**EXPLANATION**

This notebook demonstrates several key Natural Language Processing (NLP) techniques using Python's scikit-learn library.

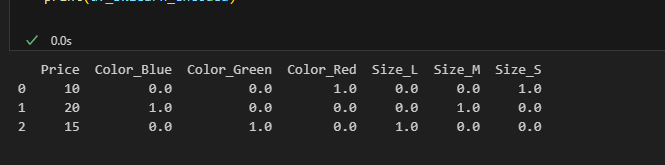
1. **One-Hot Encoding**

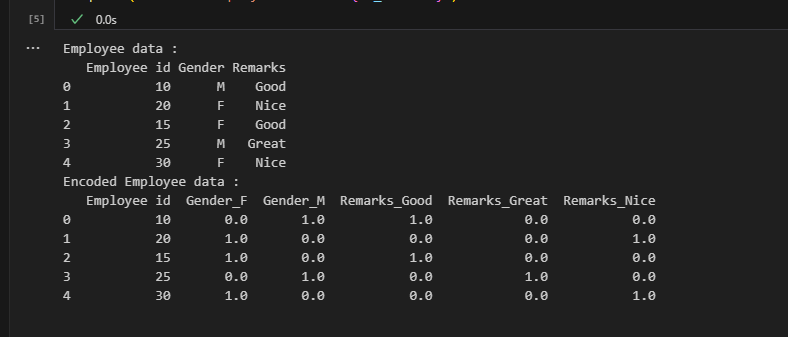
**What this does:**

* Creates a DataFrame with product data (color, size, price)
* Identifies categorical columns ('Color' and 'Size')
* Uses scikit-learn's OneHotEncoder to convert categorical values into binary columns
* Each unique value becomes a new column (e.g., Color\_Red, Size\_S)
* The original categorical columns are dropped
* The encoded columns are combined with the remaining numeric data (Price)

**Output shows:**

* Price column remains unchanged
* Each color and size value becomes a binary column (1 if present, 0 otherwise)





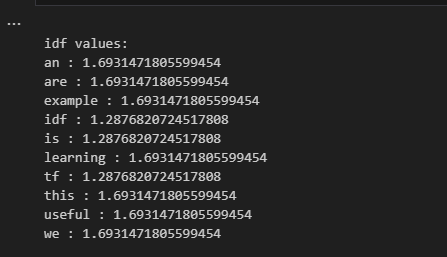
2**. TF-IDF Examples**

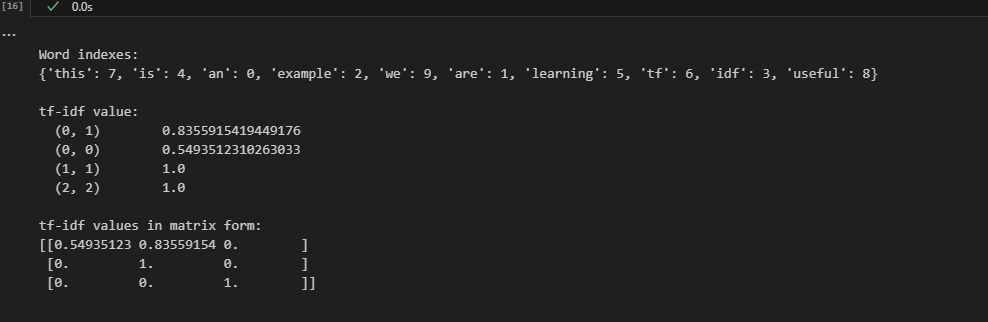
**What this does:**

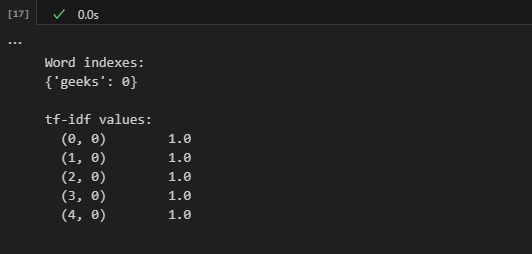
* Creates a TF-IDF vectorizer object
* Processes three text documents
* Converts text into numerical features based on term frequency-inverse document frequency

**Key features:**

* Shows the IDF (Inverse Document Frequency) values for each term
* Words that appear in more documents get lower IDF scores
* Rare words get higher IDF scores
* Demonstrates how TF-IDF weights terms based on their importance







**Scikit-learn Interfaces**:

* + Consistent .fit(), .transform(), .fit\_transform() pattern
  + Feature names extraction with get\_feature\_names\_out()
  + Sparse matrix handling for efficient storage