Question 1: Create the DataFrames

You will create two DataFrames: one for products and another for sales transactions.

Program:  
from pyspark.sql import SparkSession  
from pyspark.sql.functions import col  
  
# Initialize SparkSession  
spark = SparkSession.builder.appName("Product Sales Analysis").getOrCreate()  
  
# Sample data for products  
products = [  
    (1, "Laptop", "Electronics", 50000),  
    (2, "Smartphone", "Electronics", 30000),  
    (3, "Table", "Furniture", 15000),  
    (4, "Chair", "Furniture", 5000),  
    (5, "Headphones", "Electronics", 2000),  
]  
  
# Sample data for sales transactions  
sales = [  
    (1, 1, 2),  
    (2, 2, 1),  
    (3, 3, 3),  
    (4, 1, 1),  
    (5, 4, 5),  
    (6, 2, 2),  
    (7, 5, 10),  
    (8, 3, 1),  
]  
  
# Define schema for DataFrames  
product\_columns = ["ProductID", "ProductName", "Category", "Price"]  
sales\_columns = ["SaleID", "ProductID", "Quantity"]  
  
# Create DataFrames  
product\_df = spark.createDataFrame(products, schema=product\_columns)  
sales\_df = spark.createDataFrame(sales, schema=sales\_columns)

Question 2: Join the DataFrames

Join the `product\_df` and `sales\_df` DataFrames on `ProductID` to create a combined DataFrame with product and sales data.

Program:  
combined\_df = product\_df.join(sales\_df, on="ProductID")

Question 3: Calculate Total Sales Value

For each product, calculate the total sales value by multiplying the price by the quantity sold.

Program:  
combined\_df = combined\_df.withColumn("TotalSalesValue", col("Price") \* col("Quantity"))

Question 4: Find the Total Sales for Each Product Category

Group the data by the `Category` column and calculate the total sales value for each product category.

Program:  
category\_sales\_df = combined\_df.groupBy("Category").agg({"TotalSalesValue": "sum"})

Question 5: Identify the Top-Selling Product

Find the product that generated the highest total sales value.

Program:  
top\_selling\_product = product\_sales\_df.orderBy(col("sum(TotalSalesValue)").desc()).limit(1)

Question 6: Sort the Products by Total Sales Value

Sort the products by total sales value in descending order.

Program:  
sorted\_product\_sales\_df = product\_sales\_df.orderBy(col("sum(TotalSalesValue)").desc())

Question 7: Count the Number of Sales for Each Product

Count the number of sales transactions for each product.

Program:  
sales\_count\_df = combined\_df.groupBy("ProductName").count()

Question 8: Filter Products with Total Sales Value Greater Than ₹50,000

Filter out the products that have a total sales value greater than ₹50,000.

Program:  
high\_sales\_products\_df = sorted\_product\_sales\_df.filter(col("sum(TotalSalesValue)") > 50000)