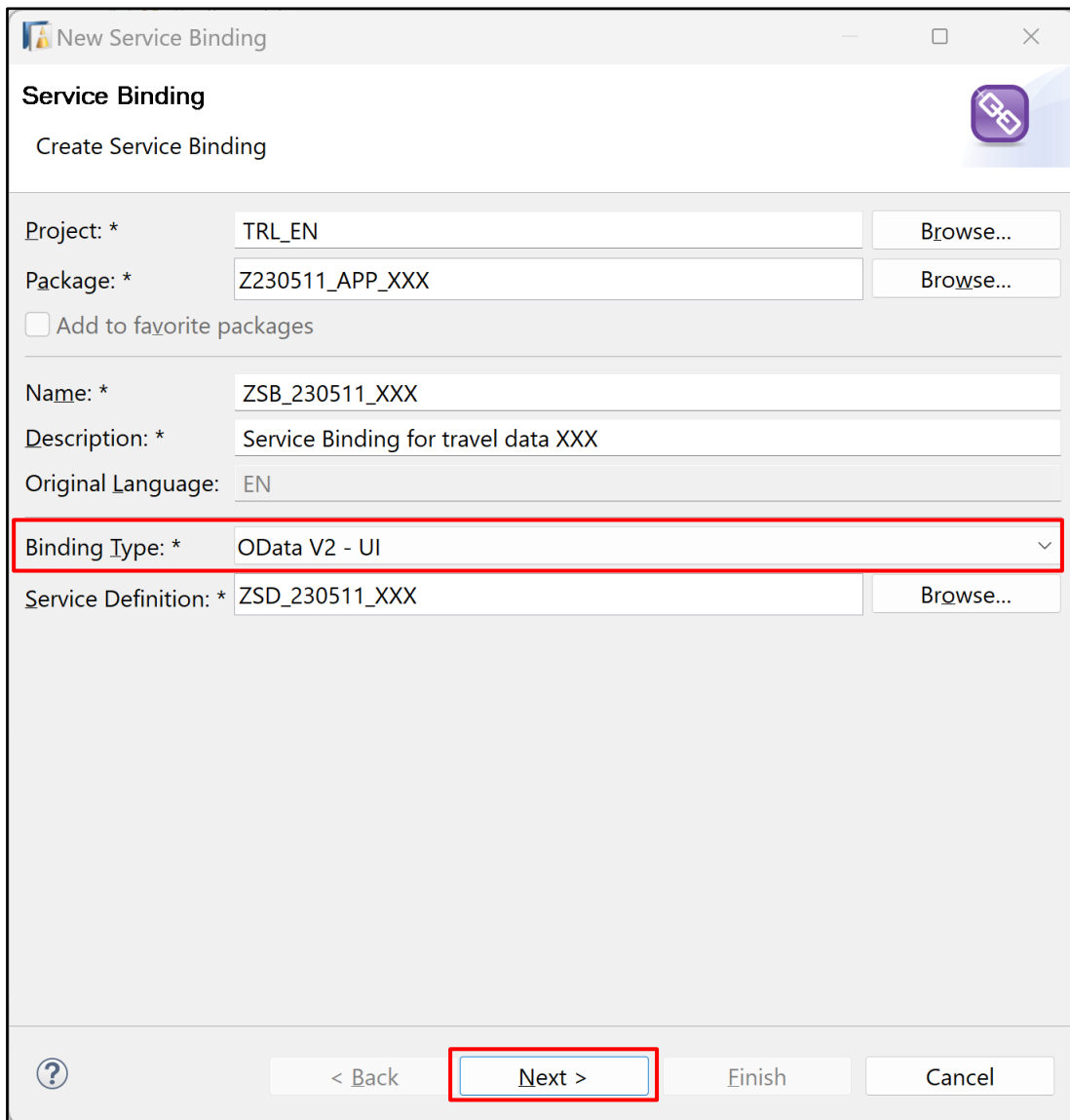
1. Navigate through your package `z230511_XXX`, Right-click on your Service Definition `ZSD_230511_XXX`, select New Service Binding.



## 2. Create a new service binding:

- Name: ZSB\_230511\_XXX
- Description: Service Binding for travel data XXX
- Binding Type: OData V2 - UI

Use your number instead of XXX.



**New Service Binding**

Create Service Binding

Project: \* TRL\_EN Browse...

Package: \* Z230511\_APP\_XXX Browse...

☐ Add to favorite packages

Name: \* ZSB\_230511\_XXX

Description: \* Service Binding for travel data XXX

Original Language: EN

Binding Type: \* OData V2 - UI

Service Definition: \* ZSD\_230511\_XXX Browse...

< Back Next > Finish Cancel

Click Next >.

3. Click Finish to create your transport request.

New Service Binding

Select Transport Request

For ZSB\_230511\_XXX (SRVB), the selected transport request will be used

Choose from requests in which I am involved

Configure Columns

type filter text

Transport Requ...	Owner	Target	Description	CTS Proje

Create a new request

Request Description: \*

CTS Project:

Browse...

Enter a request number

Request Number:

Browse...

< Back

Next >

Finish

Cancel

#### 4. Activate the Service Binding:

Service Binding ZSB\_230511\_XXX [TRL] - inactive - TRL\_EN - ABAP Development Tools

Search Project Run Window Help

CDS Navigator

- XXX (6) Package for travel XXX
- services (2)
- Bindings (1)
- 230511\_XXX Service Binding for travel XXX
- Definitions (1)
- 230511\_XXX Service Definition for travel XXX
- Services (1)
- Definitions (1)
- S\_230511\_XXX CDS for travel data XXX
- CDS\_230511\_XXX
- V\_230511\_XXX
- se Tables (1)

[TRL] ZSB\_230511\_XXX

**Service Binding: ZSB\_230511\_XXX** To enable publish of local service

**General Information**

This section describes general information about this service binding

Binding Type: OData V2 - UI

**Service Versions**

Define service versions associated with the service binding

type filter text

Version	Service Definition
1.0.0	ZSD_230511_XXX

#### 5. Publish the service.

[TRL] ZSB\_230511\_XXX

Service Binding: ZSB\_230511\_XXX

**General Information**

This section describes general information about this service binding

Binding Type: OData V2 - UI

**Service Versions**

Define service versions associated with the service binding

type filter text

Version	Service Definition
1.0.0	ZSD_230511_XXX

Add... Remove

**Service Version Details**

View information on selected service version

Default Authorization Values: 9AF746FA186C6F7634D81D198F7C8CHT

Local Service Endpoint: Unpublished

**Service Information**

Local service endpoint is not published. [Publish local service endpoint](#) to view service URL, entity sets and associations.

Publish

No modify, close or refresh the service binding during the Publish operation.

## 6. Call the service.

At the end of Publishing operation the service will be ready.

The screenshot shows the SAP Service Binding configuration interface for a service binding named **ZSB\_230511\_XXX**. The interface is divided into several sections:

- General Information:** This section describes general information about the service binding. The binding type is **OData V2 - UI**.
- Service Versions:** This section defines service versions associated with the service binding. It includes a table with the following data:

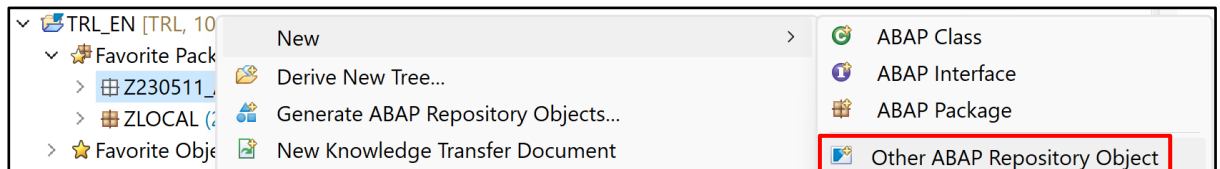
Version	Service Definition
1.0.0	ZSD_230511_XXX
- Service Version Details:** This section provides information on the selected service version. It includes the **Default Authorization Values** (9AF746FA186C6F7634D81D198F7C8CHT) and the **Local Service Endpoint** (Published).
- Service Information:** This section contains the **Service URL**, which is highlighted with a red box: `/sap/opu/odata/sap/ZSB_230511_XXX`.
- Entity Set and Association:** This section shows the **Travel** entity set and association.

The interface also includes buttons for **Add...**, **Remove**, **Unpublish**, and **Preview...**.

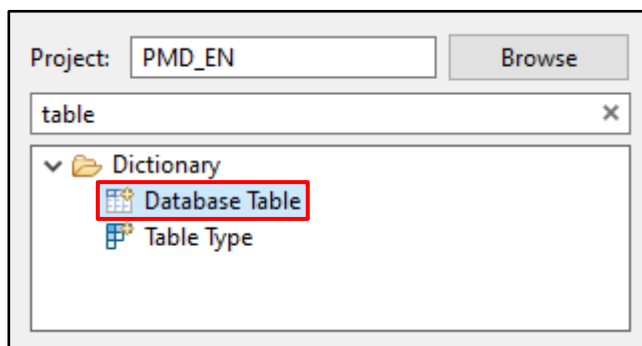
## Exercise 02      Generate a RAP Business Service With the Wizard

### Step 1      Create database table

1. Choose your package, then choose New > Other Repository Object from the context menu.



2. Choose Database Table then Next.



3. Enter the following, then choose Next.

- Name: ZRAP\_230511\_XXX
- Description: Travel Data Generated

Use your number instead of XXX.

4. Choose the transport request, then choose Finish.
5. The table appears in a new editor.

6. Delete the code `key client : abap.clnt;` and add the following:

Use your number instead of `xxx`.

ABAP

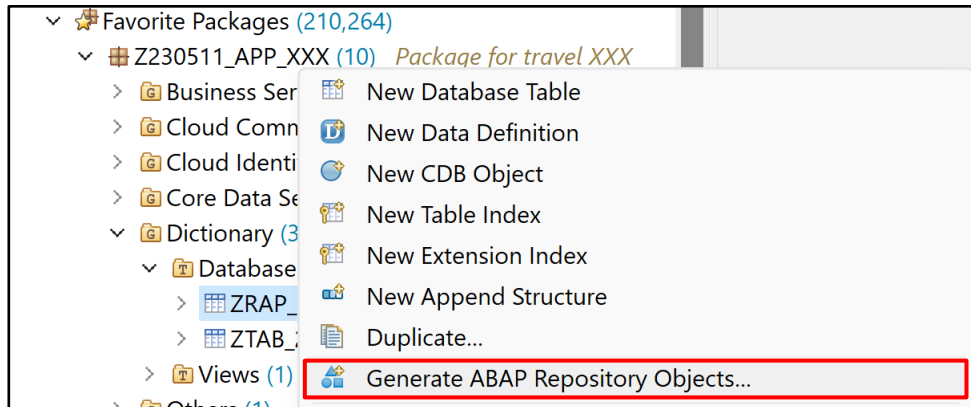
Copy

```
key client          : abap.clnt not null;
key travel_uuid     : sysuuid_x16 not null;
travel_id           : /dmo/travel_id not null;
agency_id           : /dmo/agency_id;
customer_id         : /dmo/customer_id;
begin_date          : /dmo/begin_date;
end_date            : /dmo/end_date;
@Semantics.amount.currencyCode : 'zrap_230511_XXX.currency_code'
booking_fee         : /dmo/booking_fee;
@Semantics.amount.currencyCode : 'zrap_230511_XXX.currency_code'
total_price         : /dmo/total_price;
currency_code       : /dmo/currency_code;
description         : /dmo/description;
overall_status      : /dmo/overall_status;
local_created_by    : abp_creation_user;
local_created_at    : abp_creation_tstmpl;
local_last_changed_by : abp_locinst_lastchange_user;
local_last_changed_at : abp_locinst_lastchange_tstmpl;
last_changed_at     : abp_lastchange_tstmpl;
```

7. Format, save, and activate your table ( `Shift+F1, Ctrl+S, Ctrl+F3` ).

## Step 2      Generate objects from table

1. Select your table and choose Generate ABAP Repository objects... from the context menu.





2. Enter the following and choose Next.

- Use case: ABAP RESTful Application Programming Model: UI Service

Use your number instead of xxx.

**Generate ABAP Repository Objects**

Specify package for the generator

Project: \*

Package: \*

Referenced Object: \*

Original Language:

Generator: \* ABAP RESTful Application Programming Model: UI Service ▼

Next >

3. Enter the following:

- Data definition name:
- Alias name:

Use your number instead of .

The screenshot shows the 'Generate ABAP Repository Objects' dialog box. The 'Configure Generator' tab is active, and the 'Data Model' section is selected. The 'RAP Layers' list on the left includes 'Referenced Object', 'Business Object' (expanded), 'Data Model' (selected), 'Behavior', 'Service Projection', 'Business Service' (expanded), 'Service Definition', and 'Service Binding'. The 'Data Model' section on the right contains the instruction: 'Provide the name of the CDS data definition for the root entity and an alias. CDS data definitions are used to define and organize the data model.' Below this instruction are two input fields: 'Data Definition Name: \*' with the value 'ZI\_230511\_XXX' and 'Alias Name: \*' with the value 'Travel'. Both input fields are highlighted with red rectangles.

**Generate ABAP Repository Objects**

**Configure Generator**

Specify a value for field 'Business Object -> Behavior -> Implementation Class'

**RAP Layers**

type filter text

- Referenced Object
- ✓ Business Object
  - Data Model
  - Behavior
  - Service Projection
- ✓ Business Service
  - Service Definition
  - Service Binding

**Data Model**

Provide the name of the CDS data definition for the root entity and an alias. CDS data definitions are used to define and organize the data model.

Data Definition Name: \*

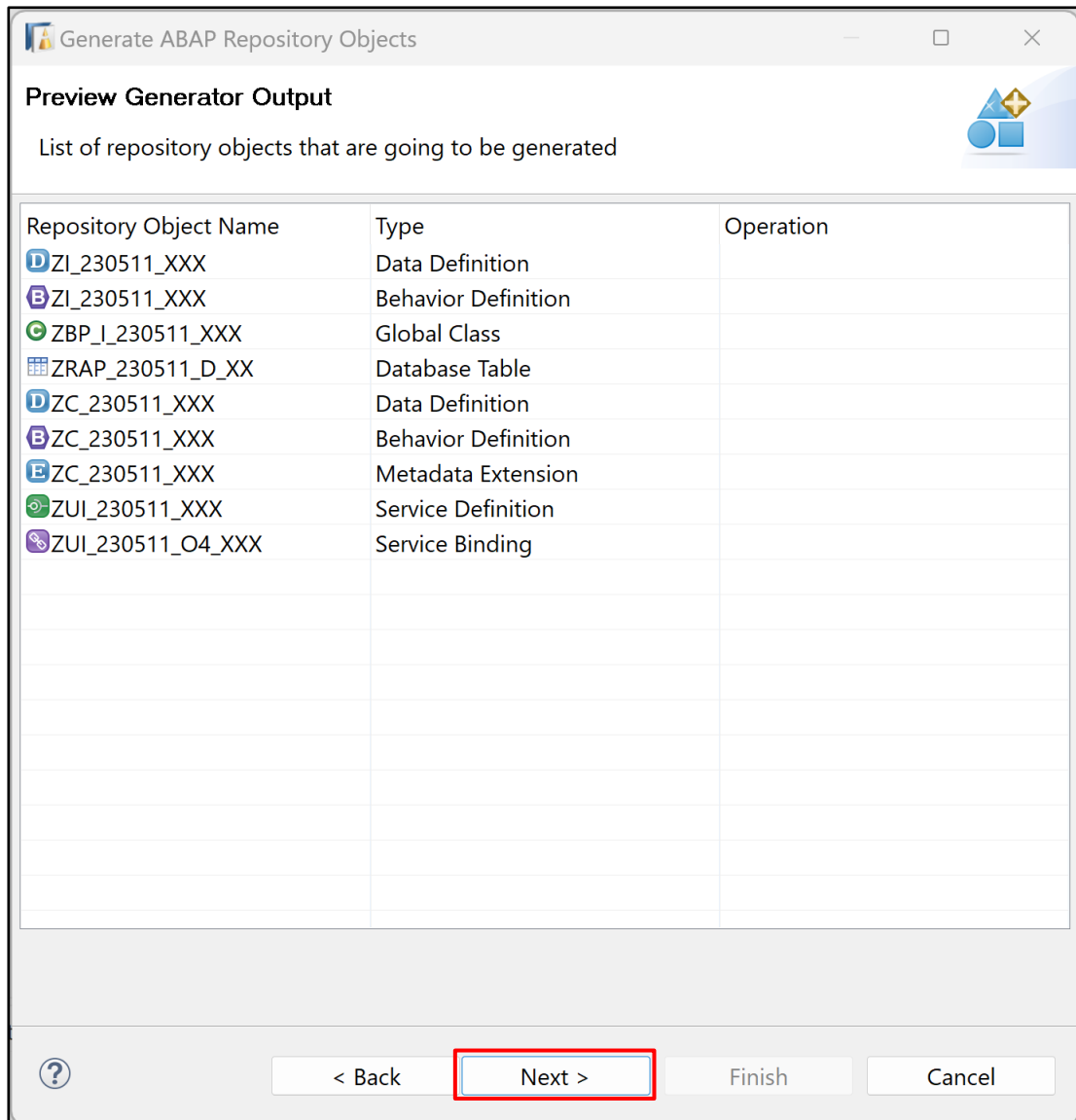
Alias Name: \*

4. Enter the following information for each artifact, then choose Next

Use your number instead of .

Node	Artifact	Value
Behavior	ABAP implementation class	<input type="text" value="ZBP_I_230511_XXX"/>
Behavior	Draft Table Name:	<input type="text" value="ZRAP_230511_D_XXX"/>
Service Projection	CDS projection view entity	<input type="text" value="ZC_230511_XXX"/>
Service Definition	CDS service definition	<input type="text" value="ZUI_230511_XXX"/>
Service Binding	CDS service binding	<input type="text" value="ZUI_230511_O4_XXX"/>
Service Binding	Binding type:	<input type="text" value="OData V4 - UI"/>

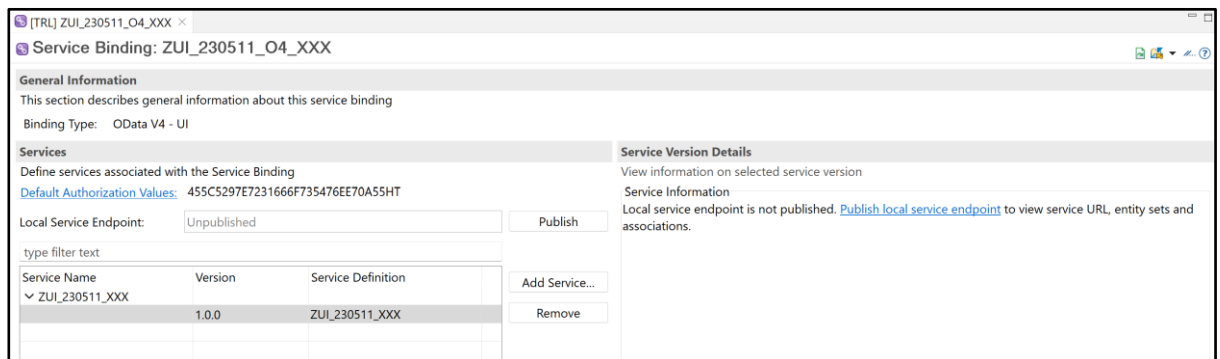
5. The system displays a list of objects that will be generated. Choose Next



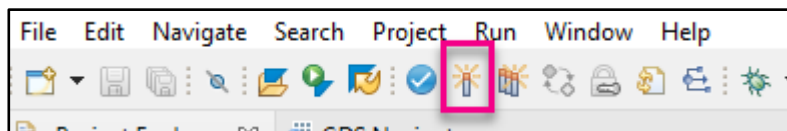
6. Choose the transport request, then choose Finish.

### Step 3      Activate and publish your business service

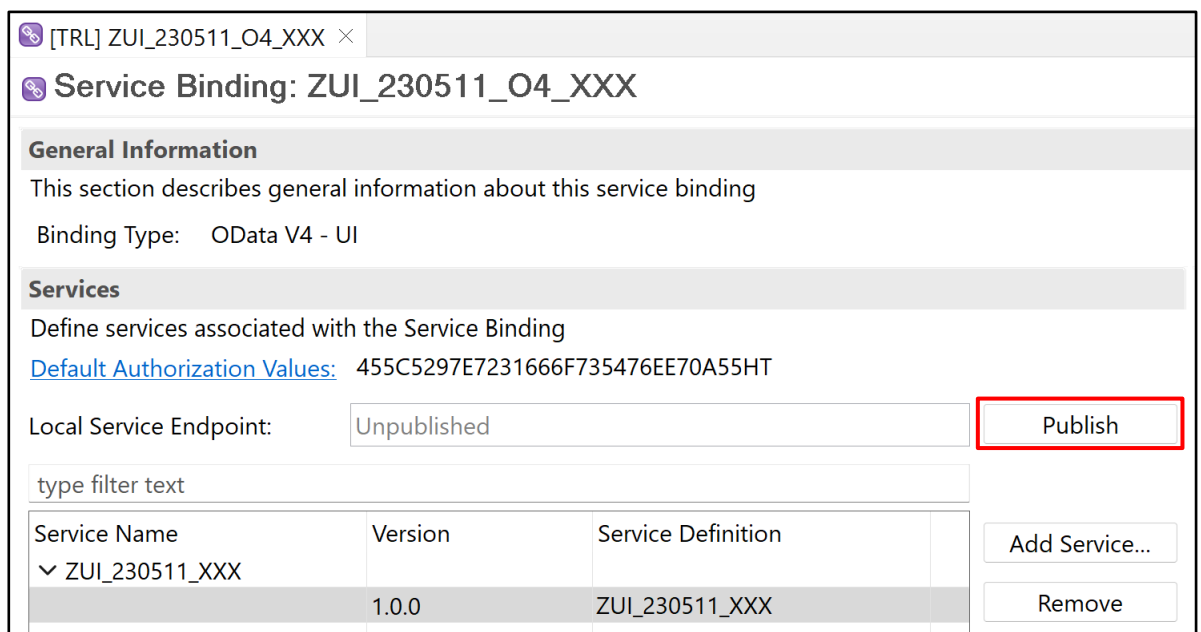
1. The service binding appears automatically in a new editor.



2. Activate the service binding.



3. Publish the service binding.



No modify, close or refresh the service binding during the Publish operation.

#### 4. Create a new service binding (OData V2):

- Name: `ZUI_230511_O2_XXX`
- Description: `Service binding for travel data V2`
- Binding Type: `OData V2 - UI`

Use your number instead of `XXX`.

**New Service Binding**

Create Service Binding

Project: \* `TRL_EN` Browse...

Package: \* `Z230511_APP_XXX` Browse...

☐ Add to favorite packages

Name: \* `ZUI_230511_O2_XXX`

Description: \* `Service binding for travel data V2`

Original Language: `EN`

**Binding Type: \* `OData V2 - UI`**

Service Definition: \* `ZUI_230511_XXX` Browse...

? < Back **Next >** Finish Cancel

Click Next >

5. Click Finish to create your transport request.

The screenshot shows a 'New Service Binding' window with the title bar 'New Service Binding'. The main heading is 'Select Transport Request'. Below the heading is an information icon and text: 'For ZUI\_230511\_O2\_XXX (SRVB), the selected transport request will be used'. There is a truck icon in the top right corner. The first radio button is selected: 'Choose from requests in which I am involved'. To its right is a 'Configure Columns' button. Below this is a search bar with the placeholder text 'type filter text'. A table with five columns is shown: 'Transport Requ...', 'Owner', 'Target', 'Description', and 'CTS Proje'. The table has several empty rows. Below the table is a horizontal scrollbar. The second radio button is 'Create a new request'. It has two input fields: 'Request Description: \*' and 'CTS Project:'. The 'CTS Project:' field has a 'Browse...' button next to it. The third radio button is 'Enter a request number'. It has an input field for 'Request Number:' and a 'Browse...' button next to it. At the bottom of the window is a navigation bar with a help icon, '< Back', 'Next >', 'Finish' (highlighted with a red rectangle), and 'Cancel'.

New Service Binding

### Select Transport Request

For ZUI\_230511\_O2\_XXX (SRVB), the selected transport request will be used

☒ Choose from requests in which I am involved Configure Columns

type filter text

Transport Requ...	Owner	Target	Description	CTS Proje

☐ Create a new request

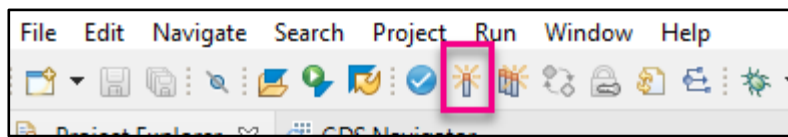
Request Description: \*

CTS Project:

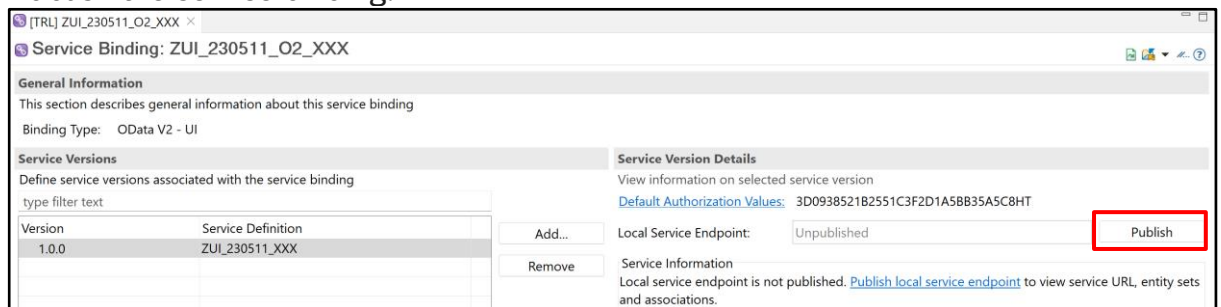
☐ Enter a request number

Request Number:

6. Activate the service binding.



7. Publish the service binding.



No modify, close or refresh the service binding during the Publish operation.

8. Call the service.

At the end of Publishing operation the service will be ready.

