

# Run codes in Ubuntu



**HCI.py**  
Python

**HCI2.py**  
Python

**HCI3.py**  
Python

## HOW TO RUN THIS ALL CODES IN UBANTU TERMINAL

Perfect You've uploaded three Python files — **HCI.py**, **HCI2.py**, and **HCI3.py** — each for different Human–Computer Interaction (HCI) projects.

Here's exactly how to **run all three in the Ubuntu terminal** (step by step):

### **1** Install Dependencies (only once)

Open your **Ubuntu terminal** and run:

```
bash

sudo apt update
sudo apt install python3 python3-pip python3-opencv -y
pip install pygame mediapipe deepface tensorflow keras
```

This installs everything needed:

- **pygame** → for HCI.py (the game)
- **mediapipe**, **opencv-python** → for HCI2.py (hand gesture detection)
- **deepface**, **tensorflow**, **keras**, **opencv-python** → for HCI3.py (emotion detection)

### **2** Run the First Program (Game) — **HCI.py**

1. Navigate to the folder containing your files (for example, if they're in Downloads):

```
bash

cd ~/Downloads
```

2. Run the game:

```
bash
python3 HCI.py
```

💡 Use the **arrow keys** or **on-screen buttons** to move and avoid falling blocks.  
Press **R** to restart or **Q** to quit.

## 👉 3 Run the Second Program (Hand Gesture Control) — HCI2.py

In the same folder:

```
bash
python3 HCI2.py
```

⌚ It will open your webcam and track your **hand movements** using **MediaPipe**.

- If thumb tip is higher than the index finger → it shows “ **Thumbs Up Detected**”
  - Otherwise → it shows “ **Hand Detected**”
- Press **q** to quit.

## 😊 4 Run the Third Program (Emotion Detection) — HCI3.py

Run:

```
bash
python3 HCI3.py
```

🧠 It uses **DeepFace** to analyze your emotions in real-time via webcam.  
Detected emotion (like *happy*, *sad*, *angry*, etc.) appears on the screen.  
Press **q** to quit.

## ⚠ 5 Common Issues and Fixes

Problem	Solution
<b>ModuleNotFoundError: No module named 'cv2'</b>	Run <b>pip install opencv-python</b>
<b>ModuleNotFoundError: No module named 'pygame'</b>	Run <b>pip install pygame</b>
<b>Could not find a version that satisfies mediapipe</b>	Make sure Python ≥ 3.8. Run <b>python3 --version</b>
Camera not working	Check if another app is using it, or run <b>sudo apt install v4l-utils</b>
TensorFlow not installing	Use <b>pip install tensorflow==2.12</b> (works with DeepFace)

Would you like me to create a **single bash script** (`run_hci.sh`) that installs everything and runs whichever file you choose automatically?

That would let you just run one command like:

```
bash  
bash run_hci.sh
```



PS C:\Users\HP> sudo apt update

Sudo is disabled on this machine. To enable it, go to the Developer Settings page in the Settings app

Got it — that message tells us you're **not actually inside Ubuntu yet**, but rather inside **Windows PowerShell** (that's why the prompt says `PS C:\Users\HP>`).

Let's fix that step by step

## ➡️🐧 HOW TO OPEN A REAL UBUNTU TERMINAL

### ⌚ Step 1: Launch Ubuntu properly

1. Press **Start** (Windows key).
2. Search for "**Ubuntu**" — you should see something like:

```
nginx  
Ubuntu 22.04 LTS
```

3. Click it to open the **Ubuntu terminal** (you'll know you're in the right place if your prompt looks like this):

```
ruby  
vishnu@DESKTOP-xxxx:~$
```

not `PS C:\Users\HP>`.

### ⌚ Step 2: Update and install required packages

Now copy-paste these commands **in the Ubuntu terminal** (not PowerShell):

```
bash  
  
sudo apt update  
sudo apt install python3 python3-pip python3-opencv -y  
pip install pygame mediapipe deepface tensorflow keras
```

You'll be asked for your Ubuntu password — type it (nothing will show while typing) and press **Enter**.

---

## 📁 Step 3: Go to your project folder

If your Python files are in your Windows *Downloads* folder, in Ubuntu it's mounted under `/mnt/c/Users/HP/Downloads`.

Run:

```
bash  
cd /mnt/c/Users/HP/Downloads
```

You can check your files are there:

```
bash  
ls
```

You should see:

```
HCI.py  HCI2.py  HCI3.py
```

## ⚡ Step 4: Run the programs

Now you can run any one of them:

```
bash  
python3 HCI.py
```

or

```
bash  
python3 HCI2.py
```

or

```
bash  
python3 HCI3.py
```

## ⚠ Step 5: If you get “no display found” or pygame window doesn’t open

That happens because Ubuntu inside Windows (WSL) doesn't have a graphical display by default.

You have **two options**:

## ✓ Option A — Run directly in Windows (recommended for graphical apps)

Open **PowerShell** and run:

```
bash
python HCI.py
```

But make sure Python and all required modules are installed on Windows (using `pip install`).

## 🧠 Option B — Use a Linux GUI setup (advanced)

You can install a Windows X Server (like **VcXsrv** or **X410**) and set up display forwarding — but for simplicity, running in Windows is easier for Pygame, OpenCV, and DeepFace apps.

---

Would you like me to give you the exact **commands to set up and run everything from Windows (no Ubuntu needed)** instead? That's the easier and faster way for your HCI projects.



can you give me separate dependencies download command

Absolutely ✓ — here are **separate dependency installation commands** for each of your three Python files (`HCI.py`, `HCI2.py`, `HCI3.py`), both for **Ubuntu** and **Windows (PowerShell)**.

## 🐧 For Ubuntu Terminal (inside Ubuntu, not PowerShell)

### 🎮 1 For `HCI.py` — (Pygame game)

```
bash
sudo apt update
sudo apt install python3 python3-pip -y
pip install pygame
```

### 👉 2 For `HCI2.py` — (Hand Gesture Detection)

```
bash
sudo apt install python3-opencv -y
pip install mediapipe opencv-python
```

### 😊 3 For `HCI3.py` — (Emotion Detection with DeepFace)

```
bash

sudo apt install python3-opencv -y
pip install deepface tensorflow keras opencv-python
```

## 💻 For Windows PowerShell (if you want to run directly in Windows)

Run each command **without sudo**, like this:

🧩 1 HCI.py

```
bash

pip install pygame
```

👉 2 HCI2.py

```
bash

pip install mediapipe opencv-python
```

😊 3 HCI3.py

```
bash

pip install deepface tensorflow keras opencv-python
```

## ✓ Verify Installation (Optional)

After installing, you can confirm everything is set up correctly:

```
bash

python3 -m pip show pygame mediapipe deepface tensorflow opencv-python
```

If it lists versions for all, you're good to go 🎉

Would you like me to combine all these into a single `requirements.txt` file (so you can just run one command like `pip install -r requirements.txt`)?



for hci 3 and hci 2 i have attached separate camera to ubuntu so what should i do to run

Perfect — that's a great setup 🙌

If you've attached an **external (USB) camera** to your Ubuntu system, here's exactly how to make it work with your **HCI2.py** (hand gesture) and **HCI3.py** (emotion detection) scripts.

---

## Step-by-Step Guide

### Step 1: Check if Ubuntu detects your camera

Open your Ubuntu terminal and run:

```
bash  
  
ls /dev/video*
```

You should see something like:

```
bash  
  
/dev/video0  
/dev/video1
```

👉 Usually:

- `/dev/video0` = internal laptop camera (if you have one)
  - `/dev/video1` = your **external USB camera**
- 

### Step 2: Test the external camera with ffplay or cheese

Install a quick camera viewer:

```
bash  
  
sudo apt install ffmpeg cheese -y
```

Then test:

```
bash  
  
cheese
```

or

```
bash  
  
ffplay /dev/video1
```

If you see your camera feed —  it's working.

---

### Step 3: Modify your Python code to use that camera

By default, both your scripts have this line:

```
python  
cap = cv2.VideoCapture(0)
```

That **0** refers to the **default camera**.

If your external camera is **/dev/video1**, change it to:

```
python  
cap = cv2.VideoCapture(1)
```

You can test different numbers (**0, 1, 2**) if unsure.

---

## 🔴 Step 4: Run the scripts

Now, navigate to your folder:

```
bash  
cd /mnt/c/Users/HP/Downloads
```

Run the hand gesture one:

```
bash  
python3 HCI2.py
```

Run the emotion detection one:

```
bash  
python3 HCI3.py
```

---

## 📌 Step 5: Verify OpenCV and camera access

If you get an error like:

```
sql  
[ WARN:0@0.123] global cap_v4l.cpp:... can't open camera by index
```

Then try:

1. Swap the index (e.g., **VideoCapture(1) → VideoCapture(0)**)
  2. Ensure the camera isn't in use by another app.
  3. Replug the camera and check again using **ls /dev/video\***.
-

## 🧠 Step 6 (Optional): List all connected cameras programmatically

You can add this small test script to see available cameras:

```
python

import cv2

for i in range(5):
    cap = cv2.VideoCapture(i)
    if cap.isOpened():
        print(f"Camera index {i} is available")
    cap.release()
```

Run it:

```
bash

python3 camera_check.py
```

It will tell you exactly which camera index to use.

---

Would you like me to modify your **HCI2.py** and **HCI3.py** so they **automatically detect and use the working camera** instead of hardcoding the index?