Question 1: Project Overview

This project aims to predict **age and gender** from live camera feeds using **deep learning and computer vision techniques**. The system utilizes **Convolutional Neural Networks (CNNs)** to classify a person's **age group** and **gender** based on facial features.

Key Steps:

1. **Face Detection:** Uses pre-trained models like **MTCNN, Haar Cascades, or SSD** to detect faces from webcam video frames.
2. **Preprocessing:** Crops, resizes, and normalizes face images for uniform input.
3. **Feature Extraction:** CNN-based models extract deep facial features.
4. **Classification:**
    - **Age Group:** Categorized into predefined groups (e.g., 0-10, 11-20, 21-30, etc.).
    - **Gender:** Binary classification (Male/Female).
5. **Real-time Display:** The system overlays age and gender predictions on live video streams.

Applications:

- **Security & Surveillance** (age verification, identity authentication).
- **Retail & Marketing** (customer segmentation).
- **Healthcare** (age-based medical recommendations).

Question 2: Dataset Details

- **Collector(s):** UTK researchers
- **Year:** 2017
- **Title of Dataset:** UTKFace
- **Version Number:** Latest available version
- **Publisher:** UTK researchers
- **DOI or URL:** UTKFace Dataset
- **Study/Paper/Reason:**
    a) UTKFace provides a large-scale dataset for **age estimation** and **gender classification**.
    b) It contains images labeled with **age, gender, and ethnicity annotations**.
    c) The dataset is widely used in facial analysis research.

Question 3: Language and Libraries

- **Language:** Python 3.x
- **Libraries:**
    - **Machine Learning & Deep Learning:** TensorFlow, Keras, PyTorch
    - **Computer Vision:** OpenCV, Dlib
    - **Data Handling:** Pandas, NumPy
    - **Visualization:** Matplotlib, Seaborn
    - **Web Framework:** Flask or Django

Question 4: Code Development

Parts of the code that will be written from scratch:

1. **Face Detection & Preprocessing**
    - Implementing **MTCNN or Haar Cascades** for face detection.
    - Applying **image resizing, normalization, and augmentation** (flipping, rotation, noise).
2. **CNN Model for Age & Gender Prediction**
    - Designing a **custom CNN architecture** for classification.
    - Implementing **transfer learning** using models like ResNet, VGG16, or MobileNet.
3. **Real-time Video Processing & Inference**
    - **Capturing frames from a webcam** using OpenCV.
    - Overlaying **predictions on the live video feed**.
4. **Web Application Development**
    - **Backend:** Flask/Django API for inference.
    - **Frontend:** HTML, CSS, JavaScript for displaying real-time results.

Question 5: Best Choice of Model(s) and Justification

- **ResNet50 (Pretrained on ImageNet):**
    - Deep and accurate, works well for face-based tasks.
- **MobileNetV2:**
    - Lightweight and optimized for real-time applications.
- **Custom CNN Model (for Fine-Tuned Performance):**
    - Trained from scratch to suit specific dataset requirements.
- **Hybrid Approach (CNN + LSTM for Temporal Analysis in Video):**
    - Useful for improving accuracy in **sequential frame-based age estimation**.

Question 6: Hyperparameters and Optimization Strategy

Key Hyperparameters:

- **Learning Rate:** Initially set at **0.001**, adjusted dynamically.
- **Batch Size: 32 or 64** (adjusted based on memory constraints).
- **Number of CNN Layers: 4–6 layers** for feature extraction.
- **Dropout Rate: 0.3–0.5** to prevent overfitting.
- **Optimizer: Adam / RMSprop** for better convergence.
- **Loss Function:**
    - **Categorical Cross-Entropy** (for age classification).
    - **Binary Cross-Entropy** (for gender classification).
- **Regularization Techniques:**
    - **Batch Normalization:** Speeds up training and stabilizes learning.
    - **Dropout:** Prevents overfitting.
    - **Data Augmentation:** Improves generalization.

Optimization Strategy:

- **Grid Search and Random Search** for hyperparameter tuning.
- **Cross-validation** for model evaluation.

Question 7: Model Evaluation Metrics

- **Accuracy:** Measures overall prediction correctness.
- **Precision & Recall:** Evaluates performance on **age groups and gender classes**.
- **F1-score:** Balances **precision and recall**.
- **Confusion Matrix:** Analyzes misclassifications.
- **Inference Latency:** Ensures the model runs efficiently in real-time.