

# **SAP Business Technology Platform (Part-1)**

**Presented To**

**Pro. Jürgen Weiner**

**By Team - C**

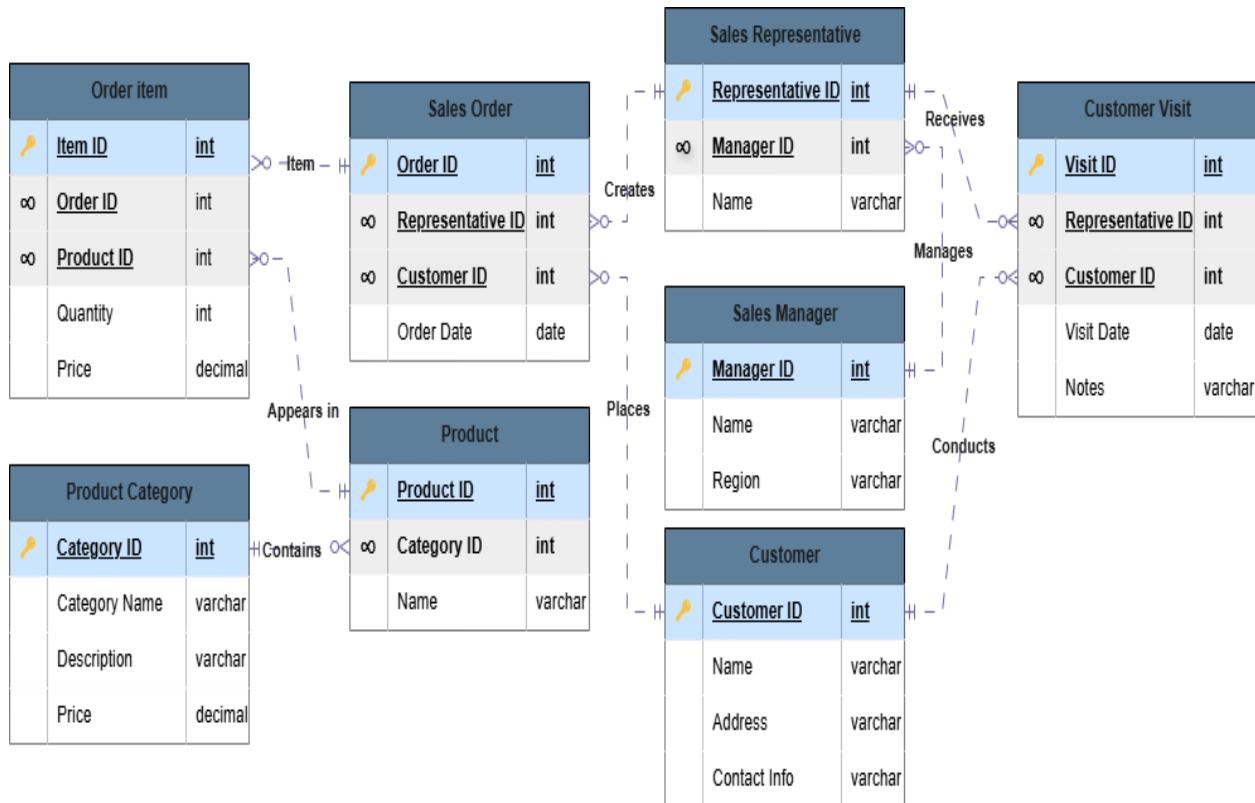
**Abdul Shukoor**

**Palash Wani**

**Tharani Natrajan**

**Use Case 3: Sales**

# 1. Entity-Relation Diagram



## 2. Create Roles

```
CREATE ROLE SALES_ADMIN;  
CREATE ROLE SALES_MANAGER;  
CREATE ROLE SALES REPRESENTATIVE;
```

The screenshot shows the SAP HANA Cloud Central interface for Role Management. The top navigation bar includes the SAP logo, instance name (HC\_PW), role management, and a refresh button. A sidebar on the left contains icons for instances, databases, and security. The main area is titled "Role Management" and shows two tabs: "Roles (3)" and "Role Groups (3)". The "Roles (3)" tab is selected. A search bar at the top right contains the text "Sales". Below the search bar is a table with three rows, each representing a role:

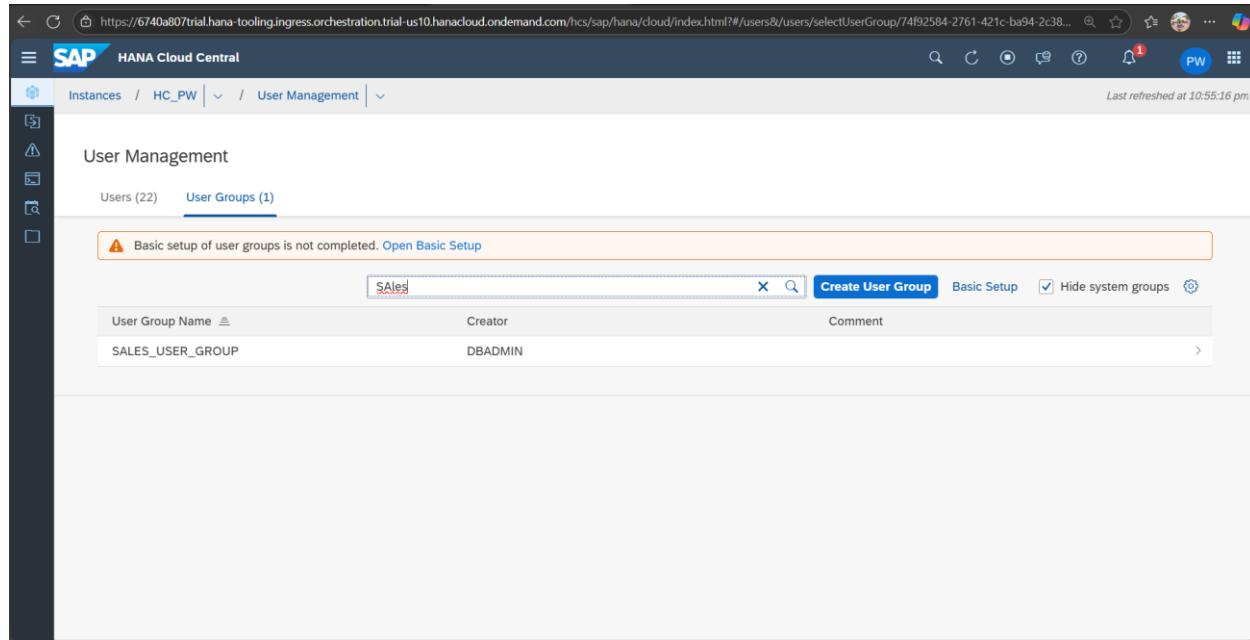
Role Name	Schema	Role Group
SALES_ADMIN	Not in any schema	Not in any group >
SALES_MANAGER	Not in any schema	Not in any group >
SALES_REPRESENTATIVE	Not in any schema	Not in any group >

A "Create Role" button is located at the top right of the table header.

### 3. Create User Group

```
CREATE USERGROUP SALES_USER_GROUP
```

```
SET PARAMETER 'minimal_password_length' = '8', 'force_first_password_change' = 'FALSE';
```



The screenshot shows the SAP HANA Cloud Central User Management interface. The URL in the browser is <https://6740a807trial.hana-tooling.ingress.orchestration.trial-us10.hanacloud.ondemand.com/fcs/sap/hana/cloud/index.html#/users&#users/selectUserGroup/74f92584-2761-421c-ba94-2c38...>. The page title is "HANA Cloud Central". The navigation bar includes "Instances", "HC\_PW", "User Management", and a timestamp "Last refreshed at 10:55:16 pm". A notification icon shows "1". The main content area is titled "User Management" and shows "Users (22)" and "User Groups (1)". A warning message "⚠ Basic setup of user groups is not completed. Open Basic Setup" is displayed. Below it, a table lists one user group: "SALES\_USER\_GROUP" (Creator: DBADMIN). At the top of the table is a search bar with "SALES" and buttons for "Create User Group", "Basic Setup", and "Hide system groups".

User Group Name	Creator	Comment
SALES_USER_GROUP	DBADMIN	

## 4. Create User

```
CREATE USER SALE_ADMIN PASSWORD Password1 no force_first_password_change SET USERGROUP  
SALES_USER_GROUP;  
CREATE USER SALES_MANAGER_1 PASSWORD Password1 no force_first_password_change SET USERGROUP  
SALES_USER_GROUP;  
CREATE USER UNI_REPRESENTATIVE_1 PASSWORD Password1 no force_first_password_change SET  
USERGROUP SALES_USER_GROUP;
```

The screenshot shows the SAP HANA Cloud Central User Management interface. The top navigation bar includes the SAP logo, HANA Cloud Central, and various icons for search, refresh, and notifications. The current path is Instances / HC\_PW / User Management. A message at the top right indicates the data was last refreshed at 10:58:05 pm. On the left, there's a sidebar with icons for instances, databases, and logs. The main area is titled 'User Management' and shows a table of users. The table has columns for User Name, User Group, User Status, and Email. A search bar at the top allows filtering by user name or email. A 'Create User' button is available, along with a checkbox for 'Hide system users'. The users listed are:

User Name	User Group	User Status	Email
WWWD_RT	BROKER_UG_HDISHARED	Active	>
HDI_TUTORIAL_68IT5NJKXV7L1RA6BX669 WW8D_RT	BROKER_UG_HDISHARED	Active	>
HDI_TUTORIAL_8ZT3VUZDCUZ3GLZGW5F QMFLFG_DT	BROKER_UG_HDISHARED	Active	>
HDI_TUTORIAL_8ZT3VUZDCUZ3GLZGW5F QMFLFG_RT	BROKER_UG_HDISHARED	Active	>
SALES_MANAGER_1	SALES_USER_GROUP	<span style="color: orange;">⚠ Initial password has expired</span>	>
SALE_ADMIN	SALES_USER_GROUP	Active	>
UNI_REPRESENTATIVE_1	SALES_USER_GROUP	<span style="color: orange;">⚠ Initial password has expired</span>	>
UPS_GRANTOR	DEFAULT	<span style="color: orange;">⚠ Initial password has expired</span>	>
UPS_GRANTOR10	DEFAULT	<span style="color: orange;">⚠ Initial password has expired</span>	>
UPS_GRANTOR11	DEFAULT	<span style="color: orange;">⚠ Initial password has expired</span>	>
UPS_GRANTOR12	DEFAULT	Active	>
UPS_GRANTOR18	DEFAULT	Active	>

## 5. Grant

```
-- Role-specific
GRANT TRUST ADMIN TO SALES_ADMIN; -- required to create a PSE
GRANT CERTIFICATE ADMIN TO SALES_ADMIN; --required to create a certificate
GRANT CREDENTIAL ADMIN TO SALES_ADMIN --required to create a credential
GRANT EXPORT TO SALES_ADMIN; --required to enable export of data
GRANT IMPORT TO SALES_ADMIN; --required to enable import of data GRANT CREATE REMOTE SOURCE
TO SALES_ADMIN; --allow setting the PSE purpose to REMOTE SOURCE and to create REMOTE SOURCES
GRANT RESOURCE ADMIN TO SALES_ADMIN; --allow viewing of diagnostic files
GRANT CATALOG READ TO SALES_ADMIN; --allow access to system views
GRANT INIFILE ADMIN TO SALES_ADMIN; --allow altering of system settings
-- User-specific
GRANT CREATE SCHEMA TO SALE_ADMIN;
GRANT SALES_ADMIN TO SALE_ADMIN;
GRANT SALES_MANAGER TO SALES_MANAGER_1;
GRANT SALES_REPRESENTATIVE TO UNI_REPRENTATIVE_1;
-- Create Schema and corresponding rights
CONNECT SALE_ADMIN PASSWORD Password1;
CREATE SCHEMA SALES;
GRANT ALL PRIVILEGES ON SCHEMA SALES TO SALES_ADMIN;
GRANT SELECT ON SCHEMA SALES TO SALES_MANAGER;
```

## 6. Create Table

```
Create Table
CONNECT SALE_ADMIN PASSWORD Password1;
SET SCHEMA SALES;

-- Sales Manager Table
CREATE TABLE SalesManager (
    ManagerID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Region VARCHAR(50) NOT NULL
);

-- Sales Representative Table
CREATE TABLE SalesRepresentative (
    RepresentativeID INT PRIMARY KEY,
    ManagerID INT NOT NULL,
    Name VARCHAR(100) NOT NULL,
    FOREIGN KEY (ManagerID) REFERENCES SalesManager(ManagerID)
);

-- Customer Table (with INT ContactInfo as requested)
CREATE TABLE Customer (
    CustomerID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Address VARCHAR(200) NOT NULL,
    ContactInfo INT NOT NULL
);

-- Product Category Table
CREATE TABLE ProductCategory (
    CategoryID INT PRIMARY KEY,
    CategoryName VARCHAR(100) NOT NULL,
    Description VARCHAR(255)
);

-- Product Table
CREATE TABLE Product (
    ProductID INT PRIMARY KEY,
    CategoryID INT NOT NULL,
    Name VARCHAR(100) NOT NULL,
    FOREIGN KEY (CategoryID) REFERENCES ProductCategory(CategoryID)
);

-- Order Table
CREATE TABLE SalesOrder (
    OrderID INT PRIMARY KEY,
    RepresentativeID INT NOT NULL,
    CustomerID INT NOT NULL,
```

```
        OrderDate DATE NOT NULL,  
        FOREIGN KEY (RepresentativeID) REFERENCES SalesRepresentative(RepresentativeID),  
        FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID)  
);
```

```
-- Order Item Table  
CREATE TABLE OrderItem (  
    ItemID INT PRIMARY KEY,  
    OrderID INT NOT NULL,  
    ProductID INT NOT NULL,  
    Quantity INT NOT NULL,  
    Price DECIMAL(10,2) NOT NULL,  
    FOREIGN KEY (OrderID) REFERENCES SalesOrder(OrderID),  
    FOREIGN KEY (ProductID) REFERENCES Product(ProductID)  
);
```

```
-- Customer Visit Table  
CREATE TABLE CustomerVisit (  
    VisitID INT PRIMARY KEY,  
    RepresentativeID INT NOT NULL,  
    CustomerID INT NOT NULL,  
    VisitDate DATE NOT NULL,  
    Notes VARCHAR(500),  
    FOREIGN KEY (RepresentativeID) REFERENCES SalesRepresentative(RepresentativeID),  
    FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID)  
);
```

## 7. Populate Data

```
-- Insert sample data into SalesManager
INSERT INTO SalesManager VALUES (1, 'Robert Johnson', 'Northeast');
INSERT INTO SalesManager VALUES (2, 'Sarah Williams', 'Midwest');
INSERT INTO SalesManager VALUES (3, 'Michael Chen', 'West Coast');
INSERT INTO SalesManager VALUES (4, 'Laura Simmons', 'Southeast');
INSERT INTO SalesManager VALUES (5, 'Carlos Mendoza', 'Southwest');
INSERT INTO SalesManager VALUES (6, 'Anita Kapoor', 'Northwest');
INSERT INTO SalesManager VALUES (7, 'Thomas Becker', 'Central Europe');
INSERT INTO SalesManager VALUES (8, 'Yuki Tanaka', 'Asia-Pacific');
INSERT INTO SalesManager VALUES (9, 'Fatima Al-Fayed', 'Middle East & Africa');
INSERT INTO SalesManager VALUES (10, 'Lucas Martins', 'South America');

-- Insert sample data into SalesRepresentative
INSERT INTO SalesRepresentative VALUES (101, 1, 'Emily Davis');
INSERT INTO SalesRepresentative VALUES (102, 1, 'David Wilson');
INSERT INTO SalesRepresentative VALUES (103, 2, 'Jennifer Brown');
INSERT INTO SalesRepresentative VALUES (104, 3, 'James Miller');
INSERT INTO SalesRepresentative VALUES (105, 4, 'Nathan Lee');
INSERT INTO SalesRepresentative VALUES (106, 4, 'Alicia Gomez');
INSERT INTO SalesRepresentative VALUES (107, 5, 'Raj Patel');
INSERT INTO SalesRepresentative VALUES (108, 5, 'Sophia Martinez');
INSERT INTO SalesRepresentative VALUES (109, 6, 'Ethan Wright');
INSERT INTO SalesRepresentative VALUES (110, 6, 'Mia Johansson');
INSERT INTO SalesRepresentative VALUES (111, 7, 'Felix Schneider');
INSERT INTO SalesRepresentative VALUES (112, 7, 'Hanna Müller');
INSERT INTO SalesRepresentative VALUES (113, 8, 'Kenji Sato');
INSERT INTO SalesRepresentative VALUES (114, 8, 'Mei Lin');
INSERT INTO SalesRepresentative VALUES (115, 9, 'Omar Hassan');
INSERT INTO SalesRepresentative VALUES (116, 9, 'Layla Mansour');
INSERT INTO SalesRepresentative VALUES (117, 10, 'Bruno Silva');
INSERT INTO SalesRepresentative VALUES (118, 10, 'Camila Torres');

-- Insert sample data into Customer
INSERT INTO Customer VALUES (c, 'Tech Solutions Inc.', '123 Business Ave, Boston, MA', 6175551001);
INSERT INTO Customer VALUES (1002, 'Global Enterprises', '456 Corporate Blvd, Chicago, IL', 3125552002);
INSERT INTO Customer VALUES (1003, 'Innovative Systems', '789 Tech Park, San Francisco, CA', 4155553003);
INSERT INTO Customer VALUES (1004, 'Data Analytics Co.', '321 Digital Lane, New York, NY', 2125554004);
INSERT INTO Customer VALUES (1005, 'NextGen Robotics Ltd.', '987 Innovation Drive, Austin, TX', 5125555005);
INSERT INTO Customer VALUES (1006, 'Quantum Tech Labs', '654 Quantum Blvd, Seattle, WA', 2065556006);
INSERT INTO Customer VALUES (1007, 'Bright Future Education', '321 Learning Way, Denver, CO', 3035557007);
INSERT INTO Customer VALUES (1008, 'Green Energy Corp.', '159 Eco Street, Portland, OR', 5035558008);
INSERT INTO Customer VALUES (1009, 'Skyline Architecture Group', '222 Skyline Ave, Atlanta, GA', 4045559009);
INSERT INTO Customer VALUES (1010, 'Fusion Medical Systems', '110 Health Park, Raleigh, NC', 9195551010);
INSERT INTO Customer VALUES (1011, 'Visionary Marketing LLC', '443 Campaign Blvd, Miami, FL', 3055551111);
INSERT INTO Customer VALUES (1012, 'SecureNet Solutions', '876 Firewall Lane, Dallas, TX', 2145551212);
```

```
INSERT INTO Customer VALUES (1013, 'Urban Mobility Inc.', '132 Transit Road, Chicago, IL', 3125551313);
INSERT INTO Customer VALUES (1014, 'NeoSpace Innovations', '789 Orbit Drive, Los Angeles, CA', 2135551414);
```

-- Insert sample data into ProductCategory

```
INSERT INTO ProductCategory VALUES (10, 'Software', 'Business applications and tools');
INSERT INTO ProductCategory VALUES (20, 'Hardware', 'Computer equipment and devices');
INSERT INTO ProductCategory VALUES (30, 'Services', 'Consulting and support services');
INSERT INTO ProductCategory VALUES (40, 'Networking', 'Routers, switches, and network hardware');
INSERT INTO ProductCategory VALUES (50, 'Cloud Services', 'Hosted solutions and storage');
INSERT INTO ProductCategory VALUES (60, 'Mobile Devices', 'Smartphones, tablets, and accessories');
INSERT INTO ProductCategory VALUES (70, 'Security', 'Cybersecurity software and tools');
INSERT INTO ProductCategory VALUES (80, 'AI Solutions', 'Artificial Intelligence and machine learning tools');
INSERT INTO ProductCategory VALUES (90, 'IoT Devices', 'Internet of Things hardware and controllers');
INSERT INTO ProductCategory VALUES (100, 'Training & Certification', 'Employee training programs and certification services');
INSERT INTO ProductCategory VALUES (110, 'Energy Solutions', 'Smart energy management and power backup systems');
INSERT INTO ProductCategory VALUES (120, 'E-commerce Tools', 'Platforms and plugins for online stores');
INSERT INTO ProductCategory VALUES (130, 'Collaboration Tools', 'Project management and communication platforms');
```

-- Insert sample data into Product

```
INSERT INTO Product VALUES (5001, 10, 'Enterprise CRM Suite', 4999.99);
INSERT INTO Product VALUES (5002, 10, 'Data Analytics Pro', 3499.99);
INSERT INTO Product VALUES (5003, 20, 'Business Server X9', 8999.99);
INSERT INTO Product VALUES (5004, 20, 'Workstation Pro', 2499.99);
INSERT INTO Product VALUES (5005, 30, 'Premium Support Package', 1999.99);
INSERT INTO Product VALUES (5006, 40, 'Enterprise Router 500', 1299.99);
INSERT INTO Product VALUES (5007, 40, 'Smart Switch 24-Port', 899.99);
INSERT INTO Product VALUES (5008, 50, 'Cloud Storage 1TB Plan', 999.99);
INSERT INTO Product VALUES (5009, 50, 'Hosted Email Service', 799.99);
INSERT INTO Product VALUES (5010, 60, 'ProTab 12X Tablet', 749.99);
INSERT INTO Product VALUES (5011, 60, 'Smartphone XZ Max', 999.99);
INSERT INTO Product VALUES (5012, 70, 'CyberShield Antivirus Suite', 299.99);
INSERT INTO Product VALUES (5013, 70, 'SecureVPN Enterprise', 199.99);
INSERT INTO Product VALUES (5014, 80, 'AI Forecasting Engine', 5999.99);
INSERT INTO Product VALUES (5015, 80, 'Vision Recognition SDK', 4299.99);
INSERT INTO Product VALUES (5016, 90, 'Smart Sensor Hub', 899.99);
INSERT INTO Product VALUES (5017, 90, 'IoT Gateway Pro', 1199.99);
INSERT INTO Product VALUES (5018, 100, 'Employee Skills Training Pack', 499.99);
INSERT INTO Product VALUES (5019, 100, 'Cloud Certification Course', 799.99);
INSERT INTO Product VALUES (5020, 110, 'Smart UPS 2000VA', 1499.99);
INSERT INTO Product VALUES (5021, 110, 'EnergyMonitor Pro', 1299.99);
INSERT INTO Product VALUES (5022, 120, 'E-Store Builder', 999.99);
INSERT INTO Product VALUES (5023, 120, 'Payment Gateway Plugin', 299.99);
INSERT INTO Product VALUES (5024, 130, 'TeamCollab Suite', 699.99);
INSERT INTO Product VALUES (5025, 130, 'Agile Project Manager Pro', 899.99);
INSERT INTO Product VALUES (5026, 40, 'NextGen Firewall Appliance', 1599.99);
```

```
INSERT INTO Product VALUES (5027, 40, 'CloudEdge Router X1', 1099.99);
INSERT INTO Product VALUES (5028, 50, 'Virtual Machine Hosting - Pro Plan', 1299.99);
INSERT INTO Product VALUES (5029, 50, 'Cloud Backup Unlimited', 899.99);
INSERT INTO Product VALUES (5030, 60, 'Business Tablet M10', 699.99);
INSERT INTO Product VALUES (5031, 60, 'Executive Smartphone Z Ultra', 1099.99);
INSERT INTO Product VALUES (5032, 70, 'Email Protection Suite', 399.99);
INSERT INTO Product VALUES (5033, 70, 'Enterprise Firewall Software', 599.99);
INSERT INTO Product VALUES (5034, 80, 'Smart Recommendation Engine', 4999.99);
INSERT INTO Product VALUES (5035, 80, 'Predictive Maintenance Module', 4599.99);
INSERT INTO Product VALUES (5036, 90, 'IoT Motion Detector Kit', 649.99);
INSERT INTO Product VALUES (5037, 90, 'Industrial IoT Controller', 1499.99);
INSERT INTO Product VALUES (5038, 100, 'Leadership Development Course', 599.99);
INSERT INTO Product VALUES (5039, 100, 'Tech Onboarding Program', 399.99);
INSERT INTO Product VALUES (5040, 110, 'Solar Battery Storage Unit', 2999.99);
INSERT INTO Product VALUES (5041, 110, 'Smart Energy Controller', 999.99);
INSERT INTO Product VALUES (5042, 120, 'SEO Optimization Plugin', 199.99);
INSERT INTO Product VALUES (5043, 120, 'Abandoned Cart Recovery Tool', 249.99);
INSERT INTO Product VALUES (5044, 130, 'Remote Meeting Suite', 499.99);
INSERT INTO Product VALUES (5045, 130, 'Workflow Automation Tool', 799.99);
```

-- Insert sample data into SalesOrder

```
INSERT INTO SalesOrder VALUES (9001, 101, 1001, '2023-06-15');
INSERT INTO SalesOrder VALUES (9002, 102, 1002, '2023-06-18');
INSERT INTO SalesOrder VALUES (9003, 103, 1003, '2023-06-20');
INSERT INTO SalesOrder VALUES (9004, 104, 1004, '2023-06-22');
INSERT INTO SalesOrder VALUES (9005, 101, 1002, '2023-06-25');
INSERT INTO SalesOrder VALUES (9006, 105, 1005, '2023-07-03');
INSERT INTO SalesOrder VALUES (9007, 106, 1006, '2023-07-05');
INSERT INTO SalesOrder VALUES (9008, 107, 1007, '2023-07-07');
INSERT INTO SalesOrder VALUES (9009, 108, 1008, '2023-07-09');
INSERT INTO SalesOrder VALUES (9010, 109, 1009, '2023-07-11');
INSERT INTO SalesOrder VALUES (9011, 110, 1010, '2023-07-13');
INSERT INTO SalesOrder VALUES (9012, 111, 1011, '2023-07-15');
INSERT INTO SalesOrder VALUES (9013, 112, 1012, '2023-07-17');
INSERT INTO SalesOrder VALUES (9014, 113, 1013, '2023-07-19');
INSERT INTO SalesOrder VALUES (9015, 114, 1014, '2023-07-21');
INSERT INTO SalesOrder VALUES (9016, 115, 1001, '2023-07-23');
INSERT INTO SalesOrder VALUES (9017, 116, 1002, '2023-07-25');
INSERT INTO SalesOrder VALUES (9018, 117, 1003, '2023-07-27');
INSERT INTO SalesOrder VALUES (9019, 118, 1004, '2023-07-29');
```

-- Insert sample data into OrderItem

```
INSERT INTO OrderItem VALUES (8001, 9001, 5001, 1, 4999.99);
INSERT INTO OrderItem VALUES (8002, 9001, 5005, 1, 1999.99);
INSERT INTO OrderItem VALUES (8003, 9002, 5003, 2, 8999.99);
INSERT INTO OrderItem VALUES (8004, 9003, 5002, 1, 3499.99);
INSERT INTO OrderItem VALUES (8005, 9004, 5004, 5, 2499.99);
```

```
INSERT INTO OrderItem VALUES (8006, 9005, 5001, 1, 4999.99);
INSERT INTO OrderItem VALUES (8007, 9005, 5002, 1, 3499.99);
INSERT INTO OrderItem VALUES (8008, 9006, 5026, 1, 1599.99);
INSERT INTO OrderItem VALUES (8009, 9006, 5030, 2, 699.99);
INSERT INTO OrderItem VALUES (8010, 9007, 5028, 1, 1299.99);
INSERT INTO OrderItem VALUES (8011, 9008, 5032, 3, 399.99);
INSERT INTO OrderItem VALUES (8012, 9009, 5040, 1, 2999.99);
INSERT INTO OrderItem VALUES (8013, 9010, 5037, 4, 1499.99);
INSERT INTO OrderItem VALUES (8014, 9011, 5022, 1, 999.99);
INSERT INTO OrderItem VALUES (8015, 9012, 5014, 1, 5999.99);
INSERT INTO OrderItem VALUES (8016, 9013, 5044, 5, 499.99);
INSERT INTO OrderItem VALUES (8017, 9014, 5010, 2, 749.99);
INSERT INTO OrderItem VALUES (8018, 9015, 5034, 1, 4999.99);
INSERT INTO OrderItem VALUES (8019, 9016, 5007, 3, 899.99);
INSERT INTO OrderItem VALUES (8020, 9017, 5020, 1, 1499.99);
INSERT INTO OrderItem VALUES (8021, 9018, 5012, 2, 299.99);
INSERT INTO OrderItem VALUES (8022, 9019, 5023, 1, 249.99);
```

-- Insert sample data into CustomerVisit

```
INSERT INTO CustomerVisit VALUES (7001, 101, 1001, '2023-06-10', 'Initial product demonstration');
INSERT INTO CustomerVisit VALUES (7002, 102, 1002, '2023-06-12', 'Contract renewal discussion');
INSERT INTO CustomerVisit VALUES (7003, 103, 1003, '2023-06-14', 'New feature presentation');
INSERT INTO CustomerVisit VALUES (7004, 104, 1004, '2023-06-19', 'Installation follow-up');
INSERT INTO CustomerVisit VALUES (7005, 101, 1002, '2023-06-24', 'Upsell opportunity identified');
INSERT INTO CustomerVisit VALUES (7006, 101, 1005, '2023-07-01', 'Product demo and feasibility discussion');
INSERT INTO CustomerVisit VALUES (7007, 102, 1006, '2023-07-02', 'Security needs assessment');
INSERT INTO CustomerVisit VALUES (7008, 103, 1007, '2023-07-04', 'Training program proposal');
INSERT INTO CustomerVisit VALUES (7009, 104, 1008, '2023-07-05', 'Energy solution introduction');
INSERT INTO CustomerVisit VALUES (7010, 101, 1009, '2023-07-06', 'System upgrade consultation');
INSERT INTO CustomerVisit VALUES (7011, 102, 1010, '2023-07-07', 'Healthcare compliance briefing');
INSERT INTO CustomerVisit VALUES (7012, 103, 1011, '2023-07-08', 'Marketing platform integration');
INSERT INTO CustomerVisit VALUES (7013, 104, 1012, '2023-07-09', 'Firewall and endpoint security demo');
INSERT INTO CustomerVisit VALUES (7014, 101, 1013, '2023-07-10', 'Smart mobility and IoT applications');
INSERT INTO CustomerVisit VALUES (7015, 102, 1014, '2023-07-11', 'AI-based analytics workshop');
```

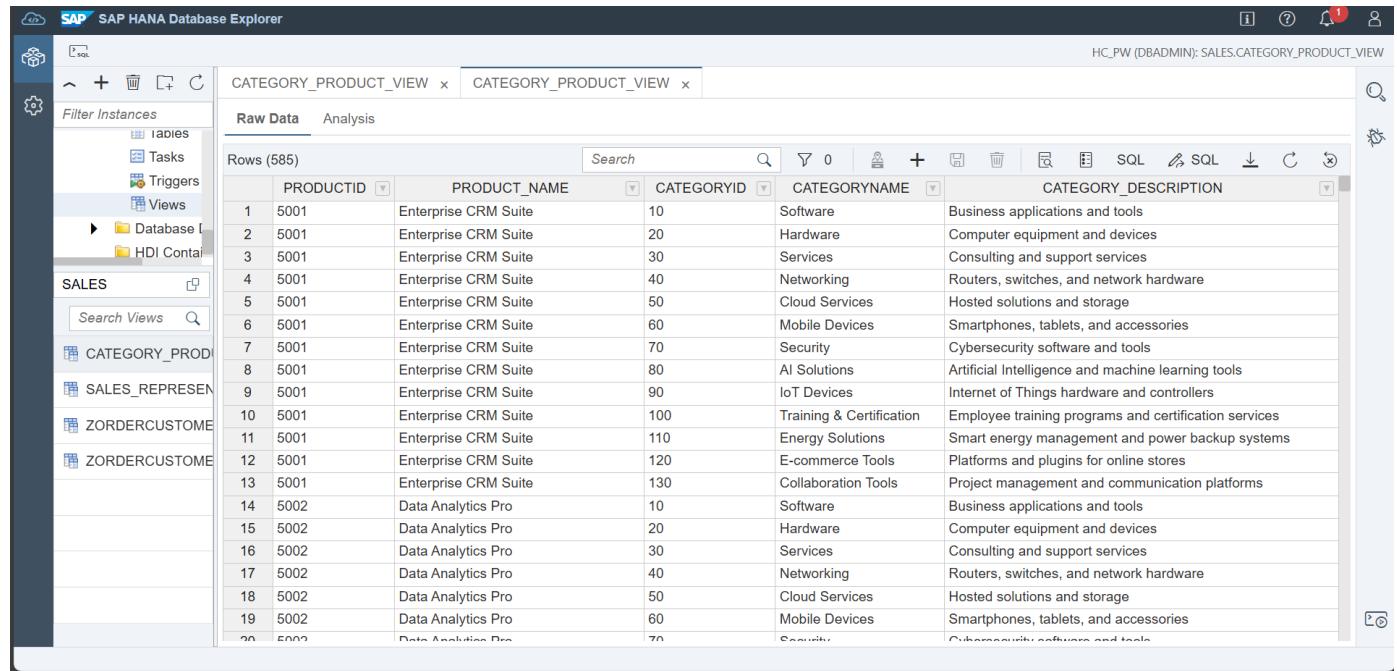
## 8. Alter Table

```
ALTER TABLE Product ADD (Price DECIMAL(10,2) DEFAULT 0 NOT NULL);
```

## 9. Create View

### a. CATEGORY\_PRODUCT\_VIEW

```
CREATE OR REPLACE VIEW CATEGORY_PRODUCT_VIEW AS
SELECT
    P.ProductID,
    P.Name AS Product_Name,
    C.CategoryID,
    C.CategoryName,
    C.Description AS Category_Description
FROM Product P
LEFT JOIN SALES.ProductCategory C ON P.CategoryID = C.CategoryID
```



The screenshot shows the SAP HANA Database Explorer interface. On the left, there's a sidebar with a tree view of database objects under 'Filter Instances' and 'SALES'. The main area displays a table titled 'CATEGORY\_PRODUCT\_VIEW' with 585 rows. The columns are: PRODUCTID, PRODUCT\_NAME, CATEGORYID, CATEGORYNAME, and CATEGORY\_DESCRIPTION. The data includes various product categories like Enterprise CRM Suite, Data Analytics Pro, Software, Hardware, Services, Networking, Cloud Services, Mobile Devices, Security, AI Solutions, IoT Devices, Training & Certification, Energy Solutions, E-commerce Tools, Collaboration Tools, and more.

PRODUCTID	PRODUCT_NAME	CATEGORYID	CATEGORYNAME	CATEGORY_DESCRIPTION
1	Enterprise CRM Suite	10	Software	Business applications and tools
2	Enterprise CRM Suite	20	Hardware	Computer equipment and devices
3	Enterprise CRM Suite	30	Services	Consulting and support services
4	Enterprise CRM Suite	40	Networking	Routers, switches, and network hardware
5	Enterprise CRM Suite	50	Cloud Services	Hosted solutions and storage
6	Enterprise CRM Suite	60	Mobile Devices	Smartphones, tablets, and accessories
7	Enterprise CRM Suite	70	Security	Cybersecurity software and tools
8	Enterprise CRM Suite	80	AI Solutions	Artificial Intelligence and machine learning tools
9	Enterprise CRM Suite	90	IoT Devices	Internet of Things hardware and controllers
10	Enterprise CRM Suite	100	Training & Certification	Employee training programs and certification services
11	Enterprise CRM Suite	110	Energy Solutions	Smart energy management and power backup systems
12	Enterprise CRM Suite	120	E-commerce Tools	Platforms and plugins for online stores
13	Enterprise CRM Suite	130	Collaboration Tools	Project management and communication platforms
14	Data Analytics Pro	10	Software	Business applications and tools
15	Data Analytics Pro	20	Hardware	Computer equipment and devices
16	Data Analytics Pro	30	Services	Consulting and support services
17	Data Analytics Pro	40	Networking	Routers, switches, and network hardware
18	Data Analytics Pro	50	Cloud Services	Hosted solutions and storage
19	Data Analytics Pro	60	Mobile Devices	Smartphones, tablets, and accessories
20	Data Analytics Pro	70	Security	Cybersecurity software and tools

## b. SALES\_REPRESENTATIVE\_VIEW

```
CREATE OR REPLACE VIEW SALES_REPRESENTATIVE_VIEW AS
```

```
SELECT  
    SR.REPRESENTATIVEID,  
    SR.Name,  
    COUNT(SO.OrderID) AS Total_Orders  
FROM  
    SalesRepresentative SR  
LEFT JOIN  
    SalesOrder SO ON SO.RepresentativeID = SR.RepresentativeID  
    GROUP BY  
        SR.RepresentativeID, SR.Name
```

The screenshot shows the SAP HANA Database Explorer interface. On the left, there is a sidebar with a tree view of database objects under the 'SALES' schema, including tables, triggers, and views. The 'Views' node is currently selected. In the main area, two tabs are open: 'SALES\_REPRESENTATIVE\_V...' and 'SALES\_REPRESENTATIVE\_V...'. The 'Raw Data' tab is active, displaying 18 rows of data from the view. The columns shown are 'REPRESENTATIVEID', 'NAME', and 'TOTAL\_ORDERS'. The data is as follows:

	REPRESENTATIVEID	NAME	TOTAL_ORDERS
1	101	Emily Davis	2
2	102	David Wilson	1
3	103	Jennifer Brown	1
4	104	James Miller	1
5	105	Nathan Lee	1
6	106	Alicia Gomez	1
7	107	Raj Patel	1
8	108	Sophia Martinez	1
9	109	Ethan Wright	1
10	110	Mia Johansson	1
11	111	Felix Schneider	1
12	112	Hanna Müller	1
13	113	Kenji Sato	1
14	114	Mei Lin	1
15	115	Omar Hassan	1
16	116	Layla Mansour	1
17	117	Bruno Silva	1
18	118	Camila Torres	1

## 10.Create Function

### a. AVERAGE\_SALES\_PER\_CUSTOMER

```
CREATE OR REPLACE FUNCTION AVERAGE_SALES_PER_CUSTOMER(IN CustomerID INT)
RETURNS avg_sales NUMERIC(10, 2)
AS
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION avg_sales := '-1';
    SELECT TO_DECIMAL(ROUND(SUM(OI.Price * OI.Quantity)/COUNT(DISTINCT SO.CustomerID), 2,
    ROUND_HALF_UP))
    INTO avg_sales
    FROM SalesOrder SO
    JOIN OrderItem OI ON SO.ORDERID = OI.ORDERID;
END;
```

For Input Parameter: 1001

The screenshot shows the SAP HANA Database Explorer interface. On the left, the object browser displays a tree structure under the 'SALES' schema, with nodes for 'Cubes', 'Function', and two entries labeled 'AVERAGE\_SALES\_'. In the center, a SQL console window titled 'SQL Console 1.sql' contains the following code:

```
1 SELECT "SALES"."AVERAGE_SALES_PER_CUSTOMER"(1001).AVG_SALES AS AVG_SALES
2 FROM DUMMY
```

The 'Result' tab is selected, showing a single row of data:

	AVG_SALES
1	9699.95

b. AVERAGE\_SALES\_PER\_SALES\_REPRESENTATIVE

```
CREATE OR REPLACE FUNCTION AVERAGE_SALES_PER_SALES_REPRESENTATIVE(IN  
REPRESENTATIVEID INT)  
RETURNS avg_sales NUMERIC(10, 2)  
AS  
BEGIN  
    DECLARE EXIT HANDLER FOR SQLEXCEPTION avg_sales := '-1';  
    SELECT TO_DECIMAL(ROUND(SUM(OI.Price * OI.Quantity)/COUNT(DISTINCT SO.RepresentativeID), 2,  
ROUND_HALF_UP))  
    INTO avg_sales  
    FROM OrderItem OI  
    JOIN SalesOrder SO ON OI.ORDERID = SO.ORDERID  
    WHERE SO.REPRESENTATIVEID = :REPRESENTATIVEID;  
END;
```

For Input Parameter: 101

The screenshot shows the SAP HANA Database Explorer interface. On the left, there's a sidebar with a tree view of database objects under the 'SALES' schema, including two instances of the 'AVERAGE\_SALES' function. The main area has tabs for 'SQL Console 2.sql' and 'Result'. The SQL tab contains the following code:

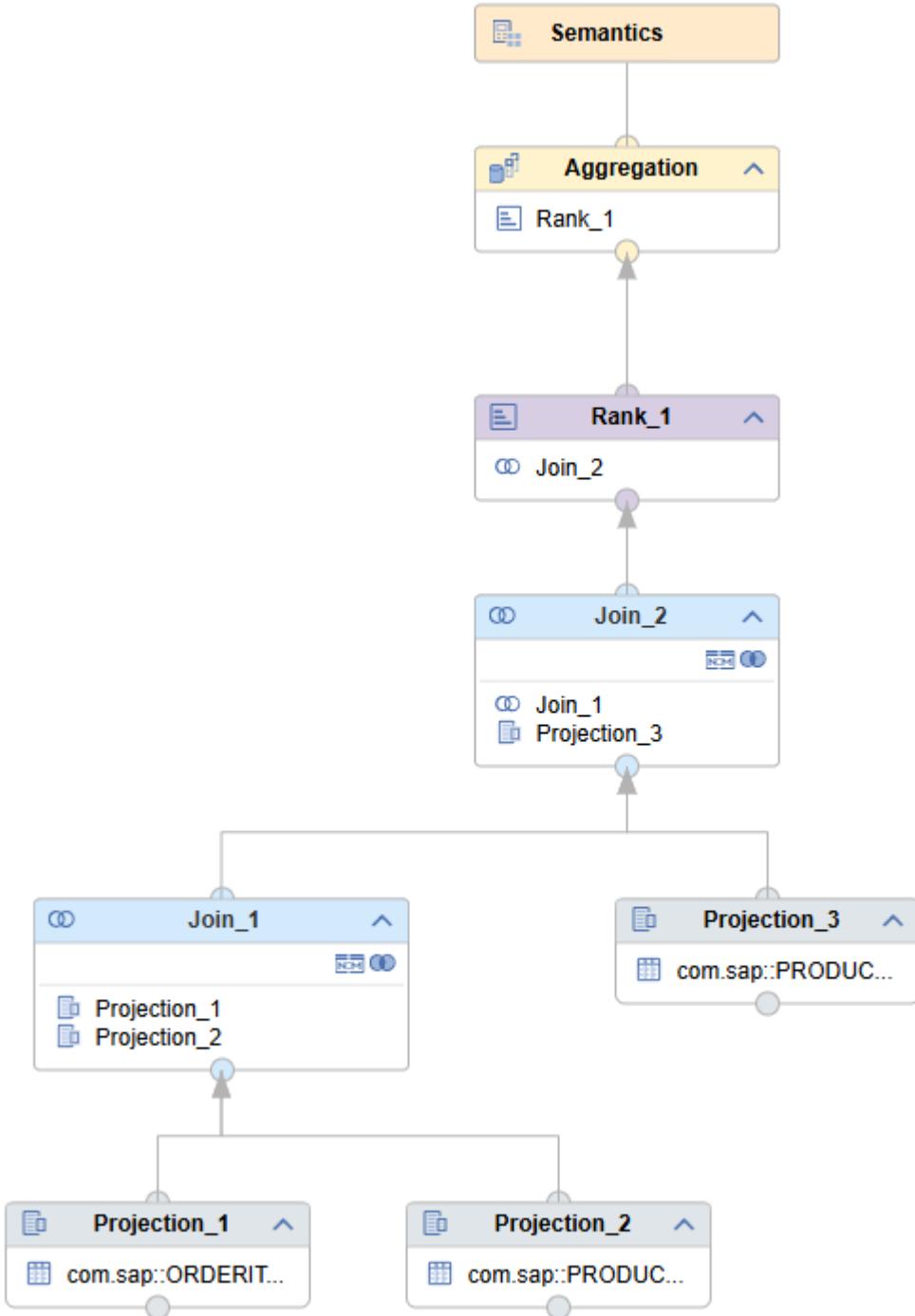
```
1 v SELECT "SALES"."AVERAGE_SALES_PER_SALES_REPRESENTATIVE"(101) .AVG_SALES AS AVG_SALES  
2 FROM DUMMY
```

The Result tab shows the output of the query:

	AVG_SALES
1	15499.96

## 11. Create Calculation View

### a. CVIEW\_TOP\_5\_CATEGORIES SOLD



SAP HANA Database Explorer

HBI\_Tutorial (dev): HDI\_TUTORIAL.CVIEW\_TOP\_5\_CATEGORIES\_SOLD

**Filter Instances**

- HBI\_Tutorial (dev)
  - Column Views
  - Cubes
  - Functions
  - Graph Workspaces
  - Indexes
  - JSON Collections
  - Libraries
  - Procedures
  - Public Synonyms
  - Remote Subscriptions

**Analysis** **Raw Data**

Rows (5)

	CATEGORYID	CATEGORYNAME	DESCRIPTION	QUANTITY
1	20	Hardware	Computer equipment and devices	5
2	130	Collaboration Tools	Project management and communication platforms	5
3	90	IoT Devices	Internet of Things hardware and controllers	4
4	40	Networking	Routers, switches, and network hardware	3
5	70	Security	Cybersecurity software and tools	3

Search Column Views

CalculationView

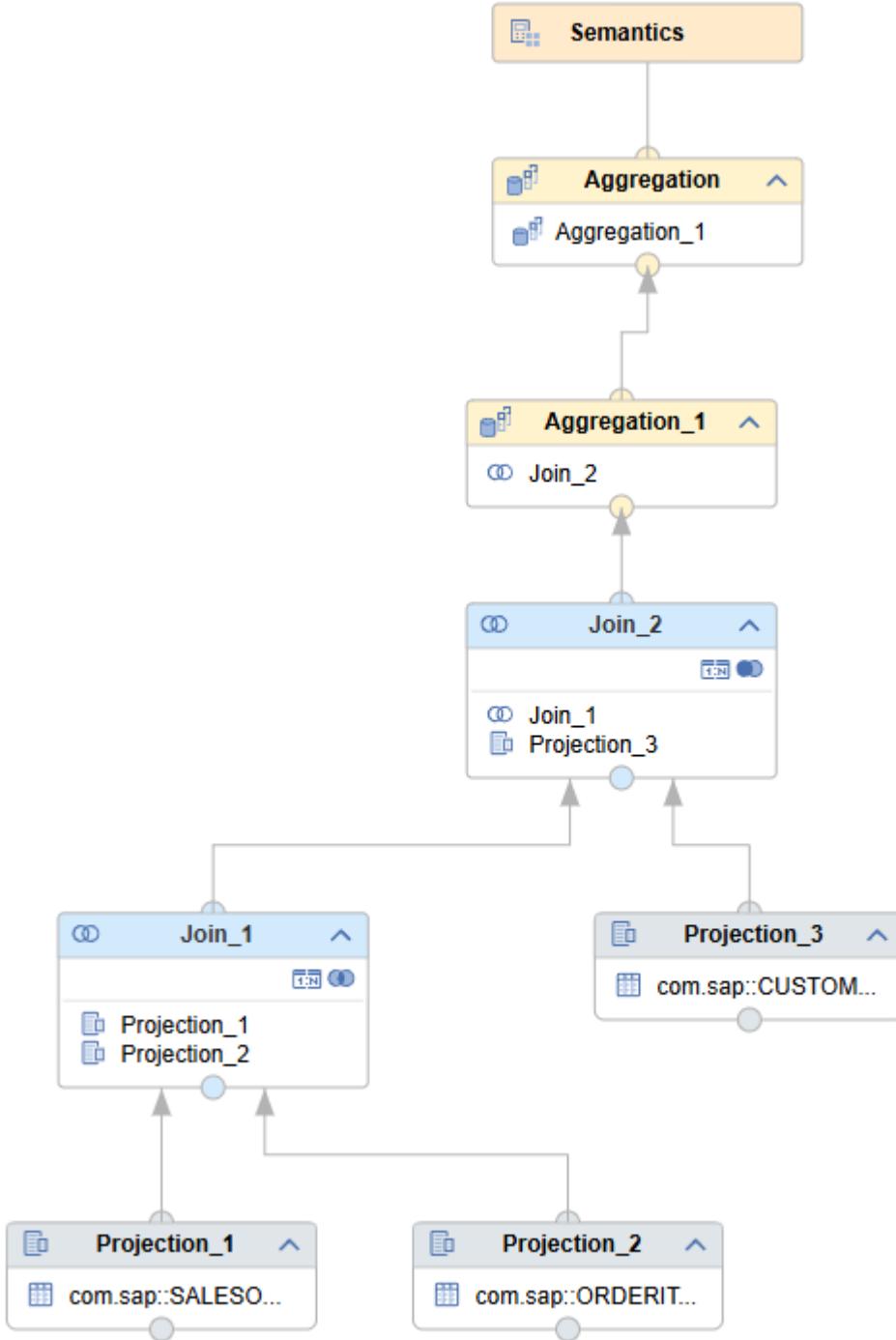
CVIEW\_AVG\_SALES\_ORDER\_PRICE\_F

CVIEW\_TOP\_3\_SALES\_REPRENSTATI

CVIEW\_TOP\_5\_CATEGORIES SOLD

The screenshot shows the SAP HANA Database Explorer interface. On the left, there's a sidebar with a tree view of database instances, currently expanded to show 'HBI\_Tutorial (dev)' which contains various objects like Column Views, Cubes, Functions, etc. Below this is a search bar for 'Search Column Views'. The main area is titled 'Raw Data' and displays a table with 5 rows of data. The columns are labeled 'CATEGORYID', 'CATEGORYNAME', 'DESCRIPTION', and 'QUANTITY'. The data shows five categories: Hardware (Computer equipment and devices, quantity 5), Collaboration Tools (Project management and communication platforms, quantity 5), IoT Devices (Internet of Things hardware and controllers, quantity 4), Networking (Routers, switches, and network hardware, quantity 3), and Security (Cybersecurity software and tools, quantity 3). The 'QUANTITY' column is highlighted with a blue selection bar.

## b. CVIEW\_AVG\_SALES\_ORDER\_PRICE\_PER\_CUSTOMER



SAP HANA Database Explorer

HBI\_Tutorial (dev): HDI\_TUTORIAL.CVIEW\_AVG\_SALES\_ORDER\_PRICE\_PER\_CUSTOMER

Filter Instances

HBI\_Tutorial (dev)

- Cubes
- Functions
- Graph Workspaces
- Indexes
- JSON Collections
- Libraries
- Procedures
- Public Synonyms
- Remote Subscriptions

Search Column Views

CalculationView

CVIEW\_AVG\_SALES\_ORDER\_PRICE\_PER\_CUSTOMER

CVIEW\_TOP\_3\_SALES\_REPRESENTATIVES

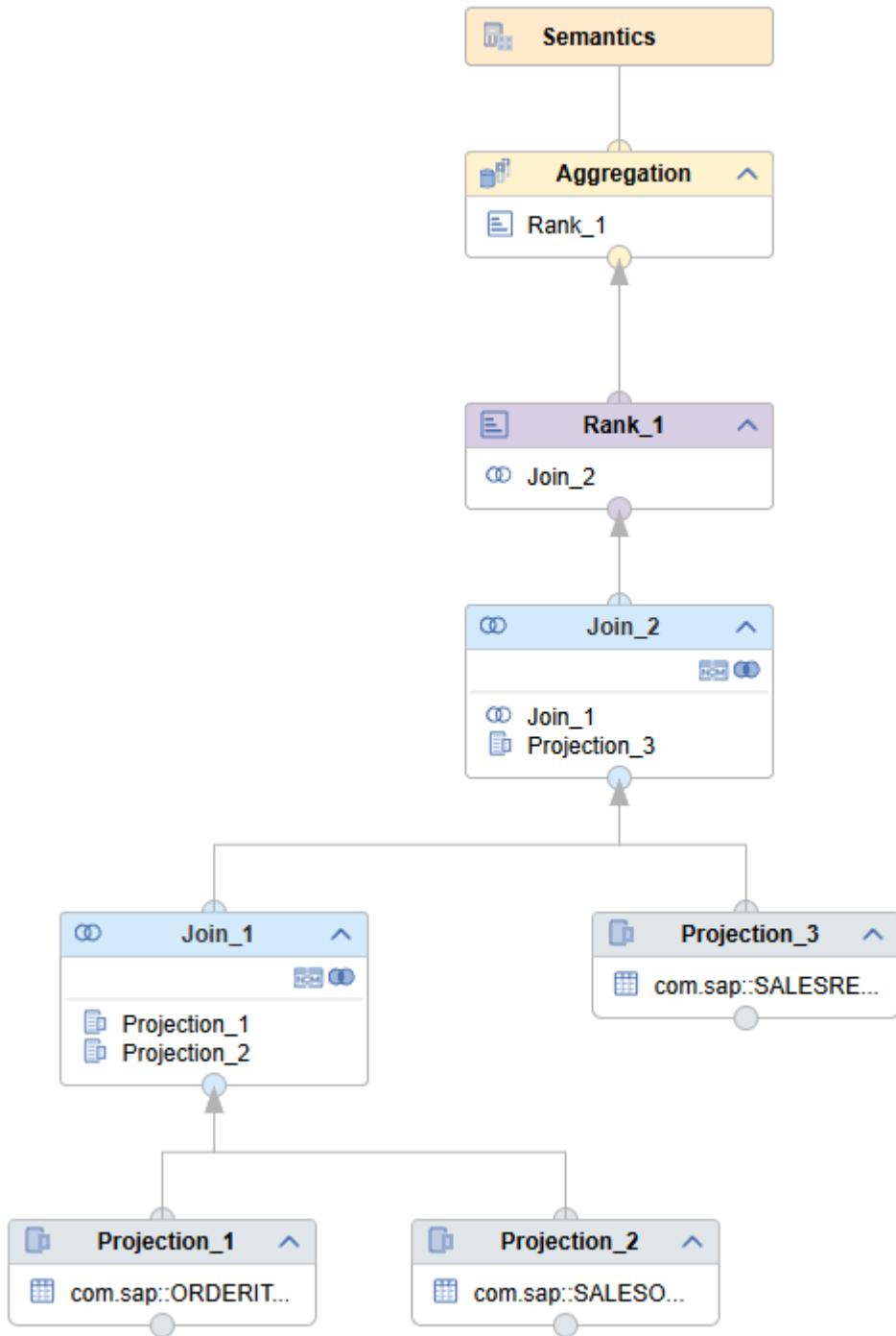
CVIEW\_TOP\_5\_CATEGORIES SOLD

Analysis Raw Data

Rows (14)

	CUSTOMERID	NAME	Number_of_Orders	AvgPricePerCustomer	Order_Total
1	1001	Tech Solutions Inc.	3	3233.31	9699.95
2	1002	Global Enterprises	4	6999.98	27999.95
3	1003	Innovative Systems	2	2049.98	4099.97
4	1004	Data Analytics Co.	2	6374.97	12749.94
5	1005	NextGen Robotics Ltd.	2	1499.98	2999.97
6	1006	Quantum Tech Labs	1	1299.99	1299.99
7	1007	Bright Future Education	1	1199.97	1199.97
8	1008	Green Energy Corp.	1	2999.99	2999.99
9	1009	Skyline Architecture Group	1	5999.96	5999.96
10	1010	Fusion Medical Systems	1	999.99	999.99
11	1011	Visionary Marketing LLC	1	5999.99	5999.99
12	1012	SecureNet Solutions	1	2499.95	2499.95
13	1013	Urban Mobility Inc.	1	1499.98	1499.98
14	1014	NeoSpace Innovations	1	4999.99	4999.99

c. CVIEW\_TOP\_3\_SALES\_REPRENSTATIVES



SAP HANA Database Explorer

HBI\_Tutorial (dev): HDI\_TUTORIAL.CVIEW\_TOP\_3\_SALES\_REPRENSTATIVES

Filter Instances

Analysis Raw Data

Rows (3)

	NAME	REPRESENTATIVEID	TotalSales
1	David Wilson	102	17999
2	James Miller	104	12499
3	Felix Schneider	111	5999

Search Column Views

CalculationView

CVIEW\_AVG\_SALES\_ORDER\_PRICE\_F

CVIEW\_TOP\_3\_SALES\_REPRENSTATIV

CVIEW\_TOP\_5\_CATEGORIES SOLD

Search Column Views

Filter Instances

Analysis Raw Data

Rows (3)

	NAME	REPRESENTATIVEID	TotalSales
1	David Wilson	102	17999
2	James Miller	104	12499
3	Felix Schneider	111	5999