ERP Application Programming (Summer Semester 2025)

Assignment No. 1

Dipl.-Inform. Alfred Kersting

Tasniha Fahmin Chowdhury

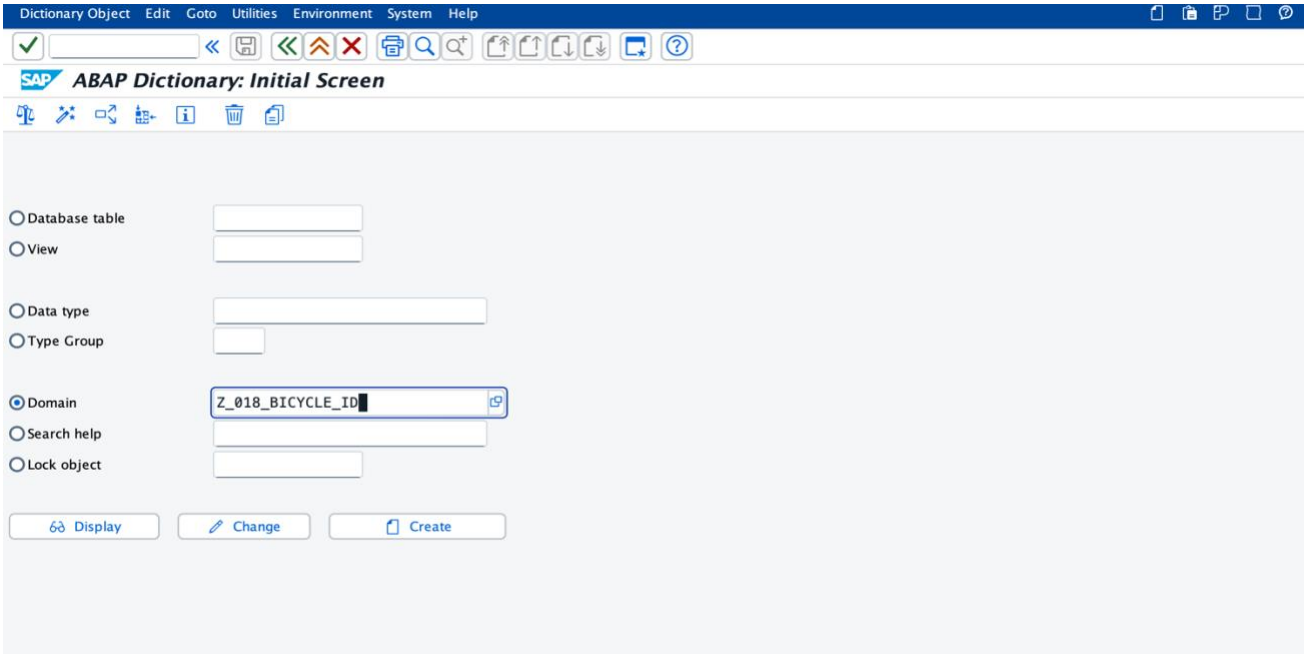
Matriculation Number: 30346633.

Table of Contents

1. SAP ABAP Dictionary

1.1. Domains

To begin the development process in ABAP, the first step is to create a **Domain** using transaction code **SE11** in the ABAP Data Dictionary. A domain defines the technical properties of a field, such as its data type, length, and possible value range. It serves as the foundation for creating data elements and ensures consistency in data definitions across the system. Once the domain is defined, it can be reused in multiple data elements and table fields.



The list of Domains those have been created are listed below,

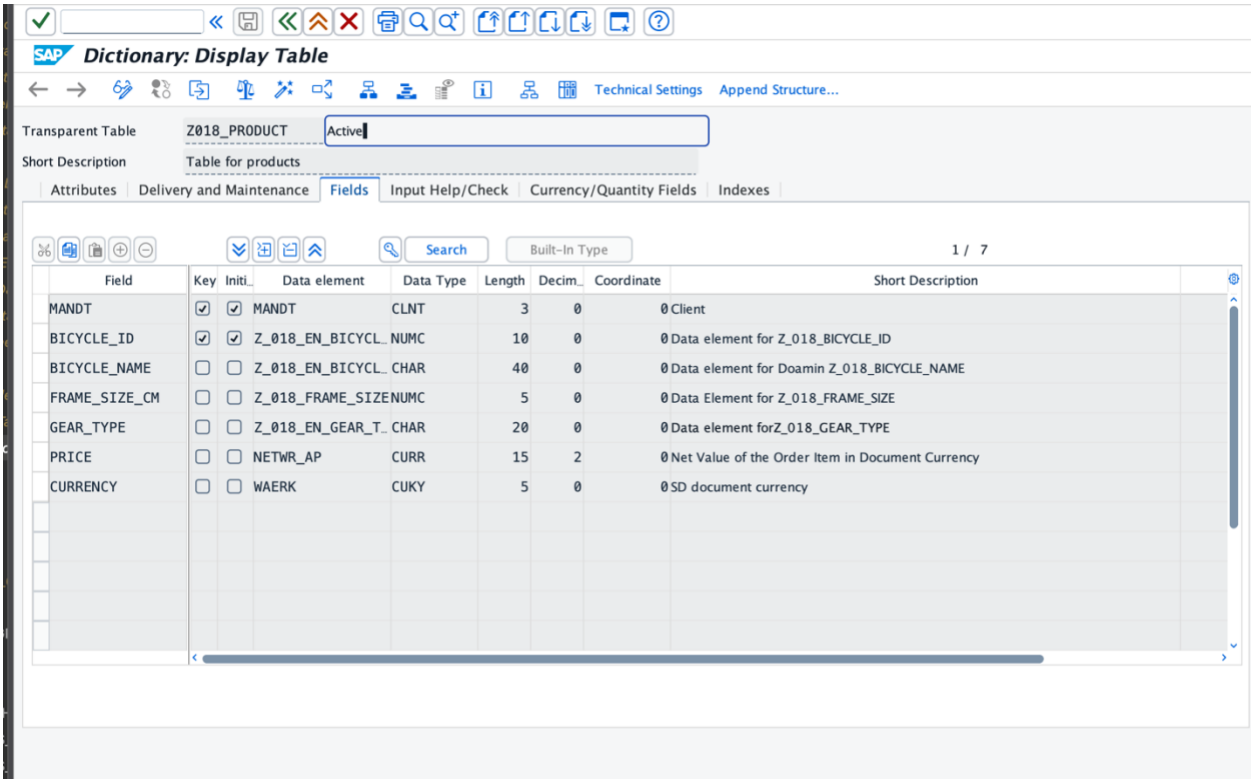
Domain Name	Short Description	Data Type
Z_018_BICYCLE_ID	Domain for Bicycle ID length 5	NUMC
Z_018_BICYCLE_NAME	Domain for Bicycle Name Length 40	CHAR
Z_018_FRAME_SIZE	Domain for Bicycle Frame Size Length 5	NUMC
Z018_GEAR_TYPE	Domain for Bicycle Gear Type Length 20	CHAR
Z_018_CUST_ID	Domain for Customer ID Length 10	CHAR
Z_018_CUST_NAME	Domain for Customer Name Length 40	CHAR
Z_018_BIRTHDAY	Domain for Customer Birthday Length 8	DATS
Z_018_CHAR_10	Domain for Character with Length 10	CHAR
Z_018_QUANTITY	Domain for Quantity Length 10	NUMC

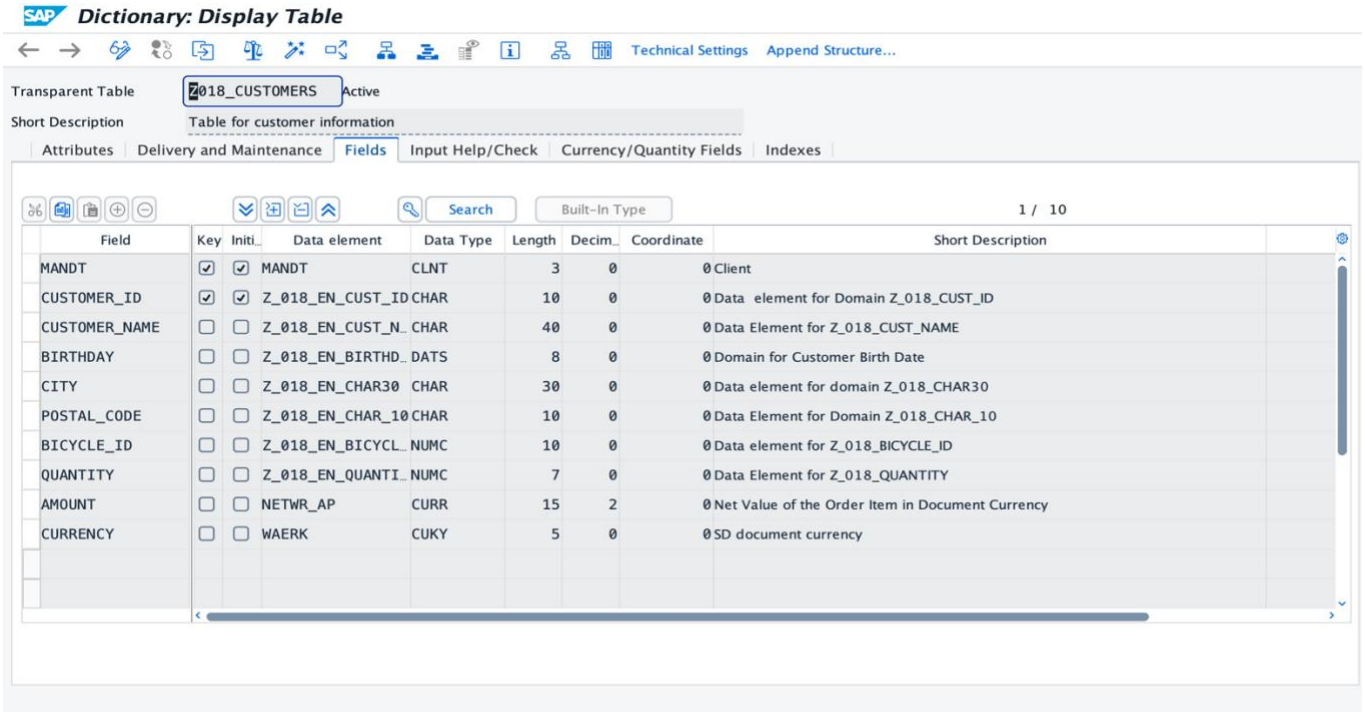
1.2. DATA ELEMENTS

Next, **Data Elements** are created in SE11 and assign it to the previously created **Domain** to define the semantic meaning of a field. The data element adds descriptive labels and documentation while inheriting the domain's technical properties.

Data Element Name	Short Description	Data Type
Z_018_EN_BICYCLE_ID	Data element for Z_018_BICYCLE_ID	NUMC
Z_018_EN_BICYCLE_NAME	Data element for Domain Z_018_BICYCLE_NAME	CHAR
Z_018_FRAME_SIZE	Data Element for Z_018_FRAME_SIZE	NUMC
Z_018_EN_GEAR_TYPE	Data element forZ_018_GEAR_TYPE	CHAR
Z_018_EN_CUST_ID	Data element for Domain Z_018_CUST_ID	CHAR
Z_018_EN_CUST_NAME	Data Element for Z_018_CUST_NAME	CHAR
Z_018_EN_BIRTHDAY	Domain for Customer Birth Date	DATS
Z_018_EN_CHAR_10	Data Element for Domain Z_018_CHAR_10	CHAR
Z_018_EN_QUANTITY	Data Element for Z_018_QUANTITY	NUMC

Data elements those are created and used are showed below in the tables Z018_CUSTOMERS and Z018_PRODUCT. Pre-built data elements **NETWR_AP** for price and **WAERK** for currency are used. Some data elements are used repeatedly.

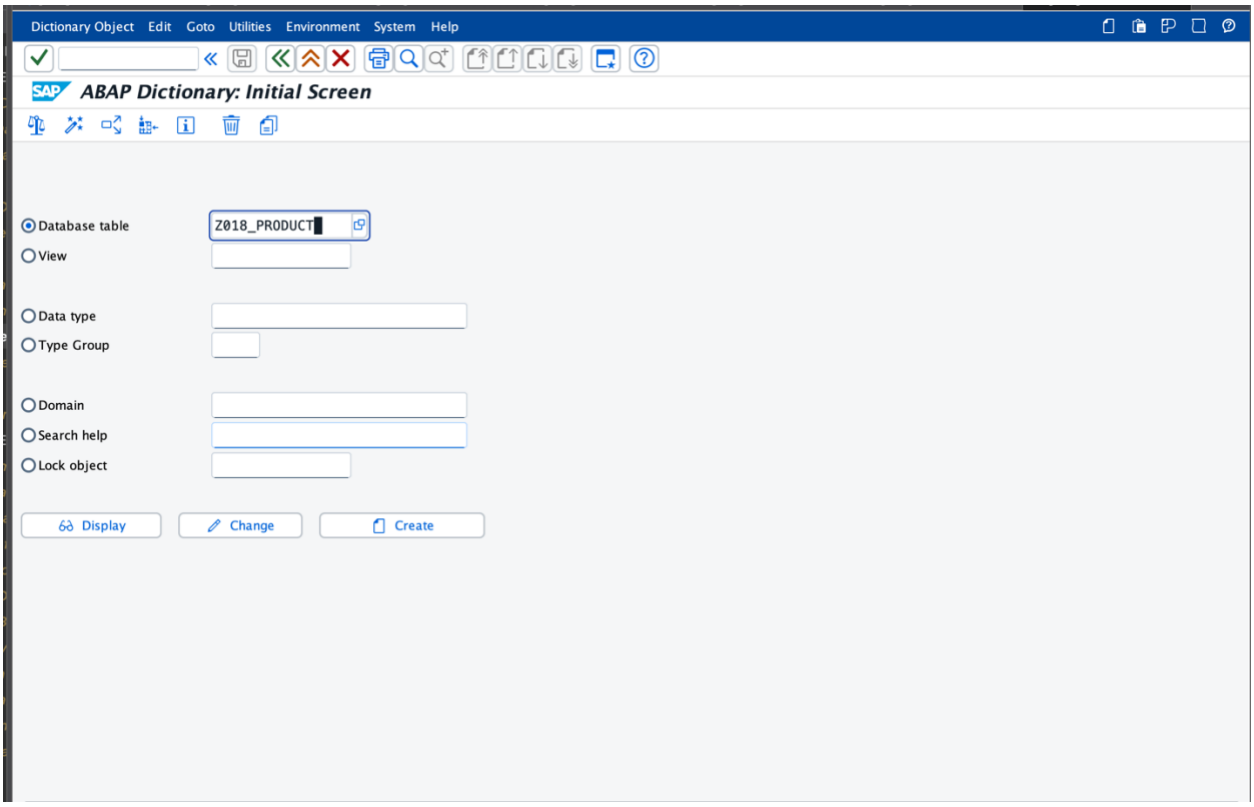



Dictionary: Display Table
Technical Settings
Append Structure...
Transparent Table
Z018_CUSTOMERS
Active
Short Description
Table for customer information
Attributes
Delivery and Maintenance
Fields
Input Help/Check
Currency/Quantity Fields
Indexes
Search
Built-In Type
1 / 10
Field
Key
Initi...
Data element
Data Type
Length
Decim...
Coordinate
Short Description
MANDT
MANDT
CLNT
3
0
Client
CUSTOMER_ID
Z_018_EN_CUST_ID
CHAR
10
0
Data element for Domain Z_018_CUST_ID
CUSTOMER_NAME
Z_018_EN_CUST_N...
CHAR
40
0
Data Element for Z_018_CUST_NAME
BIRTHDAY
Z_018_EN_BIRTHD...
DATS
8
0
Domain for Customer Birth Date
CITY
Z_018_EN_CHAR30
CHAR
30
0
Data element for domain Z_018_CHAR30
POSTAL_CODE
Z_018_EN_CHAR_10
CHAR
10
0
Data Element for Domain Z_018_CHAR_10
BICYCLE_ID
Z_018_EN_BICYCL...
NUMC
10
0
Data element for Z_018_BICYCLE_ID
QUANTITY
Z_018_EN_QUANTI...
NUMC
7
0
Data Element for Z_018_QUANTITY
AMOUNT
NETWR_AP
CURR
15
2
Net Value of the Order Item in Document Currency
CURRENCY
WAERK
CUKY
5
0
SD document currency

2. Database Table

2.1. Z018_PRODUCT

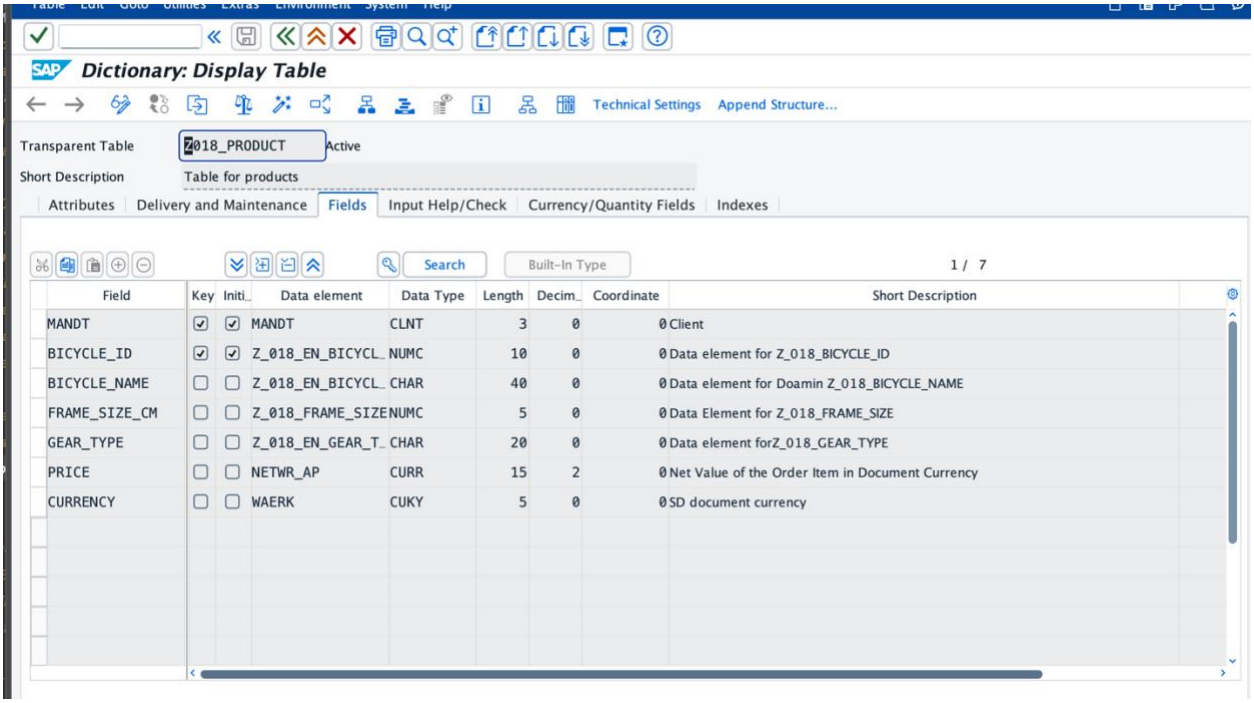
Creating a new database table called Z018_PRODUCT to handle the offered goods which contains all the relevant data. Maintain the short text and then choose ‘A’ Application table (master and transaction data) as the Delivery Class and Display/Maintenance Allowed as the value for Data Browser/Table View Editing.



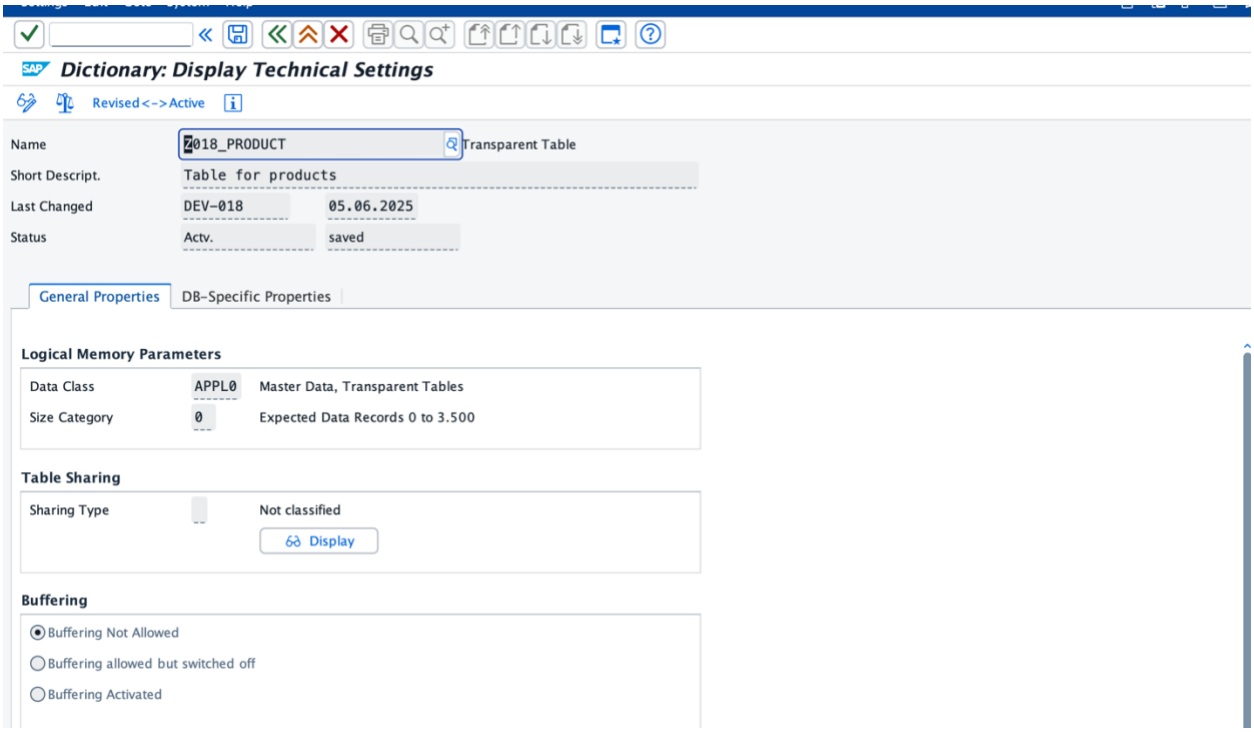
The table fields were maintained accordingly. The field **MANDT** was defined first using the data element **MANDT**, with the **Key** and **Initial Value** options selected, as required for client-dependent tables. Subsequently, the field **BICYCLE_ID** was created and marked as a primary key field by enabling both the **Key** and **Initial Value** checkboxes. The data element **Z_018_EN_BICYCLE_ID** was assigned to this field, based on the data type **NUMC** with a length of **10**. The following fields were included in the table structure:

- **MANDT** – Client (necessary for client-dependent tables)
- **BICYCLE_ID** – Number identifying the bicycle
- **BICYCLE_NAME** – Name of the bicycle
- **FRAME_SIZE_CM** – Frame size in centimeters
- **GEAR_TYPE** – Gear type(Manual or Auto)
- **PRICE** – Price of the bicycle
- **CURRENCY** – Currency used

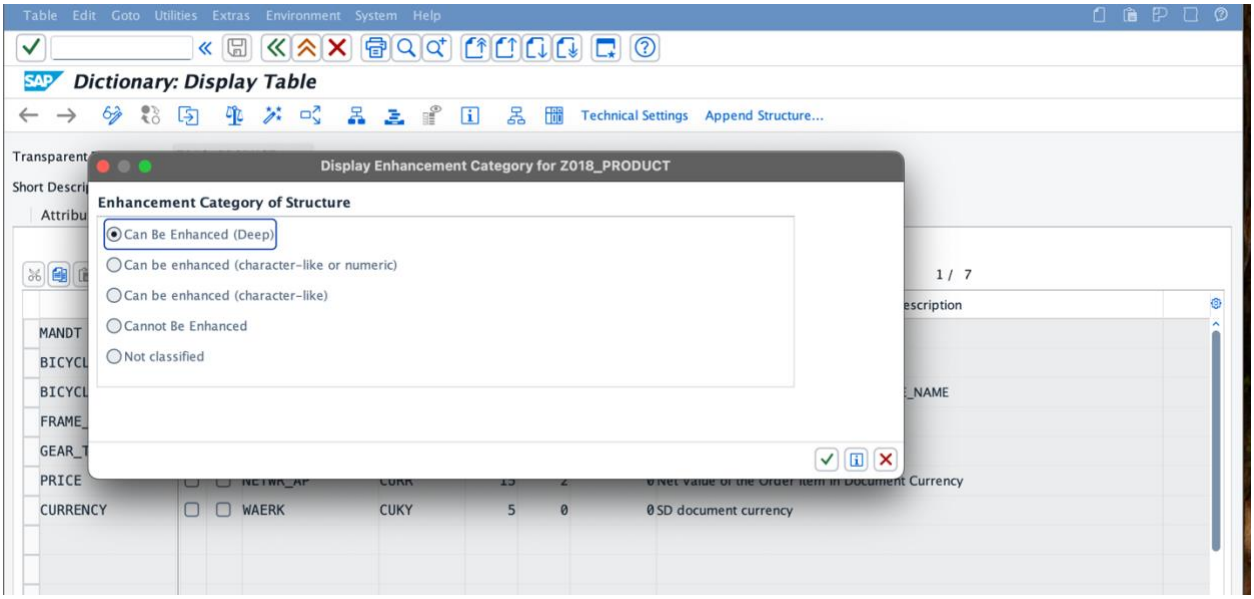
By switching to the ‘Fields’ tab the fields of the table are entered, below the fields are from SAP are added. For the product table.



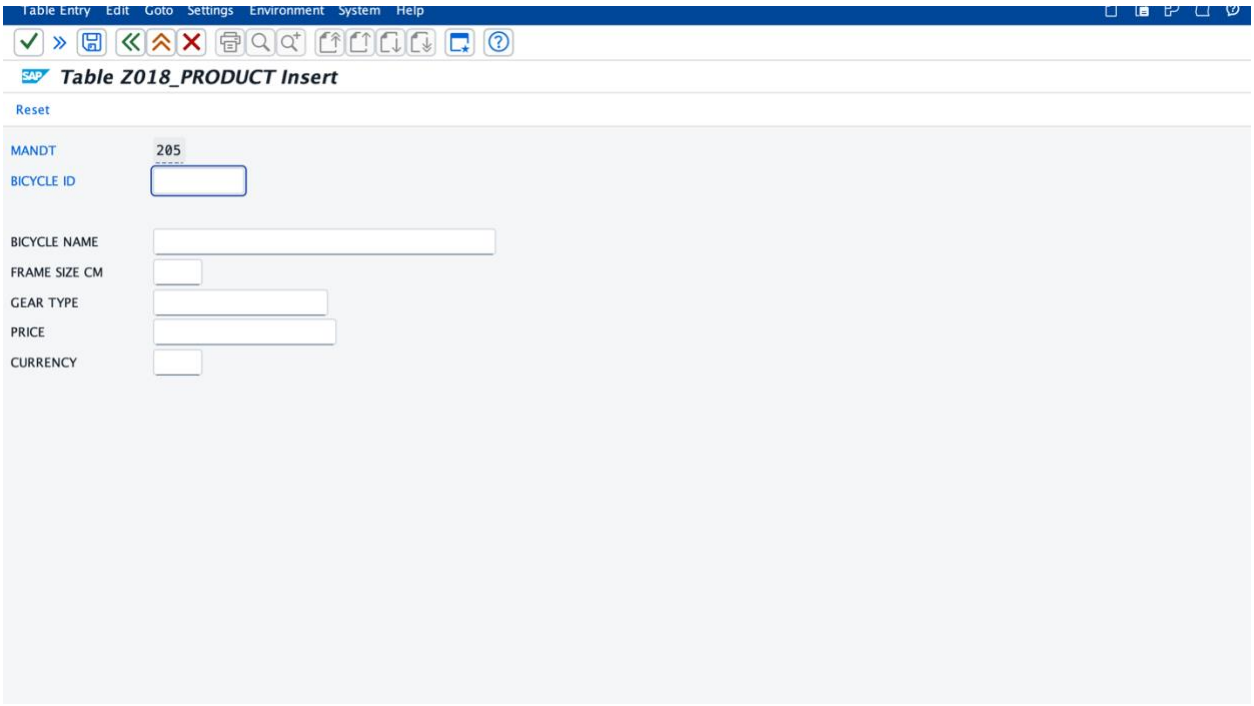
In this context, it is mandatory to maintain the data class. Therefore, the data type **APPL0** has been selected to ensure appropriate application-specific data handling. Additionally, the size category has been set to **0**, which specifies the allocated size for the table storage.



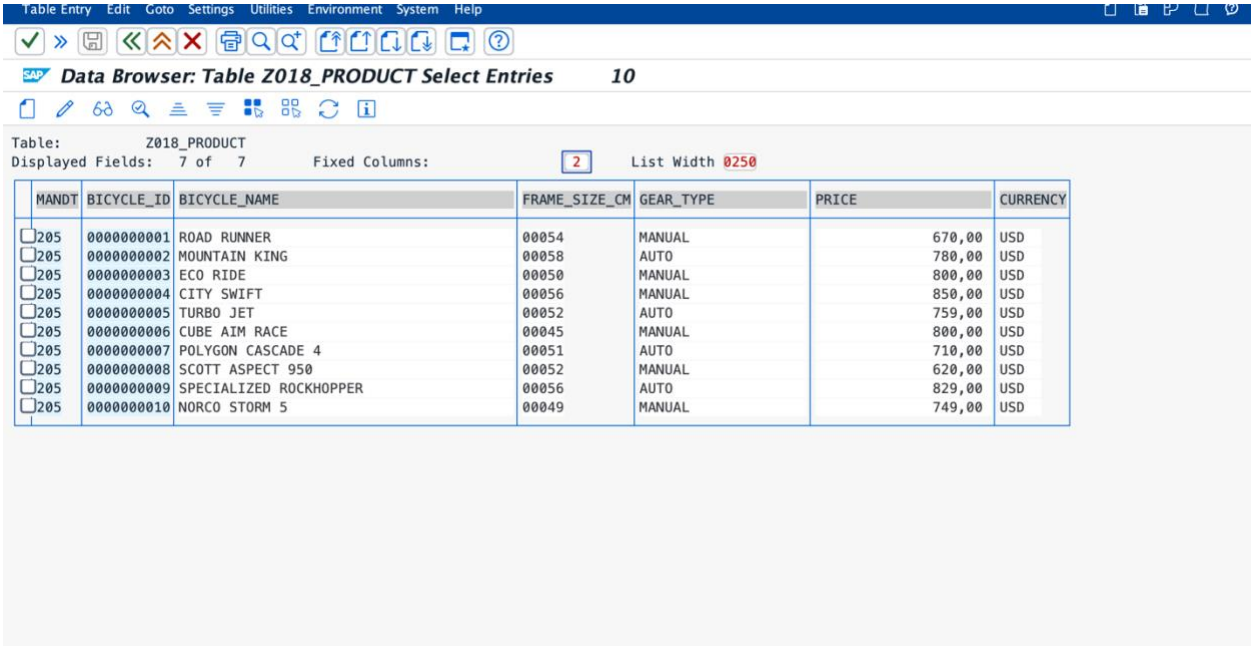
After that, define an enhancement category as Can be Enhanced (Deep). The enhancement category can be defined using following menu path: (Menu •) Extras • Enhancement Category.



In the next step, some data records are entered into the table. This can be done easily using the data browser, which can be accessed via the menu path: Menu => Utilities => Table Contents => Create Entries. Finally, save, check, and activate the table.



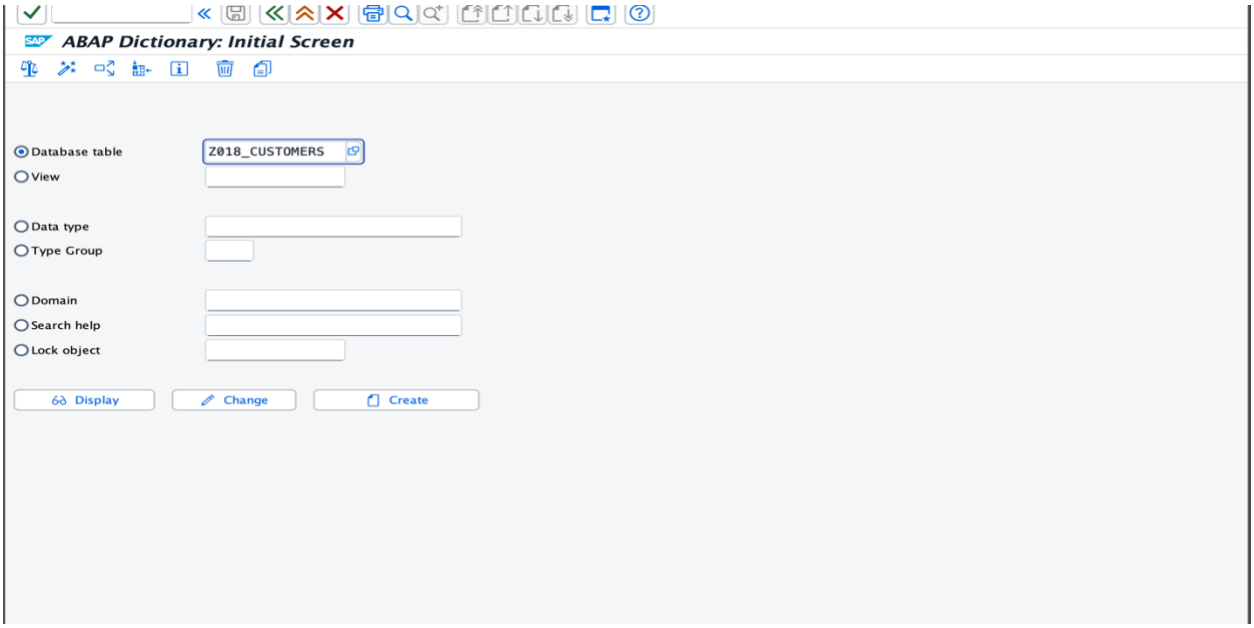
The entries have been entered for ten products are added below. Menu path: ((Menu •) Utilities • Table Contents • Display)



MANDT	BICYCLE_ID	BICYCLE_NAME	FRAME_SIZE_CM	GEAR_TYPE	PRICE	CURRENCY
205	0000000001	ROAD RUNNER	00054	MANUAL	670,00	USD
205	0000000002	MOUNTAIN KING	00058	AUTO	780,00	USD
205	0000000003	ECO RIDE	00050	MANUAL	800,00	USD
205	0000000004	CITY SWIFT	00056	MANUAL	850,00	USD
205	0000000005	TURBO JET	00052	AUTO	759,00	USD
205	0000000006	CUBE AIM RACE	00045	MANUAL	800,00	USD
205	0000000007	POLYGON CASCADE 4	00051	AUTO	710,00	USD
205	0000000008	SCOTT ASPECT 950	00052	MANUAL	620,00	USD
205	0000000009	SPECIALIZED ROCKHOPPER	00056	AUTO	829,00	USD
205	0000000010	NORCO STORM 5	00049	MANUAL	749,00	USD

2.2. Z018_CUSTOMERS

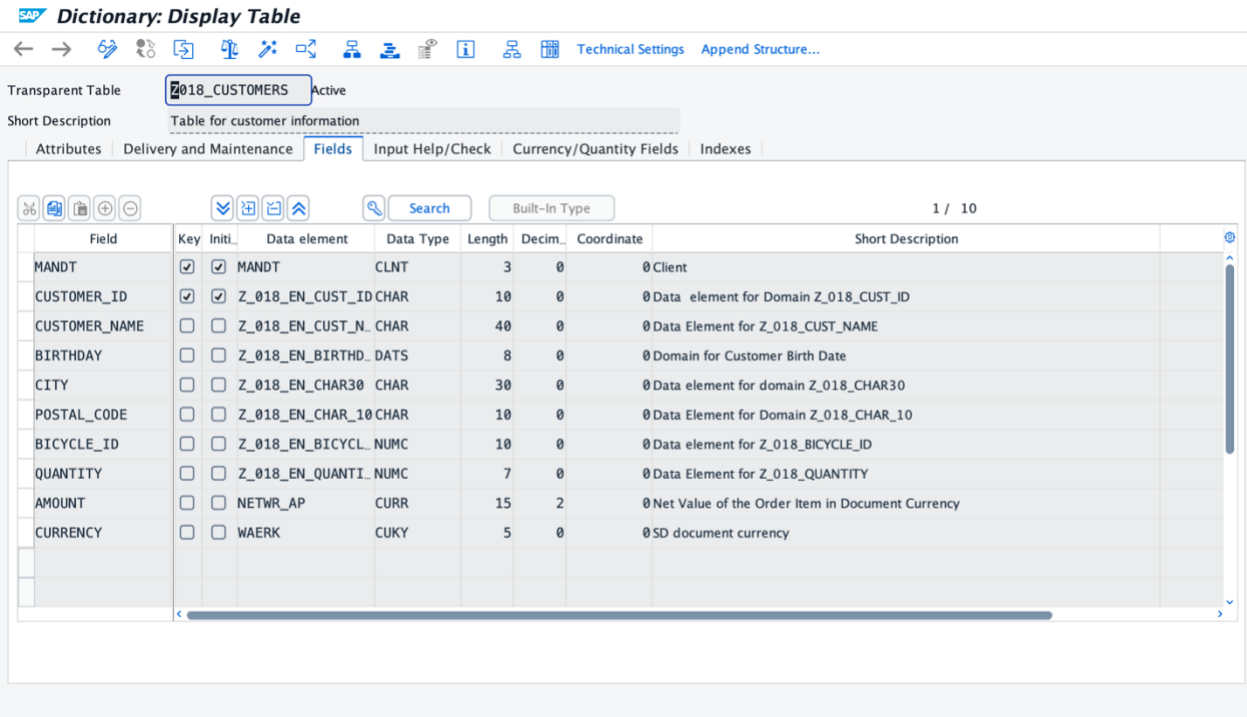
Creating another database table called Z18_CUSTOMERS to store the customers which contains all the relevant data. Maintain the short text and then choose ‘A’ Application table (master and transaction data) as the Delivery Class and Display/Maintenance Allowed as the value for Data Browser/Table View Editing.



The fields of the table are maintained by first defining the field **MANDT** with the data element **MANDT**. The checkboxes for **Key** and **Initial Value** are selected for this field, as it is required for client-dependent tables.
Next, the field **Customer_ID** is defined and designated as a primary key. The checkboxes for **Key** and **Initial Value** are also selected, and the corresponding data element is assigned to this field.
The following fields are included in the customer database table:

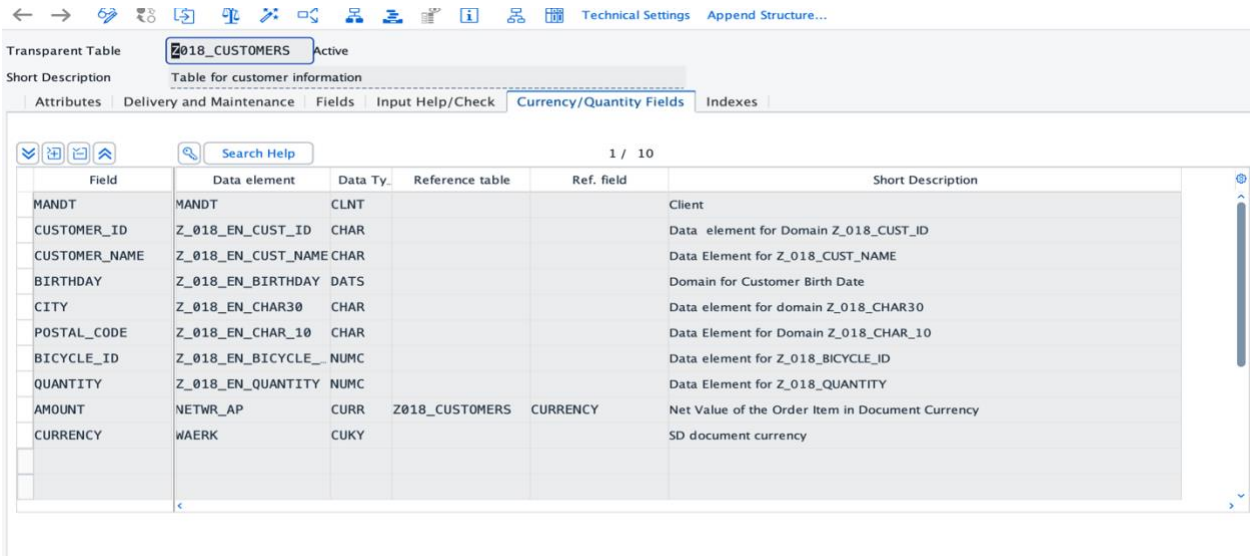
- **MANDT** (Client)
- **Customer_ID** (Customer Number)
- **Customer_Name** (Name of the Customer)
- **Birthday** (Date of Birth)
- **City** (City of Residence)
- **Postal_Code** (Postal Code)
- **Bicycle_ID** (Bicycle Identifier)
- **Quantity** (Quantity Purchased)
- **Amount** (Purchase Amount)
- **Currency** (Currency Type)

Switching to the 'Fields' tab we enter the fields of our table, below the fields are from SAP are added for the customer table.



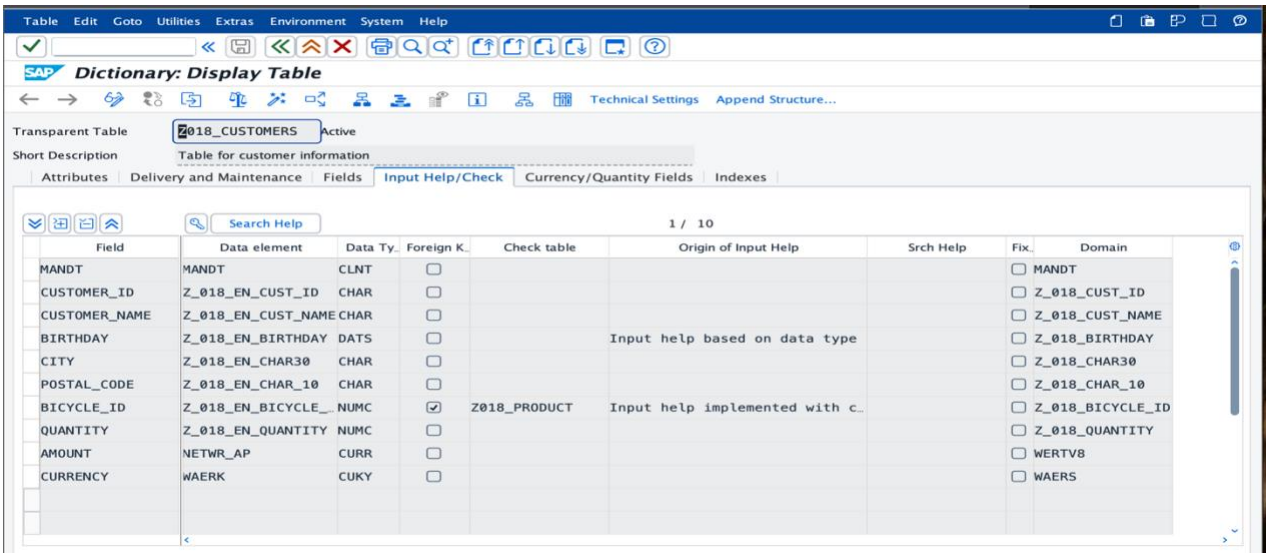
Field	Key	Initi	Data element	Data Type	Length	Decim	Coordinate	Short Description
MANDT	☑	☑	MANDT	CLNT	3	0	0	Client
CUSTOMER_ID	☑	☑	Z_018_EN_CUST_ID	CHAR	10	0	0	Data element for Domain Z_018_CUST_ID
CUSTOMER_NAME	☐	☐	Z_018_EN_CUST_N	CHAR	40	0	0	Data Element for Z_018_CUST_NAME
BIRTHDAY	☐	☐	Z_018_EN_BIRTHD	DATS	8	0	0	Domain for Customer Birth Date
CITY	☐	☐	Z_018_EN_CHAR30	CHAR	30	0	0	Data element for domain Z_018_CHAR30
POSTAL_CODE	☐	☐	Z_018_EN_CHAR_10	CHAR	10	0	0	Data Element for Domain Z_018_CHAR_10
BICYCLE_ID	☐	☐	Z_018_EN_BICYCL	NUMC	10	0	0	Data element for Z_018_BICYCLE_ID
QUANTITY	☐	☐	Z_018_EN_QUANTI	NUMC	7	0	0	Data Element for Z_018_QUANTITY
AMOUNT	☐	☐	NETWR_AP	CURR	15	2	0	Net Value of the Order Item in Document Currency
CURRENCY	☐	☐	WAERK	CUKY	5	0	0	SD document currency

Furthermore, in the ABAP Dictionary, it is necessary to assign a currency or a quantity field if the table contains a currency or quantity value. Here, in the currency field the reference table and reference field Z018_Customers and Currency field is set. It is necessary to choose reference currency from a customizing table.

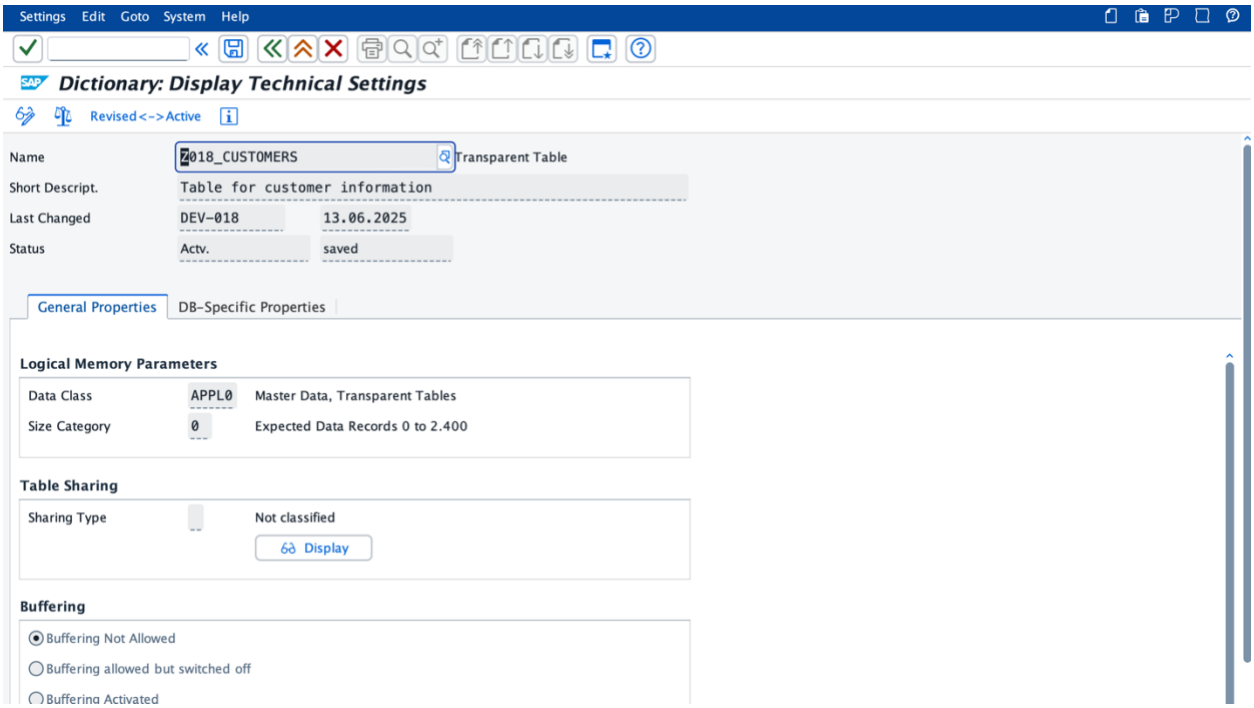


The foreign key relationship between the two tables is required to be established immediately. The **Bicycle_ID** field in the **Z018_CUSTOMERS** table is linked to the **Z018_PRODUCT** table. To do this, the **Bicycle_ID** field is selected, and the **Foreign Key** button in the toolbar is pressed. A pop-up window is displayed, requesting the check table. The product table **Z018_PRODUCT** is then selected, and the Enter key is pressed to confirm.

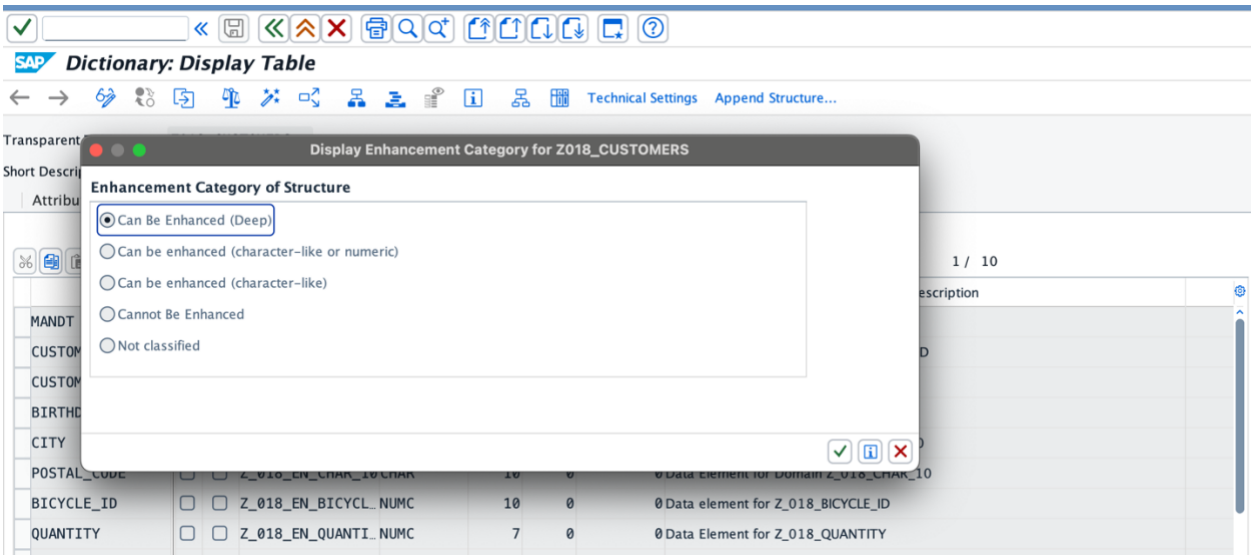
The repository information is automatically read by the system, and a foreign key definition is proposed based on the matching field names in both tables. Furthermore, the same data element must be used in the fields to define the foreign key successfully. In this case, the data element **Z_018_EN_Bicycle_ID** is assigned to the **Bicycle_ID** field in both tables to ensure this consistency.



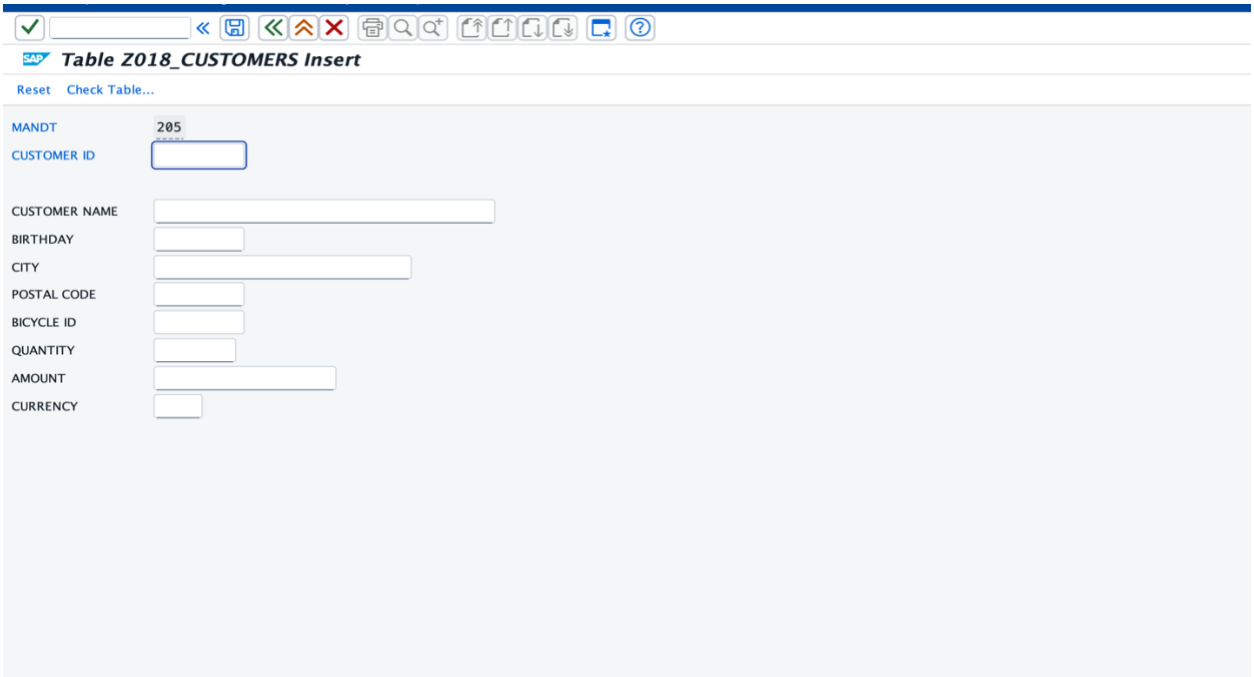
In the technical settings the I have set data class and size category. APPL0 as a data type and 0 as the size category has been chosen. After that, enhancement category has been defined as Can be Enhanced (Deep).



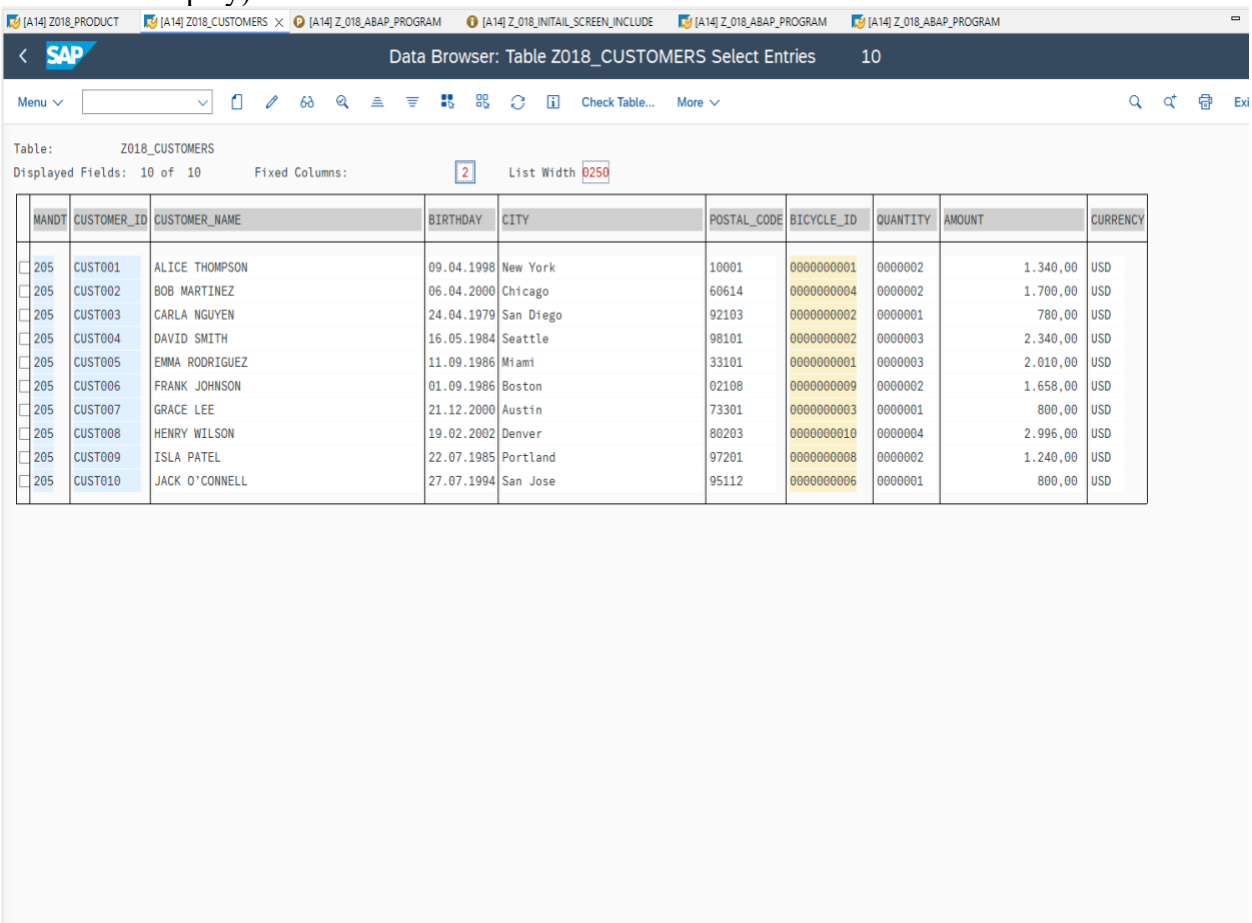
The enhancement category can be defined using following menu path: (Menu •) Extras • Enhancement Category.



In the next step some records have been entered in the table using the menu path (Menu • Utilities • Table Contents • Create Entries). Finally, Save, check, and activate the table.



The entries those had been entered can be checked using the menu path (Menu • Utilities • Table Contents • Display)



MANDT	CUSTOMER_ID	CUSTOMER_NAME	BIRTHDAY	CITY	POSTAL_CODE	BICYCLE_ID	QUANTITY	AMOUNT	CURRENCY
205	CUST001	ALICE THOMPSON	09.04.1998	New York	10001	0000000001	0000002	1.340,00	USD
205	CUST002	BOB MARTINEZ	06.04.2000	Chicago	60614	0000000004	0000002	1.700,00	USD
205	CUST003	CARLA NGUYEN	24.04.1979	San Diego	92103	0000000002	0000001	780,00	USD
205	CUST004	DAVID SMITH	16.05.1984	Seattle	98101	0000000002	0000003	2.340,00	USD
205	CUST005	EMMA RODRIGUEZ	11.09.1986	Miami	33101	0000000001	0000003	2.010,00	USD
205	CUST006	FRANK JOHNSON	01.09.1986	Boston	02108	0000000009	0000002	1.658,00	USD
205	CUST007	GRACE LEE	21.12.2000	Austin	73301	0000000003	0000001	800,00	USD
205	CUST008	HENRY WILSON	19.02.2002	Denver	80203	0000000010	0000004	2.996,00	USD
205	CUST009	ISLA PATEL	22.07.1985	Portland	97201	0000000008	0000002	1.240,00	USD
205	CUST010	JACK O'CONNELL	27.07.1994	San Jose	95112	0000000006	0000001	800,00	USD

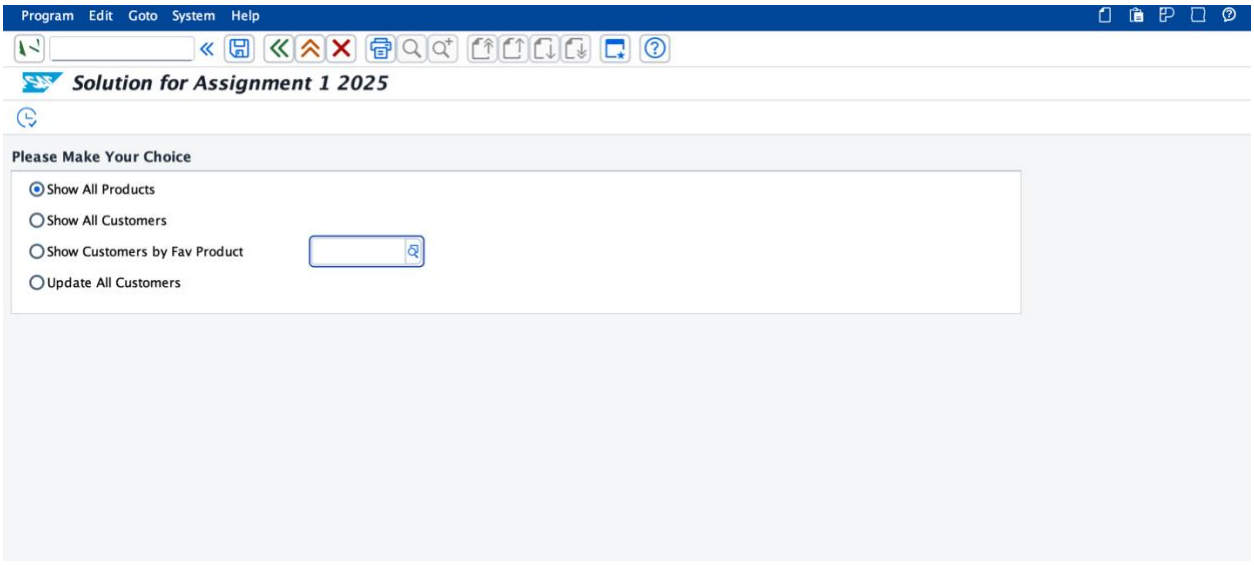
3. SOURCE CODE LIBRARY

3.1. Radio Buttons

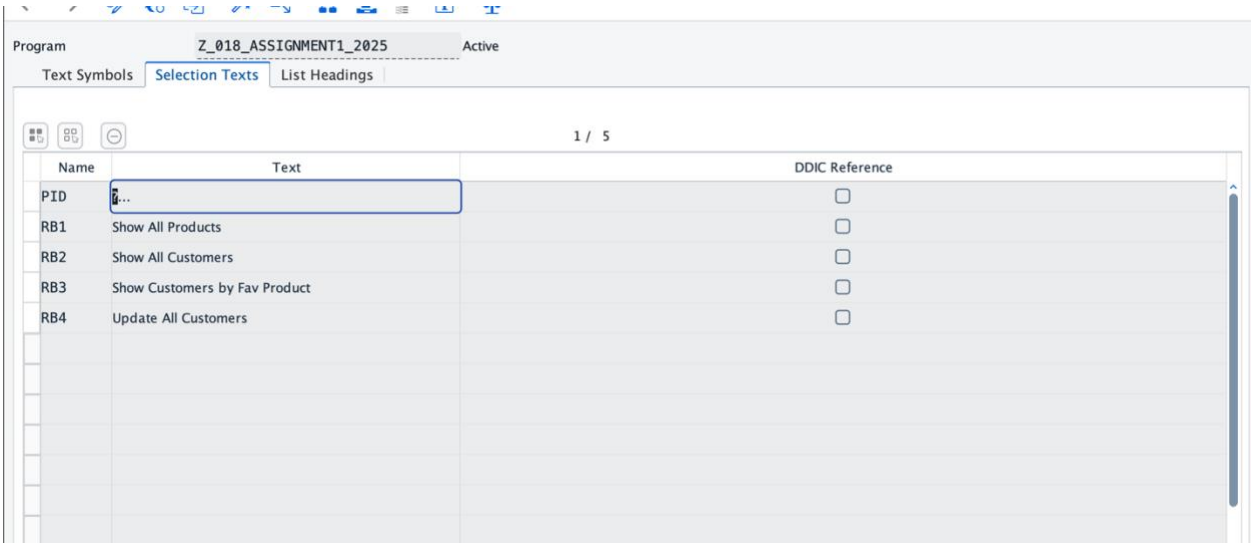
Create an input screen with four radio buttons using PARAMETERS

```
  * Begin selection screen block with title text-000
SELECTION-SCREEN: BEGIN OF BLOCK b WITH FRAME TITLE TEXT-000.
*Radio button in group 'a'
  SELECTION-SCREEN BEGIN OF LINE.
    PARAMETERS rb1 RADIOBUTTON GROUP a.
    SELECTION-SCREEN COMMENT 3(18) FOR FIELD rb1.
  SELECTION-SCREEN END OF LINE.

  SELECTION-SCREEN BEGIN OF LINE.
    PARAMETERS rb2 RADIOBUTTON GROUP a.
    SELECTION-SCREEN COMMENT 3(18) FOR FIELD rb2.

  SELECTION-SCREEN END OF LINE.

  SELECTION-SCREEN BEGIN OF LINE.
    PARAMETERS rb3 RADIOBUTTON GROUP a.
    SELECTION-SCREEN COMMENT 3(30) FOR FIELD rb3.
    PARAMETERS pid TYPE z018_customers-bicycle_id MODIF ID rb3.

  SELECTION-SCREEN END OF LINE.

  SELECTION-SCREEN BEGIN OF LINE.
    PARAMETERS rb4 RADIOBUTTON GROUP a.
    SELECTION-SCREEN COMMENT 3(23) FOR FIELD rb4.
  SELECTION-SCREEN END OF LINE.

SELECTION-SCREEN END OF BLOCK b.
```

Output:



The parameters have been changed in the selection text.



3.2. Includes

```
* Include additional logic for each radio button
INCLUDE z_018_radiobutton_include_1.
INCLUDE z_018_radiobutton_2.
INCLUDE z_018_radiobutton_3.
INCLUDE z_018_radiobutton_4.
```

These are 4 includes for 4 radio buttons. First to show all products, second is to show all customers, third to show customer by favorite product and lastly to update the customer amount in customer table.

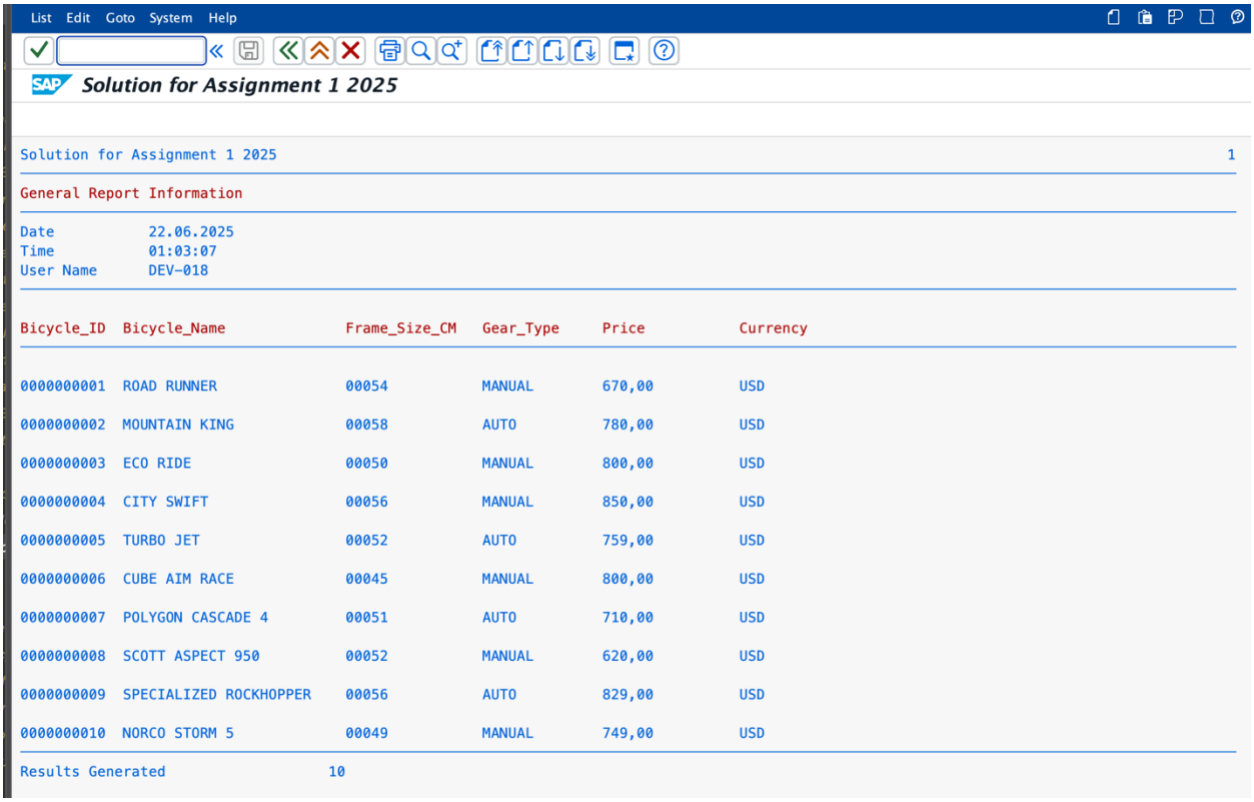
i. Show All Products (Radio button 1)

If the first radio button 1 is selected, report has to output all the product records and all columns from the product table. **(1st Include)**

```
*& Include z_018_radiobutton_include_1
*&---------------------------------------------------------------------*
* Display report title with inverted color
 WRITE: / TEXT-001 COLOR 6 INVERSE ON.
 ULINE.
* Display current date, time, user name, and underline
 WRITE: / TEXT-002, sy-datum DD/MM/YYYY,
       / TEXT-003, sy-uzeit,
       / TEXT-004, sy-uname,
       / sy-uline.
* Form to display all product records from Z018_PRODUCT table
 FORM show_products.
   " Internal table and work area to hold product records
   DATA: it_products TYPE TABLE OF z018_product,
         wa_products TYPE z018_product.

   SELECT * INTO TABLE it_products FROM z018_product.
   IF sy-subrc = 0.
     "Header
     WRITE: /.
     POSITION 1.  WRITE TEXT-005 COLOR 6 INVERSE ON. "Bicycle_ID
     POSITION 13. WRITE TEXT-006 COLOR 6 INVERSE ON. "Bicycle_Name
     POSITION 39. WRITE TEXT-007 COLOR 6 INVERSE ON. "Frame_Size_CM
     POSITION 55. WRITE TEXT-008 COLOR 6 INVERSE ON. "Gear_Type
     POSITION 69. WRITE TEXT-009 COLOR 6 INVERSE ON. "Price
     POSITION 85. WRITE TEXT-010 COLOR 6 INVERSE ON. "Currency

     ULINE.
     " Loop through each product and display its data
     LOOP AT it_products INTO wa_products.
       WRITE: /.
       POSITION 1.  WRITE wa_products-bicycle_id.
       POSITION 13. WRITE wa_products-bicycle_name.
       POSITION 39. WRITE wa_products-frame_size_cm.
       POSITION 55. WRITE wa_products-gear_type.
       POSITION 69. WRITE wa_products-price LEFT-JUSTIFIED.
       POSITION 85. WRITE wa_products-currency LEFT-JUSTIFIED.
     ENDLOOP.
     ULINE.
     " Display total number of products shown
     WRITE: / TEXT-011, sy-dbcnt.
   ENDIF.
 ENDFORM.
```

```
*Start of selection
START-OF-SELECTION.

  IF rb1 = 'X'.
    PERFORM show_products.
  ENDIF.
```

Output:



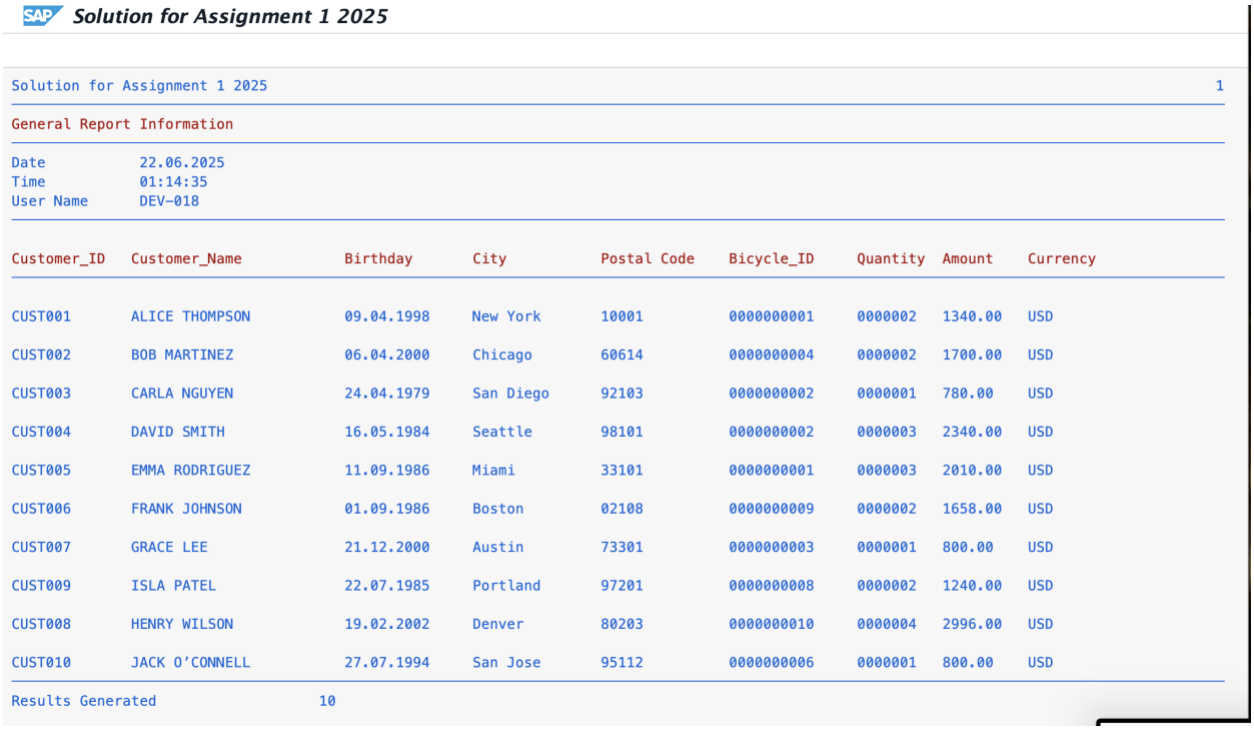
ii. Show All Customers (Radio Button 2)

If the second radio button is selected, the report must output all the customers records and all columns from the invoices table. **(2nd Include)**

```
*& Include z_018_radiobutton_2
*&---------------------------------------------------------------------*
" Internal table and variables for storing and formatting customer data
FORM show_customers.
  DATA:
      it_customers TYPE TABLE OF z018_customers,
      wa_customers TYPE z018_customers,
      lv_birthday_str TYPE string,
      lv_quantity_str TYPE string,
      lv_amount_str TYPE string.
 " Select all customers from database
  SELECT * INTO TABLE it_customers FROM z018_customers.
  IF sy-subrc = 0.
 " Header with positions and inverse color
    WRITE: /.
    POSITION 1.  WRITE TEXT-012 COLOR 6 INVERSE ON. "Customer_ID
    POSITION 15. WRITE TEXT-013 COLOR 6 INVERSE ON. "Customer_Name
    POSITION 40. WRITE TEXT-014 COLOR 6 INVERSE ON. "Birthday
    POSITION 55. WRITE TEXT-015 COLOR 6 INVERSE ON. "City
    POSITION 70. WRITE TEXT-016 COLOR 6 INVERSE ON. "Postal_Code
    POSITION 85. WRITE TEXT-005 COLOR 6 INVERSE ON. "Bicycle_ID
    POSITION 100. WRITE TEXT-017 COLOR 6 INVERSE ON. "Quantity
    POSITION 110.WRITE TEXT-018 COLOR 6 INVERSE ON. "Amount
    POSITION 120.WRITE TEXT-010 COLOR 6 INVERSE ON. "Currency

    ULINE.

    LOOP AT it_customers INTO wa_customers.
      " Format Birthday (date) to DD/MM/YYYY string
      lv_birthday_str = |{ wa_customers-birthday DATE = USER }|.
      " Convert Quantity (numeric) to string without leading zeros
      lv_quantity_str = CONV string( wa_customers-quantity ).
      " Format Amount (numeric) with 2 decimals, convert to string
      lv_amount_str = |{ wa_customers-amount DECIMALS = 2 }|.
 " Display customer data
      WRITE: /.
      POSITION 1.  WRITE wa_customers-customer_id LEFT-JUSTIFIED.
      POSITION 15. WRITE wa_customers-customer_name LEFT-JUSTIFIED.
      POSITION 40. WRITE lv_birthday_str LEFT-JUSTIFIED.
      POSITION 55. WRITE wa_customers-city LEFT-JUSTIFIED.
      POSITION 70. WRITE wa_customers-postal_code LEFT-JUSTIFIED.
      POSITION 85. WRITE wa_customers-bicycle_id LEFT-JUSTIFIED.
      POSITION 100. WRITE lv_quantity_str LEFT-JUSTIFIED.
      POSITION 110.WRITE lv_amount_str LEFT-JUSTIFIED.
      POSITION 120.WRITE wa_customers-currency LEFT-JUSTIFIED.
    ENDLOOP.
```

```
    ULINE.
    WRITE: / TEXT-011, sy-dbcnt. " Show number of records
  ENDIF.
ENDFORM.

* Start of Selection
START-OF-SELECTION.
IF rb2 = 'X'.
  PERFORM show_customers.
ENDIF.
```

Output:



Solution for Assignment 1 2025

Solution for Assignment 1 2025 1

General Report Information

Date 22.06.2025
Time 01:14:35
User Name DEV-018

Customer_ID	Customer_Name	Birthday	City	Postal Code	Bicycle_ID	Quantity	Amount	Currency
CUST001	ALICE THOMPSON	09.04.1998	New York	10001	0000000001	0000002	1340.00	USD
CUST002	BOB MARTINEZ	06.04.2000	Chicago	60614	0000000004	0000002	1700.00	USD
CUST003	CARLA NGUYEN	24.04.1979	San Diego	92103	0000000002	0000001	780.00	USD
CUST004	DAVID SMITH	16.05.1984	Seattle	98101	0000000002	0000003	2340.00	USD
CUST005	EMMA RODRIGUEZ	11.09.1986	Miami	33101	0000000001	0000003	2010.00	USD
CUST006	FRANK JOHNSON	01.09.1986	Boston	02108	0000000009	0000002	1658.00	USD
CUST007	GRACE LEE	21.12.2000	Austin	73301	0000000003	0000001	800.00	USD
CUST009	ISLA PATEL	22.07.1985	Portland	97201	0000000008	0000002	1240.00	USD
CUST008	HENRY WILSON	19.02.2002	Denver	80203	0000000010	0000004	2996.00	USD
CUST010	JACK O'CONNELL	27.07.1994	San Jose	95112	0000000006	0000001	800.00	USD

Results Generated 10

iii. Show All Customers by Favorite Product (Radio Button 3)

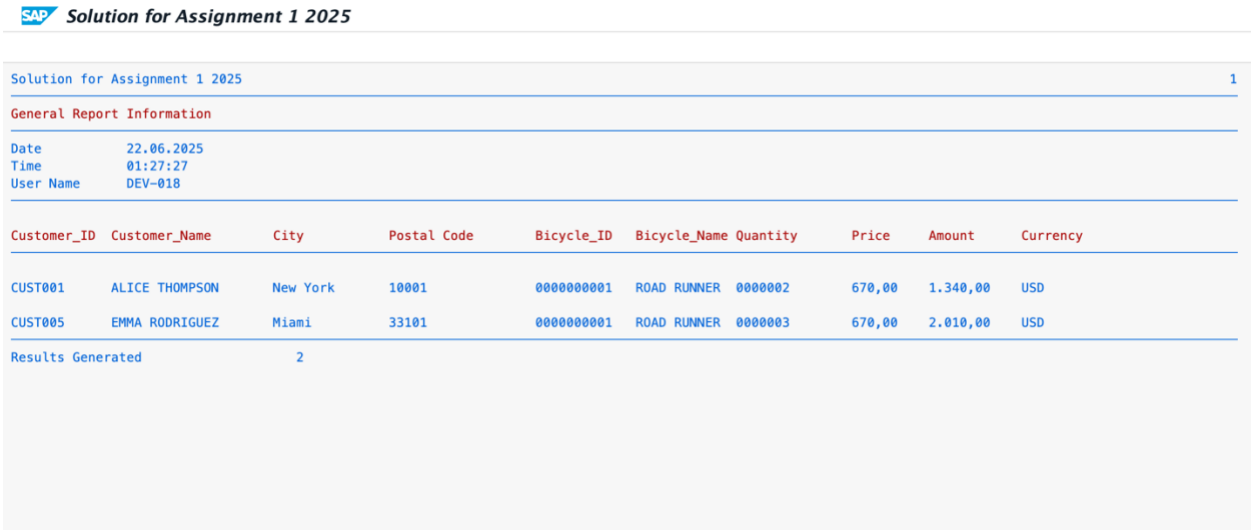
If the third radio button is selected, the report should present only those customers that prefer the product you enter in a text parameter box. In addition, all columns from both tables should be displayed. **(3rd Include)**

```
*& Include z_018_radiobutton_3
*&---------------------------------------------------------------------*

" Define a structure combining customer and product fields for joined data
FORM display_customers_by_product.

  TYPES: BEGIN OF ty_joined,
         customer_id   TYPE z_018_en_cust_id,
         customer_name TYPE z_018_en_cust_name,
         birthday      TYPE z_018_en_birthday,
         city          TYPE z_018_en_char30,
         postal_code   TYPE z_018_en_char_10,
         bicycle_id    TYPE z_018_en_bicycle_id,
         quantity      TYPE z_018_en_quantity,
         amount        TYPE netwr_ap,
         currency      TYPE waerk,
         bicycle_name  TYPE z_018_en_bicycle_name,
         frame_size_cm TYPE z_018_frame_size,
         gear_type     TYPE z_018_en_gear_type,
         price         TYPE netwr_ap,
       END OF ty_joined.
" Declare internal table and work area for joined data, and counter variable
  DATA: it_joined TYPE STANDARD TABLE OF ty_joined,
        wa_joined TYPE ty_joined,
        lv_count  TYPE i.
" Select customer and product data where bicycle IDs match and equal parameter pid
  SELECT
    a~customer_id, a~customer_name, a~birthday, a~city, a~postal_code,
    a~bicycle_id, a~quantity, a~amount, a~currency,
    b~bicycle_name, b~frame_size_cm, b~gear_type, b~price
    INTO TABLE @it_joined
    FROM z018_customers AS a
    INNER JOIN z018_product AS b
    ON a~bicycle_id = b~bicycle_id
    WHERE b~bicycle_id = @pid.

  " Print header with field names in inverse color at specific positions
  WRITE: /.
  POSITION 1.   WRITE TEXT-012 COLOR 6 INVERSE ON. "Customer_ID
  POSITION 14.  WRITE TEXT-013 COLOR 6 INVERSE ON. "Customer_Name
  POSITION 35.  WRITE TEXT-015 COLOR 6 INVERSE ON. "City
  POSITION 50.  WRITE TEXT-016 COLOR 6 INVERSE ON. "Postal_Code
  POSITION 69.  WRITE TEXT-005 COLOR 6 INVERSE ON. "Bicycle_ID
  POSITION 82.  WRITE TEXT-006 COLOR 6 INVERSE ON. "Bicycle_Name
  POSITION 95. WRITE TEXT-017 COLOR 6 INVERSE ON. "Quantity
  POSITION 110. WRITE TEXT-009 COLOR 6 INVERSE ON. "Price
```

```
  POSITION 120. WRITE TEXT-018 COLOR 6 INVERSE ON. "Amount
  POSITION 132. WRITE TEXT-010 COLOR 6 INVERSE ON. "Currency
  ULINE.
  " Loop through joined data and display each record
  LOOP AT it_joined INTO wa_joined.
    ADD 1 TO lv_count.

    WRITE: /.
    POSITION 1.   WRITE wa_joined-customer_id LEFT-JUSTIFIED.
    POSITION 14.  WRITE wa_joined-customer_name LEFT-JUSTIFIED.
    POSITION 35.  WRITE wa_joined-city LEFT-JUSTIFIED.
    POSITION 50.  WRITE wa_joined-postal_code LEFT-JUSTIFIED.
    POSITION 69.  WRITE wa_joined-bicycle_id LEFT-JUSTIFIED.
    POSITION 82.  WRITE wa_joined-bicycle_name LEFT-JUSTIFIED.
    POSITION 95. WRITE wa_joined-quantity LEFT-JUSTIFIED.
    POSITION 110. WRITE wa_joined-price LEFT-JUSTIFIED.
    POSITION 120. WRITE wa_joined-amount LEFT-JUSTIFIED.
    POSITION 132. WRITE wa_joined-currency LEFT-JUSTIFIED.
  ENDLOOP.

  ULINE.
  WRITE: / TEXT-011, lv_count. " e.g., 'Total Records: 5'

ENDFORM.

START-OF-SELECTION.
  IF rb3 = 'X'.
    PERFORM display_customers_by_product.
  ENDIF.
```

Output:

In the selection screen I have selected Bicycle_ID 1, and the output screen is showing me the results of customers those bought this bicycle.



iv. Update Customer Amount (Radio Button 4)

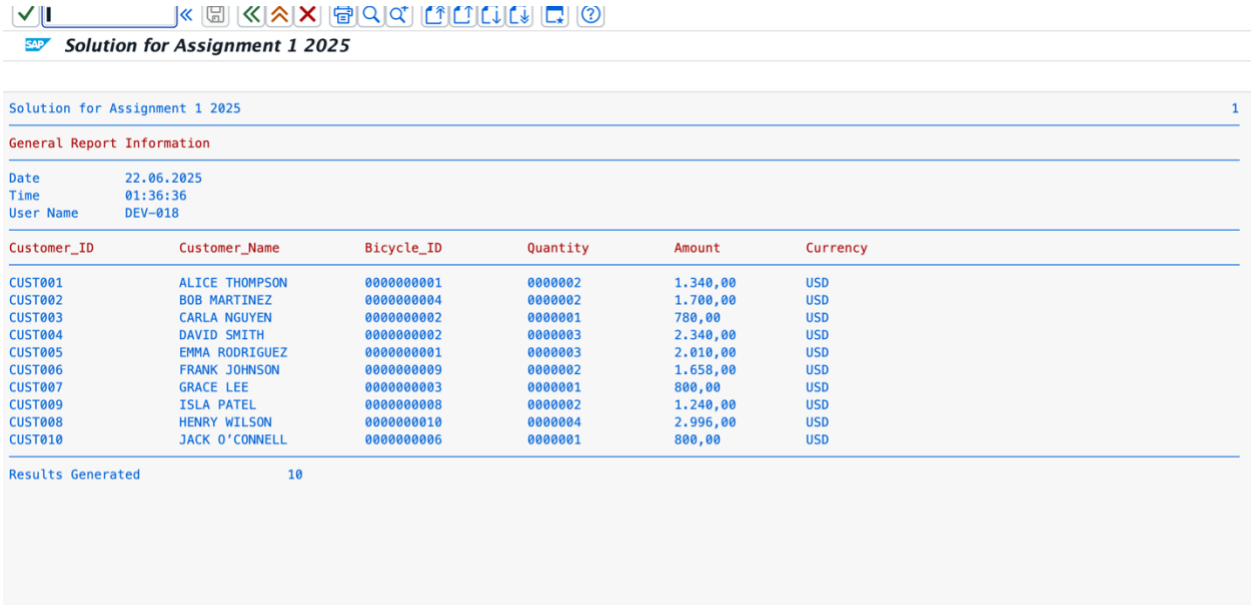
If the fourth radio button is selected, the system has to recalculate and update the amount in the customer table. Multiply the product quantity from the customer table and the product price from the product table and store the result in the customer table. During calculation output all the product records with the new calculated amount should be displayed. **(4th Include)**

```
*& Include z_018_radiobutton_4
*&---------------------------------------------------------------------*
*&---------------------------------------------------------------------*

FORM update_customer_amounts.
  " Counter for number of updated records
  DATA: lv_updated TYPE i VALUE 0.
  DATA: it_customers TYPE TABLE OF z018_customers,
        wa_customers TYPE z018_customers,
        lv_price     TYPE netwr_ap.

  " Retrieve all customer records from the database
  SELECT * INTO TABLE it_customers FROM z018_customers.

  " Print table header
  WRITE: /
         TEXT-012 COLOR 6 INVERSE ON,         "Customer_ID
         TEXT-013 COLOR 6 INVERSE ON,         "Customer_Name
         TEXT-005 COLOR 6 INVERSE ON,         "Bicycle_ID
         TEXT-017 COLOR 6 INVERSE ON,         "Quantity
         TEXT-018 COLOR 6 INVERSE ON,         "Amount
         TEXT-010 COLOR 6 INVERSE ON.         "Currency
  ULINE.

  LOOP AT it_customers INTO wa_customers.

    " Get price of the product corresponding to the customer's bicycle
    SELECT SINGLE price INTO @lv_price
      FROM z018_product
      WHERE bicycle_id = @wa_customers-bicycle_id.
    " If matching product found, calculate and update amount
    IF sy-subrc = 0.
      wa_customers-amount = wa_customers-quantity * lv_price.
      MODIFY z018_customers FROM wa_customers.
      ADD 1 TO lv_updated.
    ENDIF.
    " Display the updated customer information
    WRITE: / wa_customers-customer_id   UNDER TEXT-012 LEFT-JUSTIFIED,
           wa_customers-customer_name UNDER TEXT-013 LEFT-JUSTIFIED,
           wa_customers-bicycle_id    UNDER TEXT-005 LEFT-JUSTIFIED,
           wa_customers-quantity      UNDER TEXT-017 LEFT-JUSTIFIED,
           wa_customers-amount        UNDER TEXT-018 LEFT-JUSTIFIED,
           wa_customers-currency      UNDER TEXT-010 LEFT-JUSTIFIED.
```

```
ENDLOOP.
  ULINE.
  " Display total number of updated records
  WRITE: / TEXT-011, lv_updated.

ENDFORM.

START-OF-SELECTION.
  IF rb4 = 'X'.
    PERFORM update_customer_amounts.
  ENDIF.
```

Output:

It has calculated the amount of the purchase that each customer has made.



Solution for Assignment 1 2025

Solution for Assignment 1 2025 1

General Report Information

Date 22.06.2025
Time 01:36:36
User Name DEV-018

Customer_ID	Customer_Name	Bicycle_ID	Quantity	Amount	Currency
CUST001	ALICE THOMPSON	0000000001	0000002	1.340,00	USD
CUST002	BOB MARTINEZ	0000000004	0000002	1.700,00	USD
CUST003	CARLA NGUYEN	0000000002	0000001	780,00	USD
CUST004	DAVID SMITH	0000000002	0000003	2.340,00	USD
CUST005	EMMA RODRIGUEZ	0000000001	0000003	2.010,00	USD
CUST006	FRANK JOHNSON	0000000009	0000002	1.658,00	USD
CUST007	GRACE LEE	0000000003	0000001	800,00	USD
CUST009	ISLA PATEL	0000000008	0000002	1.240,00	USD
CUST008	HENRY WILSON	0000000010	0000004	2.996,00	USD
CUST010	JACK O'CONNELL	0000000006	0000001	800,00	USD

Results Generated 10

4. (Time, Date, Username)

For showing the username, time and date these codes are written on top of the includes.

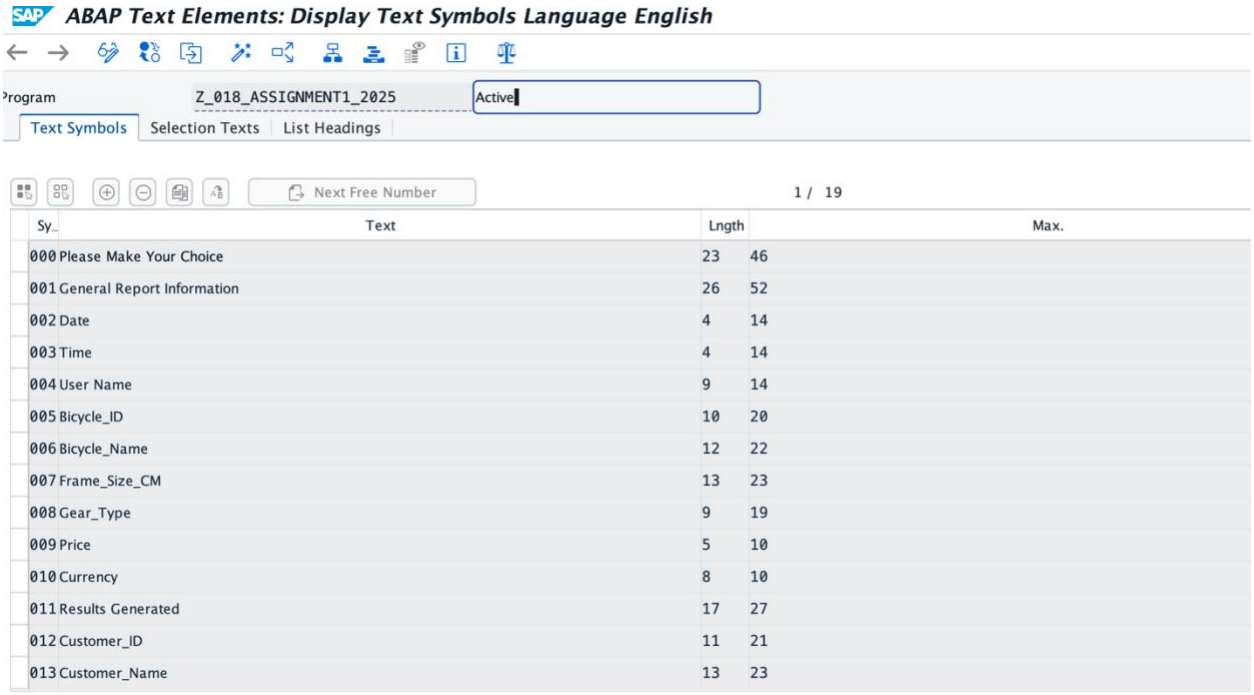
```
WRITE: / TEXT-001 COLOR 6 INVERSE ON.
 uline.
  WRITE: / TEXT-002, sy-datum DD/MM/YYYY,
         / TEXT-003, sy-uzeit,
         / TEXT-004, sy-uname,
        / sy-uline.
```

Output:

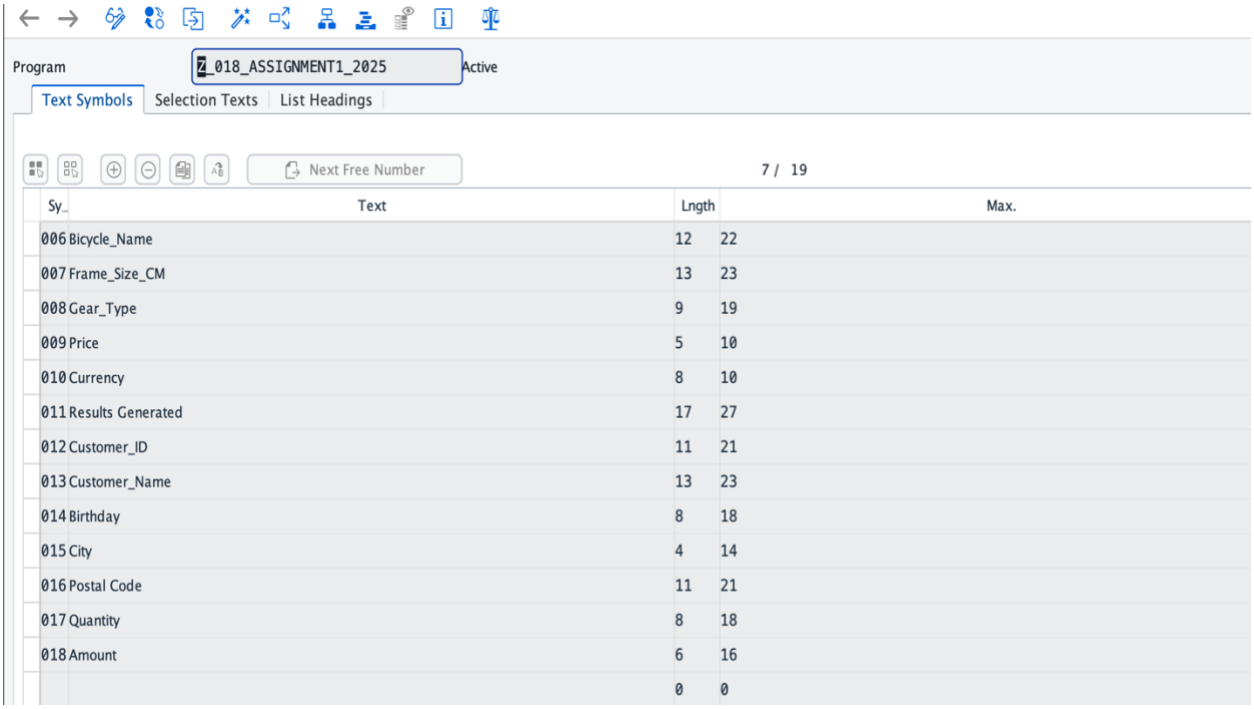
```
General Report Information

Date          18.06.2025
Time          18:18:37
User Name     DEV-018
```

5. Text-Elements

The purpose of text elements is to be used as constants or labels for titles and messages and are defined according to the language of the session.



Sy..	Text	Lngth	Max.
000	Please Make Your Choice	23	46
001	General Report Information	26	52
002	Date	4	14
003	Time	4	14
004	User Name	9	14
005	Bicycle_ID	10	20
006	Bicycle_Name	12	22
007	Frame_Size_CM	13	23
008	Gear_Type	9	19
009	Price	5	10
010	Currency	8	10
011	Results Generated	17	27
012	Customer_ID	11	21
013	Customer_Name	13	23



Program Z_018_ASSIGNMENT1_2025 Active

Text Symbols | Selection Texts | List Headings

Next Free Number 7 / 19

Sy..	Text	Lngth	Max.
006	Bicycle_Name	12	22
007	Frame_Size_CM	13	23
008	Gear_Type	9	19
009	Price	5	10
010	Currency	8	10
011	Results Generated	17	27
012	Customer_ID	11	21
013	Customer_Name	13	23
014	Birthday	8	18
015	City	4	14
016	Postal Code	11	21
017	Quantity	8	18
018	Amount	6	16
		0	0

6. Overview of the Code

```
*& Report z_018_assignment1_2025
*&---------------------------------------------------------------------*
*&
*&---------------------------------------------------------------------*
REPORT z_018_assignment1_2025.
* Begin selection screen block with title text-000
SELECTION-SCREEN: BEGIN OF BLOCK b WITH FRAME TITLE TEXT-000.
*Radio button in group 'a'
  SELECTION-SCREEN BEGIN OF LINE.
    PARAMETERS rb1 RADIOBUTTON GROUP a.
    SELECTION-SCREEN COMMENT 3(18) FOR FIELD rb1.
  SELECTION-SCREEN END OF LINE.

  SELECTION-SCREEN BEGIN OF LINE.
    PARAMETERS rb2 RADIOBUTTON GROUP a.
    SELECTION-SCREEN COMMENT 3(18) FOR FIELD rb2.

  SELECTION-SCREEN END OF LINE.

  SELECTION-SCREEN BEGIN OF LINE.
    PARAMETERS rb3 RADIOBUTTON GROUP a.
    SELECTION-SCREEN COMMENT 3(30) FOR FIELD rb3.
    PARAMETERS  pid TYPE z018_customers-bicycle_id MODIF ID rb3.

  SELECTION-SCREEN END OF LINE.

  SELECTION-SCREEN BEGIN OF LINE.
    PARAMETERS rb4 RADIOBUTTON GROUP a.
    SELECTION-SCREEN COMMENT 3(23) FOR FIELD rb4.
  SELECTION-SCREEN END OF LINE.

SELECTION-SCREEN END OF BLOCK b.
* Include additional logic for each radio button
INCLUDE z_018_radiobutton_include_1.
INCLUDE z_018_radiobutton_2.
INCLUDE z_018_radiobutton_3.
INCLUDE z_018_radiobutton_4.
```

The codes inside the includes are added in each radio button description.

7. Problems Occurred

- Z018_Customers table was not activated as the reference table and reference field for the currency field was missing.
  (To fix this the reference table and field has been added)
- The position of the data was not properly adjusted to the text of the header and the which was solved by using the code POSITION and LEFT-JUSTIFIED.

Solution for Assignment 1 2025

Solution for Assignment 1 2025 1

General Report Information

Date 21.06.2025
Time 20:20:05
User Name DEV-018

Bicycle_ID	Bicycle_Name	Frame_Size_CM	Gear_Type	Currency	Price
0000000001	ROAD RUNNER	00054	MANUAL	USD	670,00
0000000002	MOUNTAIN KING	00058	AUTO	USD	780,00
0000000003	ECO RIDE	00050	MANUAL	USD	800,00
0000000004	CITY SWIFT	00056	MANUAL	USD	850,00
0000000005	TURBO JET	00052	AUTO	USD	759,00
0000000006	CUBE AIM RACE	00045	MANUAL	USD	800,00
0000000007	POLYGON CASCADE 4	00051	AUTO	USD	710,00
0000000008	SCOTT ASPECT 950	00052	MANUAL	USD	620,00
0000000009	SPECIALIZED ROCKHOPPER	00056	AUTO	USD	829,00
0000000010	NORCO STORM 5	00049	MANUAL	USD	749,00

Results Generated 10

After arranging and fixing the code,

Solution for Assignment 1 2025

Solution for Assignment 1 2025

General Report Information

Date 22.06.2025
Time 02:15:34
User Name DEV-018

Bicycle_ID	Bicycle_Name	Frame_Size_CM	Gear_Type	Price	Currency
0000000001	ROAD RUNNER	00054	MANUAL	670,00	USD
0000000002	MOUNTAIN KING	00058	AUTO	780,00	USD
0000000003	ECO RIDE	00050	MANUAL	800,00	USD
0000000004	CITY SWIFT	00056	MANUAL	850,00	USD
0000000005	TURBO JET	00052	AUTO	759,00	USD
0000000006	CUBE AIM RACE	00045	MANUAL	800,00	USD
0000000007	POLYGON CASCADE 4	00051	AUTO	710,00	USD
0000000008	SCOTT ASPECT 950	00052	MANUAL	620,00	USD
0000000009	SPECIALIZED ROCKHOPPER	00056	AUTO	829,00	USD
0000000010	NORCO STORM 5	00049	MANUAL	749,00	USD

Results Generated 10