Problem Statement: Image Classification Using Deep Learning

You are provided with a dataset organized into four subfolders, each representing a distinct class of images:

- Cloudy
- Desert
- Green Area
- Water

Each subfolder contains a set of images corresponding to its class. The goal is to perform image classification by developing and evaluating multiple deep learning models.

Task Requirements:

You are required to implement and compare the performance of the following models:

1. Use the ConvNeXt and Swin Transformer architectures for classification.

2. Utilize the VGG19 architecture with pretrained weights. Implement transfer learning and fine-tuning to adapt the model for the 4-class classification task.

3. Design a custom CNN consisting of:

- Three Convolutional Blocks, each comprising:
  - ✔ Convolutional Layer
  - ✔ Batch Normalization Layer
  - ✔ Activation Function (choose from ReLU, Sigmoid, Softmax)
  - ✔ Pooling Layer (choose from Max Pooling, Average Pooling, Global Pooling)

- One Classification Block, consisting of:
  - ✔ Flatten Layer
  - ✔ Two Dropout Layers
  - ✔ Two Fully Connected (Dense) Layers
  - ✔ Final Output Layer (choose an appropriate activation: ReLU, Sigmoid, or Softmax)

Training and Evaluation Configuration

- **Data Splitting:**
  Partition the dataset into three subsets: training, validation, and testing to ensure effective model evaluation and generalization.

- **Training Settings:**
  Train each model for 10 epochs using optimal hyper parameters and techniques suitable for multi-class classification.

- **Model Optimization:**
  Select an appropriate optimizer, loss function, and evaluation metrics (such as accuracy, precision, recall, and F1-score) based on the characteristics of the dataset.

- **Data Handling Decisions:**
  Evaluate the need for data augmentation to enhance model robustness and reduce overfitting. Consider applying oversampling techniques if class imbalance is observed.

- **Performance Visualization:**
  Plot training, validation, and testing accuracy and loss curves to analyze model learning behavior.
  Generate and present the confusion matrix along with key performance indicators:
    1. Precision
    2. Sensitivity (Recall)
    3. Specificity