**Assignment 1: Developing a Virtual Try-On Application Using Existing Repositories**

**Step:1 – clone the repo**

!git clone <https://github.com/Zheng-Chong/CatVTON.git>

!git clone <https://github.com/yisol/IDM-VTON.git>

**Step:2- DataSet(From hugging face site)**

from datasets import load\_dataset

ds = load\_dataset("lirus18/deepfashion")

**Introduction:**

The aim of this project is to build a virtual try-on system using CAT-VTON and IDM-VTON, two state-of-the-art models. The system allows users to see how different clothing items would look on them virtually, enhancing online shopping experiences.

**Dataset**:

Utilizing the "lirus18/deepfashion" dataset from Hugging Face, which includes images of models and clothing items (tops, dresses, etc.), suitable for try-on applications.

**Input image:**

****

**Output image:**

****

**Challenges and Solutions:**

**Challenge 1: Checkpoint Availability and Compatibility**

**Solution:**

To address this, I first searched for compatible checkpoints within the official repositories or any linked resources. In cases where checkpoints were not readily available, I explored training a small subset of the DeepFashion dataset to generate new checkpoints. I adjusted model configurations to make them compatible with the available dataset properties, such as modifying input image resolutions or fine-tuning specific layers to accommodate the new data structure.

**Challenge 2: Dataset-Model Input Format Mismatch**

**Solution:**

I implemented a preprocessing pipeline to format the dataset according to model requirements. This involved generating or inferring missing elements like pose estimations or segmentation maps. Tools like OpenPose or DensePose were helpful for pose estimation. Additionally, I resized and normalized images to match the input dimensions expected by the models. In cases where specific data (e.g., paired images) was unavailable, I curated the dataset to create similar conditions for optimal model performance.