```
#!/bin/bash

# Environment Variables
export MODEL_NAME=meta-llama/Meta-Llama-3-8B
export HF_TOKEN=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  # Change to your own huggingface token.
export HF_HUB_ENABLE_HF_TRANSFER=True

# Setup
conda activate vllm
if [ $? -ne 0 ]; then
    conda create -n vllm python=3.10 -y
    conda activate vllm
fi

pip install vllm==0.4.0.post1
pip install gradio openai
pip install flash-attn
pip install hf_transfer

# Function to print colored log statements
log() {
    local GREEN='\033[0;32m'
    local NC='\033[0m' # No Color
    echo -e "${GREEN}[LOG] $1${NC}"
}
```

```
# Run VLM
conda activate vllm
log "Starting vllm api server..."
export PATH=$PATH:/sbin

python3 -u -m vllm.entrypoints.openai.api_server \
    --port 8090 \
    --model $MODEL_NAME \
    --trust-remote-code --tensor-parallel-size 4 \
    --gpu-memory-utilization 0.95 \
    --max-num-seqs 64 \
    >> /var/log/vllm_api.log 2>&1 &

# Check if VLM server started successfully
if [ $? -eq 0 ]; then
    log "VLLM API server started successfully."
else
    log "Failed to start VLLM API server."
    exit 1
fi

# Run Gradio
log "Starting gradio server..."
git clone https://github.com/vllm-project/vllm.git || true
```

```
python3 vllm/examples/gradio_openai_chatbot_webserver.py \
    -m $MODEL_NAME \
    --port 8811 \
    --model-url http://localhost:8081/v1 \
    --stop-token-ids 128009,128001 \
    >> /var/log/gradio_server.log 2>&1 &

# Check if Gradio server started successfully
if [ $? -eq 0 ]; then
    log "Gradio server started successfully."
else
    log "Failed to start Gradio server."
    exit 1
fi
```

919039