**Exercise: Building a k-NN Model for Regression**

**Objective:** Predict the 'Total' sales amount based on other features using a k-NN regression model.

Step 1: Data Preparation

1. Load the dataset into a suitable data structure (e.g., DataFrame).
2. Explore and understand the dataset (check for missing values, data types, etc.).

Step 2: Feature Selection

1. Choose relevant features for building the k-NN regression model.
2. Split the dataset into features (X) and the target variable ('Total').

Step 3: Data Preprocessing

1. Encode categorical variables if needed.
2. Standardize or normalize numerical features.

Step 4: Model Building

1. Split the data into training and testing sets.
2. Build a k-NN regression model with a suitable value of k.
3. Train the model on the training set.

Step 5: Model Evaluation

1. Make predictions on the test set.
2. Evaluate the model's performance using appropriate regression metrics (e.g., Mean Squared Error, R-squared).

Step 6: Tuning the Model

1. Experiment with different values of k to see how it affects model performance.
2. Choose the best k based on the evaluation results.

Additional Questions:

1. What features have the most impact on the 'Total' sales amount?
2. How does the choice of k affect the model's predictive performance?

**Exercise: Building a k-NN Model for Classification**

Objective:

Build a k-NN classification model to predict the likelihood of a user taking a product based on various features provided in the dataset.

Steps:

1. **Data Preprocessing:**
   * Handle any missing values in the dataset.
   * Convert categorical variables (e.g., **preferred\_device**) into numerical representations if needed.
   * Split the dataset into features (X) and the target variable (y).
2. **Feature Scaling:**
   * Scale the numerical features to ensure they have a similar range.
3. **Train-Test Split:**
   * Split the dataset into training and testing sets.
4. **Build a k-NN Model:**
   * Choose an appropriate value for k.
   * Use the training set to build the k-NN model.
5. **Model Evaluation:**
   * Use the testing set to evaluate the model's performance.
   * Calculate and display relevant classification metrics such as accuracy, precision, recall, and F1-score.
   * Experiment with different values of k to observe their impact on the model's performance.
6. **Visualization (Optional):**
   * If applicable, create visualizations such as a confusion matrix to provide a more intuitive understanding of the model's performance.

Instructions:

* Use the provided dataset (**your\_dataset.csv**).
* Comment your code appropriately to explain each step.
* Experiment with different values of k and observe how it affects the model's performance.
* Provide interpretations of the results and any insights gained from the analysis.

Submission:

Submit the completed Jupyter Notebook or Python script with the code, comments, and interpretations.