Clustering the Iris Dataset
A Comparative Study of KMeans and Hierarchical Clustering

The **Iris dataset**, available in sklearn, contains:

- **Features:** Four numerical measurements (sepal length, sepal width, petal length, petal width) for each flower.
- **Target column (species):** Denoting three species of Iris flowers (*setosa*, *versicolor*, *virginica*).

However, for clustering (an unsupervised learning task), we drop the target column (`species`) because clustering aims to group data points without prior knowledge of labels.

Code Breakdown:

- The dataset is loaded into a **DataFrame** for better manipulation.
- The `species` column is excluded, leaving us with the feature columns only.

A) KMeans Clustering

How KMeans Works:

1. **Initialization:** Randomly initializes kkk cluster centroids (where kkk is the desired number of clusters).
2. **Assignment:** Each data point is assigned to the nearest centroid based on a distance metric (e.g., Euclidean distance).
3. **Update Centroids:** Centroids are recalculated as the mean of all data points assigned to each cluster.
4. **Repeat:** Steps 2 and 3 are repeated until centroids stabilize or the maximum number of iterations is reached.

The algorithm optimizes the **inertia**, which is the sum of squared distances between points and their nearest cluster centroid.

Why KMeans is Suitable for the Iris Dataset:

1. **Well-Separated Clusters:** The Iris dataset has distinct feature patterns (e.g., petal length and width), making it suitable for a centroid-based approach.
2. **Fixed Number of Clusters:** The Iris dataset contains three known species, which aligns with k=3k = 3k=3 clusters in KMeans.

To visualize the clusters (which are in a 4-dimensional space), we reduce the dataset to **2 dimensions** using **PCA (Principal Component Analysis)**.
Data points are plotted in the reduced space, colored by their cluster assignments.

Output Visualization:

- A scatterplot shows the clusters formed by KMeans in 2D space, where:
    - Each color represents a cluster.
    - PCA components are used as axes for easy visualization.

B) Hierarchical Clustering

How Hierarchical Clustering Works:

Hierarchical clustering builds a hierarchy of clusters. The process can be:

1. **Agglomerative (Bottom-Up):**
    - Start with each data point as its own cluster.
    - Iteratively merge the closest clusters based on a **linkage criterion** (e.g., Ward's method, single linkage).
2. **Divisive (Top-Down):**
    - Start with a single cluster containing all points.
    - Recursively split clusters into smaller groups.

The result is represented as a **dendrogram**, showing how clusters are merged at different distance thresholds.

Why Hierarchical Clustering is Suitable for the Iris Dataset:

1. **Small Dataset:** Hierarchical clustering is computationally expensive, but it works well for the small size of the Iris dataset.

2. **Exploration of Cluster Structure:** The dendrogram allows us to analyze cluster formation at various levels of granularity.
3. **Dendrogram:**
    a. Displays how clusters merge.
    b. The vertical axis represents the distance at which clusters merge.
    c. Horizontal cuts through the dendrogram correspond to k=3k = 3k=3 clusters.
4. **2D PCA Visualization:**
    a. As with KMeans, the reduced 2D space is used to visualize cluster assignments.