(Affiliated to University of Mumbai)	***End Semester Examination (R-24/R-25) SH 2026 Answer Key with marking scheme***
Branch: ECS/CE/EXTC	Course: Data Analytics and Visualization
Year/ Semester: TE / V	Course code: MDMC5022
Time: 03 hours	Marks: 80

		Marks
Q. 1	All questions are compulsory.	30
A.	Business understanding identifies the decision/problem, stakeholders, goals, constraints and success criteria (2). Hypothesis creation converts the problem into testable claims, e.g., "long delivery times reduce repeat orders" (2). Hypotheses guide data collection, variables and analysis (1).	05
B.	(a) First difference $\Delta Yt = Yt - Y(t-1)$ (1): [16−12, 21−16, 27−21, 34−27, 42−34] = [4, 5, 6, 7, 8] (1). [2 marks] (b) Second difference $\Delta^2 Yt = \Delta Yt - \Delta Y(t-1)$ (1): [5−4, 6−5, 7−6, 8−7] = [1, 1, 1, 1] (1). [2 marks] (c) Differencing removes trends/changes in mean and can help achieve stationarity required for ARIMA (1). Note: the constant second-difference sequence is degenerate; differencing order should be selected using diagnostics, not assumed automatically. [1 mark]	05
C.	Raster: regular grid of cells/pixels, suitable for continuous surfaces; example satellite imagery/elevation (2). Vector: points, lines and polygons representing discrete objects; example roads or hospital locations (2). Difference: cell resolution versus coordinate geometry, and typical uses (1).	05
D.	(a) Lowercase and remove punctuation: "the products are amazing i purchased two products and they were delivered quickly" (1). (b) With illustrative stop words {the, are, i, two, and, they, were}, remaining tokens: [products, amazing, purchased, products, delivered, quickly]. Stop-word removal (1), correct tokens (1). [2 marks] (c) Stemming/lemmatization maps variants to common forms, reducing sparsity (1); examples: products → product, purchased → purchase, delivered → deliver (1). [2 marks]. Accept reasonable alternative stop-word lists.	05
E.	(a) import pandas as pd; df = pd.read_csv("sales.csv"); print(df.head()). Import/load (0.5), first five records (0.5). [1 mark] (b) print(df["Sales"].isna().sum()); df["Sales"] = df["Sales"].fillna(df["Sales"].median()). Detect missing (0.5), median imputation (0.5). [1 mark] (c) overall_mean = df["Sales"].mean() (1); store_mean = df.groupby("Store_ID")["Sales"].mean() (1). [2 marks] (d) print(df[df["Sales"] < overall_mean]) (0.5); low-value transactions can be investigated by store/category/date for marketing or operational action (0.5). [1 mark]. Accept equivalent pandas syntax; actual numeric output cannot be calculated without the CSV.	05
F.	(a) Add a Slicer visual, drag Region into its field, choose a region (e.g., West); interacting report visuals update for the selected region, subject to configured visual interactions. Correct setup (1), selection/result (1). [2 marks] (b) Select sales bar chart → Filters pane → Filters on this visual → Product_Category → Electronics. Visual-level filter identified (1), correct application and effect limited to selected chart (1). [2 marks] (c) A filter restricts data at visual/page/report scope; a slicer is an on-canvas interactive selection control. One correct distinction (1). [1 mark]	05

Q.2A	Lifecycle diagram (2): Discovery/business understanding → Data preparation → Model planning → Model building → Communicate results → Operationalize (with feedback). Discovery: frame objectives, roles and hypotheses (1). Data preparation: acquire, clean and integrate (1). Model planning: select features and methods (1). Model building: train/validate (1). Communicate: explain findings (1). Operationalize: deploy and monitor (1). ETL reduces errors, missingness and inconsistency (1); variable selection removes irrelevant/redundant inputs, improving generalization and interpretability (1).	10
	OR	

Q.2B.	(A) Diagram (2): Sources → Extract → Transform 1 → Load → Transform 2 → Analytics.
	(b) Stages with E-commerce Examples (4 marks):Extract: retrieve records from databases/files/APIs. First transform: validate types, standardize fields, handle duplicates/missing data. Load: store curated data in warehouse/lake. Second transform: build analytical features, aggregates and model-ready datasets.
	(c) Second Transformation Calculations (2 marks):
	• Order 101 (Laptop): ₹80,000
	• Order 102 (Mobile): ₹45,000
	• Order 103 (Laptop): ₹40,000
	• Product-wise Sales Revenue:
	  ○ Laptop: ₹80,000 + ₹40,000 = ₹1,20,000
	  ○ Mobile: ₹45,000
	• Total Sales Revenue: ₹1,20,000 + ₹45,000 = ₹1,65,000
	(d) Impact on Data Quality & Business Decisions (2 marks):
	• Data Quality: Eliminates duplicates, resolves missing values, and enforces uniform schemas across CSVs, DBs, and APIs, creating a reliable single source of truth.
	• Business Decisions: Enables accurate revenue tracking, identifies top-performing products (e.g., Laptops generating 72.7% of revenue), and supports data-driven inventory, pricing, and marketing strategies.
		10

Q.3A	Box–Jenkins stages: identify stationarity/order using plots, differencing, ACF/PACF (2); estimate parameters (1); diagnose residuals for white noise (1); forecast and validate (1). AR(p): current value depends on previous observations (1). MA(q): current value depends on past shocks/errors (1). ARIMA(p,d,q): combines AR, differencing and MA (1). ACF displays correlations across lags; helps reveal persistence/MA patterns and check residual autocorrelation (2).	10
	OR	
Q.3B	Total = TP+FP+FN+TN = 100 (1). Accuracy=(TP+TN)/Total=(36+51)/100=87% (2). Precision=TP/(TP+FP)=36/45=80% (2). Recall=TP/(TP+FN)=36/40=90% (2). F1=2PR/(P+R)=2(0.8)(0.9)/(1.7)=0.8471=84.71% (2). Interpretation: high recall captures 90% of positives; precision means 20% of positive predictions are false (1).	10
Q.4A	(a) sales <- read.csv("sales.csv"); head(sales). Import (1), first six rows (1). [2 marks] (b) missing_count <- sum(is.na(sales$Sales)); mean_sales <- mean(sales$Sales, na.rm=TRUE); sales$Sales[is.na(sales$Sales)] <- mean_sales. Detect missing (1), impute with mean (1). [2 marks] (c) overall_mean <- mean(sales$Sales) (1); category_mean <- aggregate(Sales ~ Product_Category, data=sales, FUN=mean) (2). [3 marks] (d) barplot(category_mean$Sales, names.arg=category_mean$Product_Category, main="Mean Sales by Product Category", xlab="Product Category", ylab="Mean Sales", col="steelblue", las=2). Correct barplot/data (2), appropriate labels/title (1). [3 marks]. Accept equivalent base R or ggplot2 code.	10
	OR	
Q.4B	(a) NumPy: arrays/numerical operations (1); pandas: tabular data cleaning/aggregation (1); Matplotlib: customizable static plots (1); Seaborn: statistical plots built on Matplotlib (1); Plotly: interactive charts (1). (b) Python (4): import matplotlib.pyplot as plt months = ["Jan", "Feb", "Mar", "Apr", "May"]	10

	sales = [20, 35, 30, 45, 50] plt.plot(months, sales, marker="o") plt.title("Monthly Sales") plt.xlabel("Month"); plt.ylabel("Sales (₹ thousands)") plt.show() Interpretation (1): sales rise overall from 20 to 50, with a dip in March (35 to 30).	
Q.5A	Normalized TF = 4/100 = 0.04 (2). IDF = ln(N/DF)=ln(50/5)=ln(10)=2.302585 (3). TF–IDF = 0.04 × 2.302585 = 0.092103 ≈ 0.0921 (3). Use: weights words by importance in a document while downweighting terms common across the corpus; supports retrieval/classification (2). Natural logarithm assumed. Alternative Solution Provided Below	10
	OR	
Q.5B	Seven steps, with examples (7 × 1): 1. Language identification (English); 2. Tokenization (split into words); 3. Sentence splitting; 4. Part-of-speech tagging (noun/verb); 5. Parsing/syntactic structure; 6. Entity extraction/recognition (Mumbai = place); 7. Sentence chaining/coreference (link "she" to a named person). NER extracts named people, organizations, places etc. (1.5). Sentiment analysis labels opinions positive/negative/neutral (1.5). Equivalent seven-stage terminology consistent with taught material accepted.	10
Q.6A	Five essential design decisions (2 marks each): (1) KPIs: select meaningful cards such as revenue, margin, orders with units/timeframe. (2) Visuals: line for trends, bar for categories, map only for geographical data; avoid clutter. (3) Slicers: region/date/product filters with clear interactions. (4) DAX: accurate measures, e.g. Total Sales = SUM(Sales[Amount]); verify filter context. (5) Storytelling: arrange overview → trends → drivers → action; label axes, disclose context and avoid misleading scales.	10
	OR	
Q.6B	(a) Assuming all June sales recorded on 30 June: YTD = 50,000+60,000+80,000+45,000+70,000+95,000 = ₹4,00,000 (2); QTD (Apr–Jun)=45,000+70,000+95,000=₹2,10,000 (2); MTD (June)=₹95,000 (1). (b) Using a proper contiguous Date table and [Total Sales] measure: TOTALYTD([Total Sales], 'Date'[Date]) — year-to-date (2); TOTALMTD([Total Sales], 'Date'[Date]) — month-to-date (1.5); TOTALQTD([Total Sales], 'Date'[Date]) — quarter-to-date (1.5).	10
	********** All the Best**********	

Alternative Solution 5A: Solution 2 (Base-10 Log $\log_{10}$):

- $\text{IDF} = \log_{10}\left(\frac{50}{5}\right) = \log_{10}(10) = 1$
- $\text{TF-IDF} = 0.04 \times 1 = 0.04$