

Nama:

**Muhammad Fauzi Azizi (122140106)**

Tugas Ke:

**Worksheet 1: Setup Python**

**Environment untuk Multimedia**

Mata Kuliah: **Sistem Teknologi Multimedia (IF25-40305)** Tanggal: August 25, 2025

# Tujuan Pembelajaran

Setelah menyelesaikan worksheet ini, mahasiswa diharapkan mampu:

* Memahami pentingnya manajemen environment Python untuk pengembangan multimedia
* Menginstall dan mengkonfigurasi Python environment menggunakan conda, venv, atau uv
* Menginstall library-library Python yang diperlukan untuk multimedia processing
* Memverifikasi instalasi dengan mengimpor dan menguji library multimedia
* Mendokumentasikan proses konfigurasi dan hasil pengujian dalam format LATEX

# Latar Belakang

Python telah menjadi bahasa pemrograman yang sangat populer untuk multimedia processing karena memiliki ekosistem library yang sangat kaya. Namun, untuk dapat bekerja dengan multimedia secara efektif, kita perlu mengatur environment Python dengan benar dan menginstall library-library yang tepat.

Manajemen environment Python sangat penting untuk:

* Menghindari konflik antar library (dependency conflict)
* Memastikan reproducibility dari project
* Memudahkan kolaborasi antar developer
* Memisahkan project yang berbeda dengan requirement yang berbeda

# Instruksi Tugas

## Persiapan

**Sebelum memulai, pastikan Anda telah:**

* Menginstall Python 3.8 atau lebih baru di sistem Anda
* Memilih salah satu tool manajemen environment: **conda**, **venv**, atau **uv**
* Membuka terminal/command prompt
* Menyiapkan dokumen LATEX ini untuk dokumentasi

## Bagian 1: Membuat Environment Python

Pilih **SALAH SATU** dari tiga opsi berikut dan ikuti langkah-langkahnya:

**3.2.1 Opsi 1: Menggunakan Conda (Direkomendasikan untuk pemula)**

Jalankan perintah berikut di terminal:

|  |
| --- |
| # Membuat environment baru dengan nama 'multimedia' conda create -n multimedia python=3.11  # Mengaktifkan environment conda activate multimedia  # Verifikasi environment aktif conda info --envs |

1

2

3

4

5

6

7

8

Kode 1: Membuat environment dengan Conda

**3.2.2 Opsi 2: Menggunakan venv (Built-in Python)**

|  |
| --- |
| # Membuat environment baru python3 -m venv multimedia-env  # Mengaktifkan environment (Linux/Mac) source multimedia-env/bin/activate  # Mengaktifkan environment (Windows)  # multimedia-env\Scripts\activate  # Verifikasi environment aktif which python |

1

2

3

4

5

6

7

8

9

10

11

Kode 2: Membuat environment dengan venv

**3.2.3 Opsi 3: Menggunakan uv (Modern dan cepat)**

|  |
| --- |
| # Install uv terlebih dahulu jika belum ada  # pip install uv  # Membuat environment baru uv venv multimedia-uv  # Mengaktifkan environment (Linux/Mac) source multimedia-uv/bin/activate  # Mengaktifkan environment (Windows)  # multimedia-uv\Scripts\activate  # Verifikasi environment aktif which python |

1

2

3

4

5

6

7

8

9

10

11

12

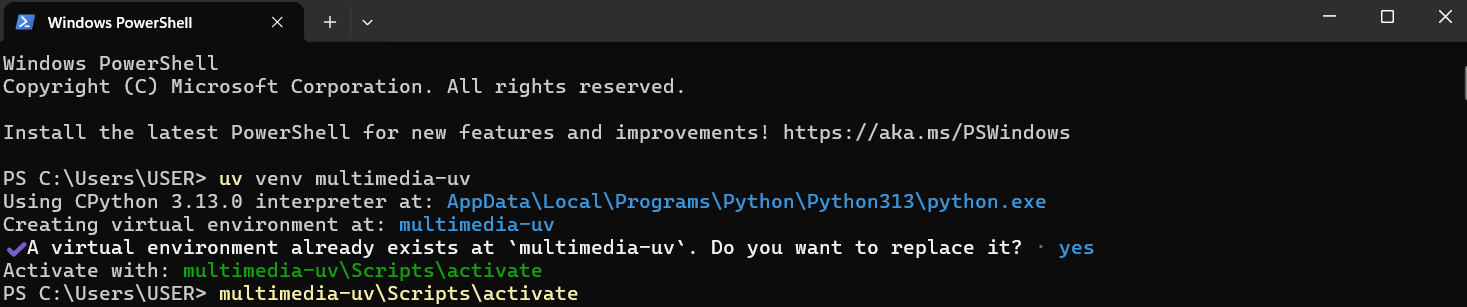
13

14

Kode 3: Membuat environment dengan uv

**Dokumentasikan di sini:**

* Tool manajemen environment yang Anda pilih: **UV**
* Screenshot atau copy-paste output dari perintah verifikasi environment



## Bagian 2: Instalasi Library Multimedia

Setelah environment aktif, install library-library berikut:

**3.3.1 Library Audio Processing**

|  |
| --- |
| # Untuk conda: conda install -c conda-forge librosa soundfile scipy  # Untuk pip (venv/uv):  pip install librosa soundfile scipy |

1

2

3

4

5

Kode 4: Instalasi library audio

**3.3.2 Library Image Processing**

|  |
| --- |
| # Untuk conda: conda install -c conda-forge opencv pillow scikit-image matplotlib  # Untuk pip (venv/uv):  pip install opencv-python pillow scikit-image matplotlib |

1

2

3

4

5

Kode 5: Instalasi library image

**3.3.3 Library Video Processing**

|  |
| --- |
| # Untuk conda: conda install -c conda-forge ffmpeg pip install moviepy  # Untuk pip (venv/uv): pip install moviepy |

Kode 6: Instalasi library video

**3.3.4 Library General Purpose**

|  |
| --- |
| # Untuk conda: conda install numpy pandas jupyter  # Untuk pip (venv/uv):  pip install numpy pandas jupyter |

1

2

3

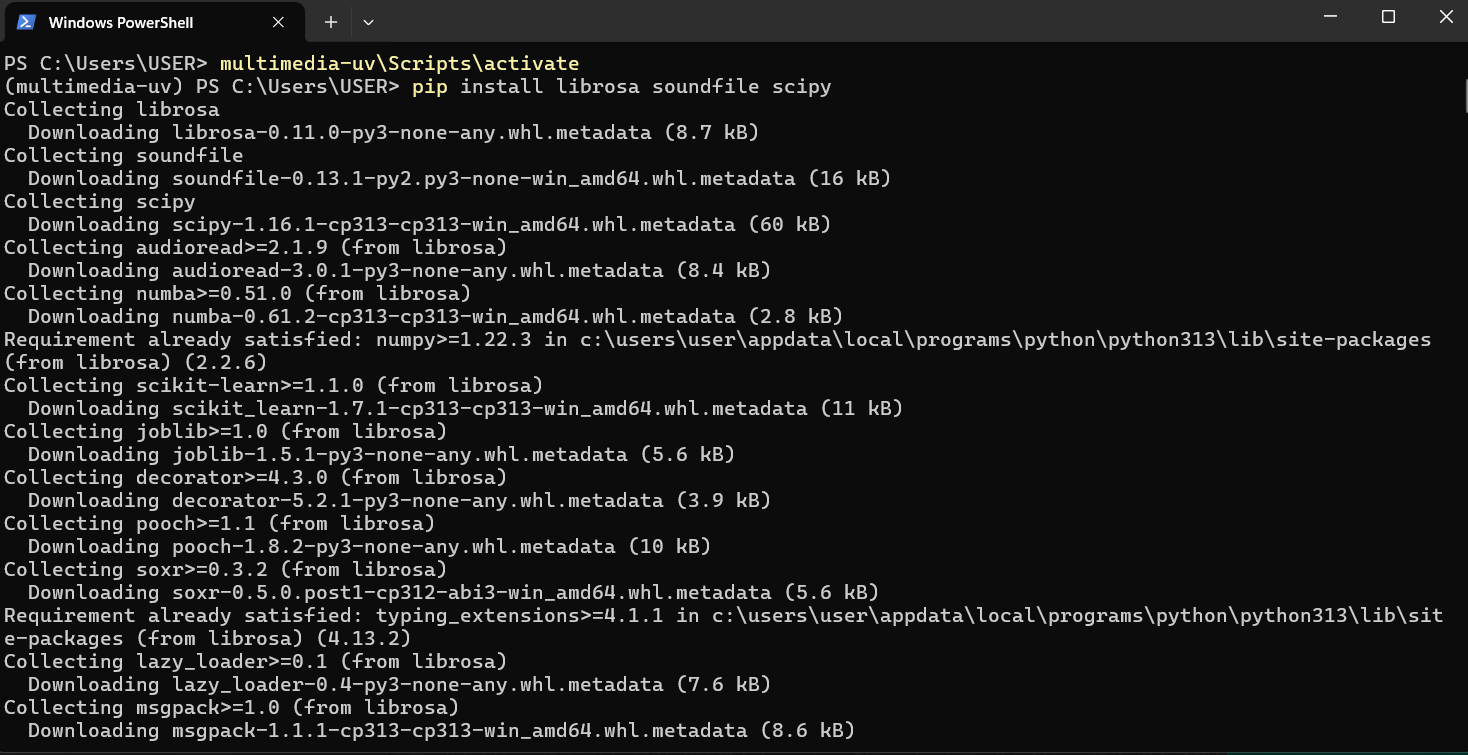
4

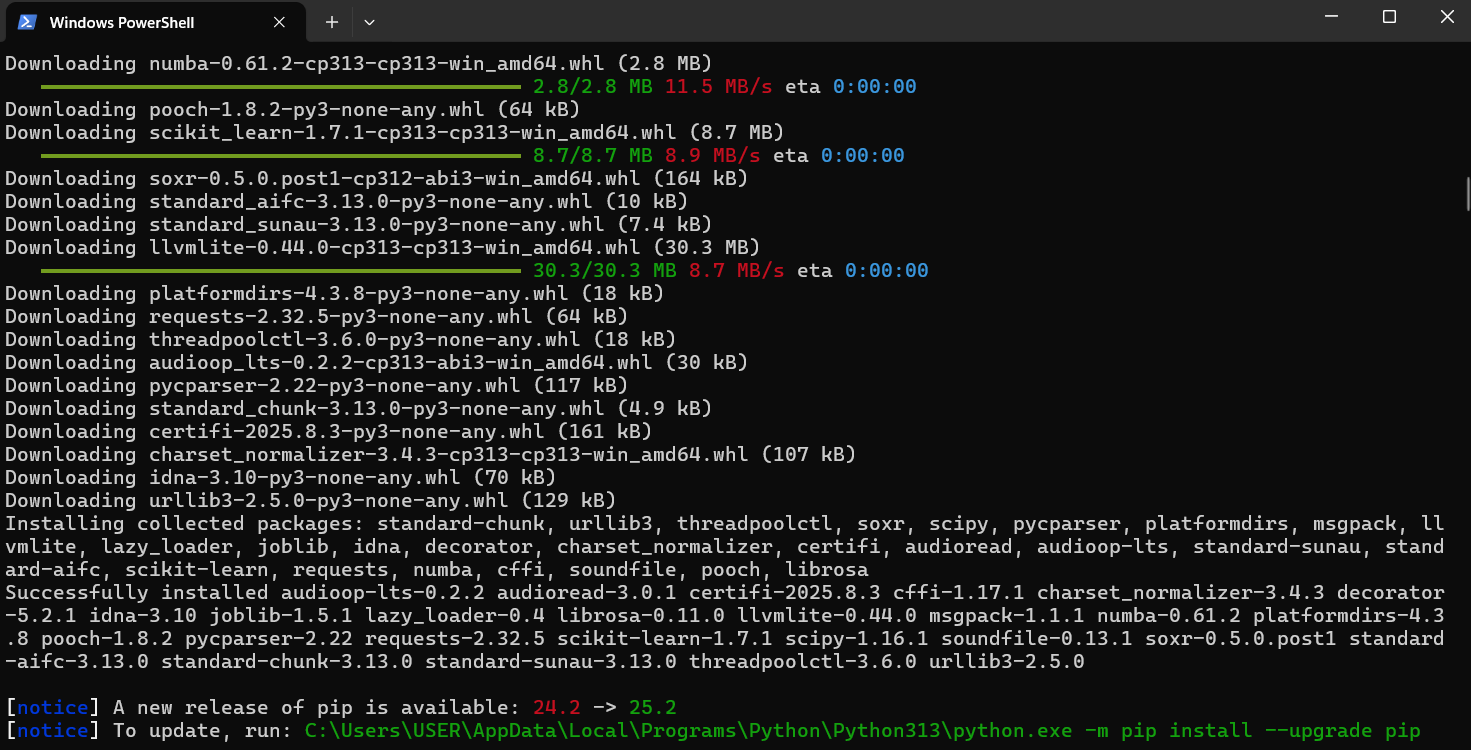
5

Kode 7: Instalasi library umum

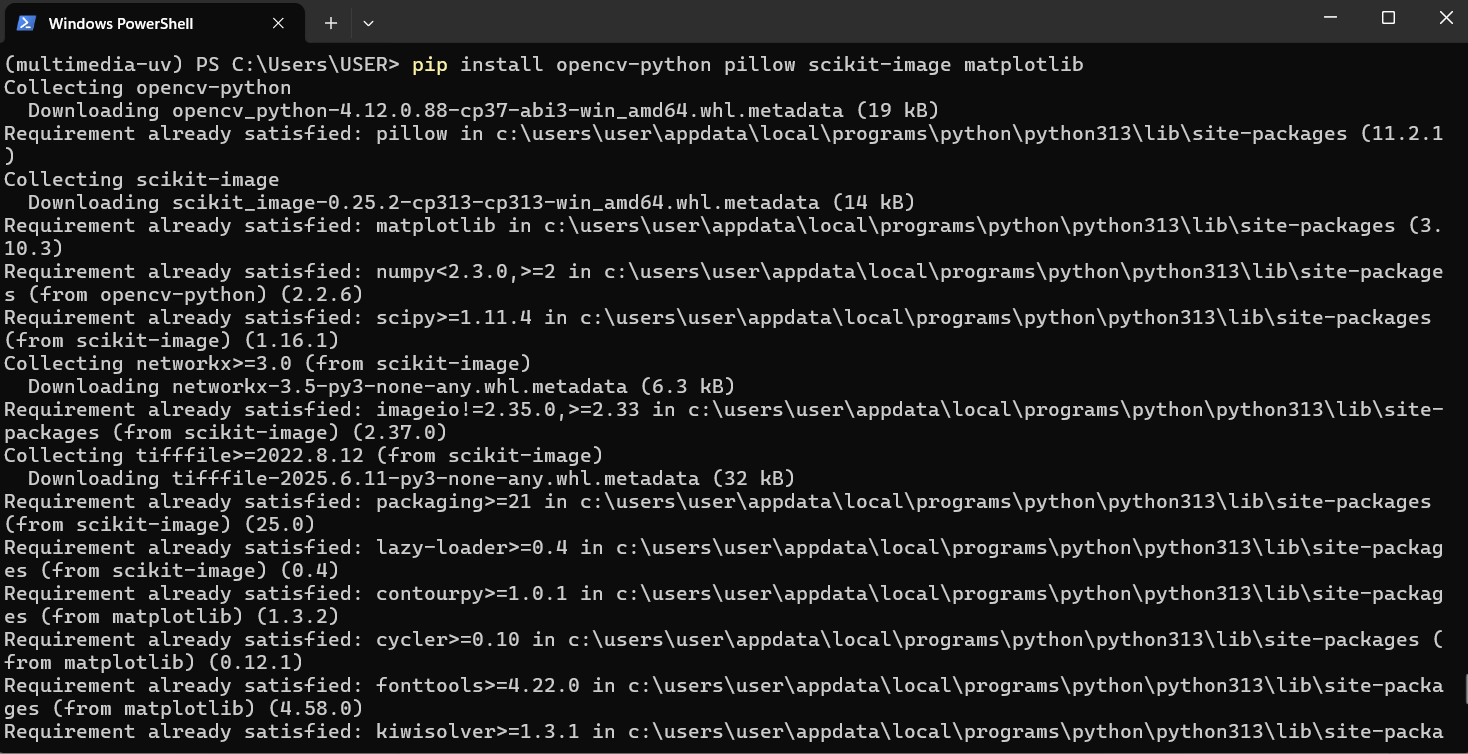
**Dokumentasikan di sini:**

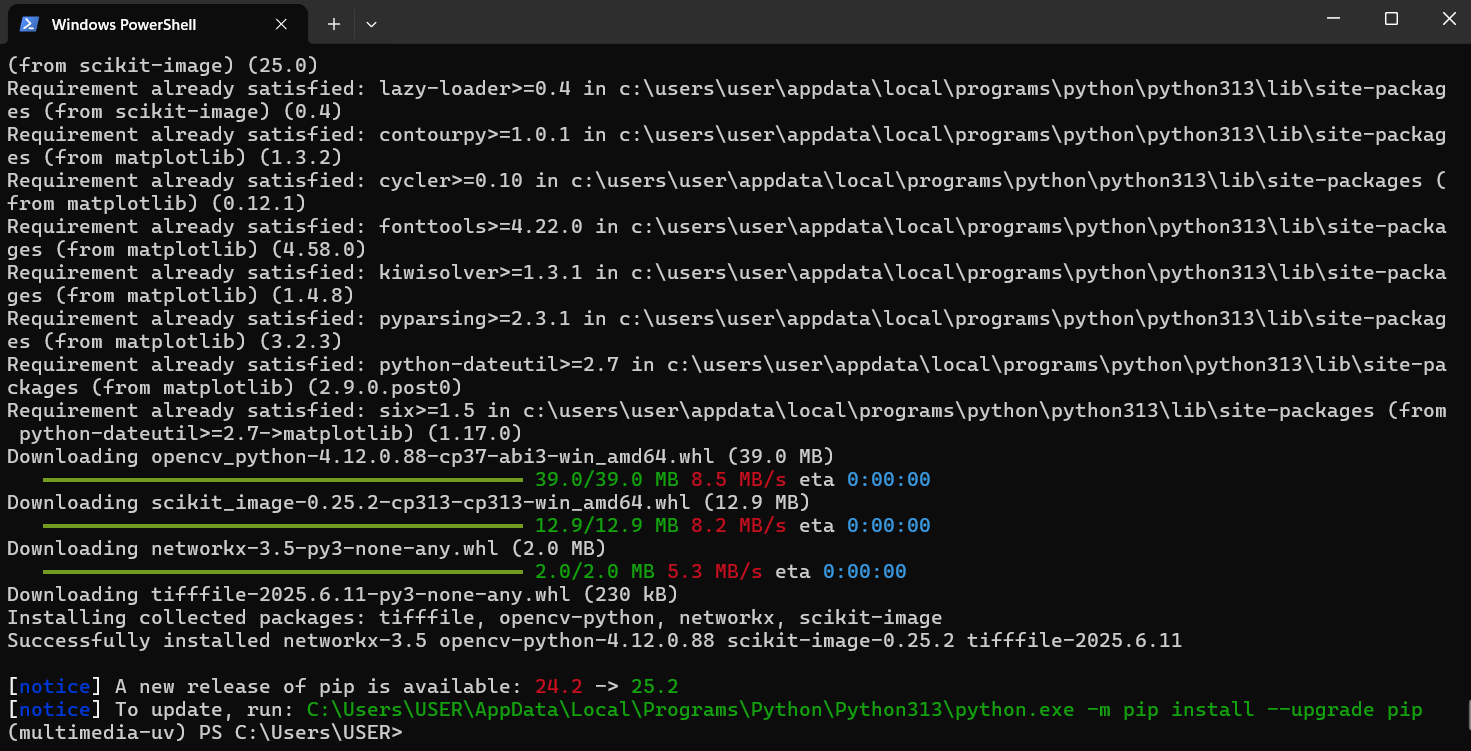
1. Install audio processing



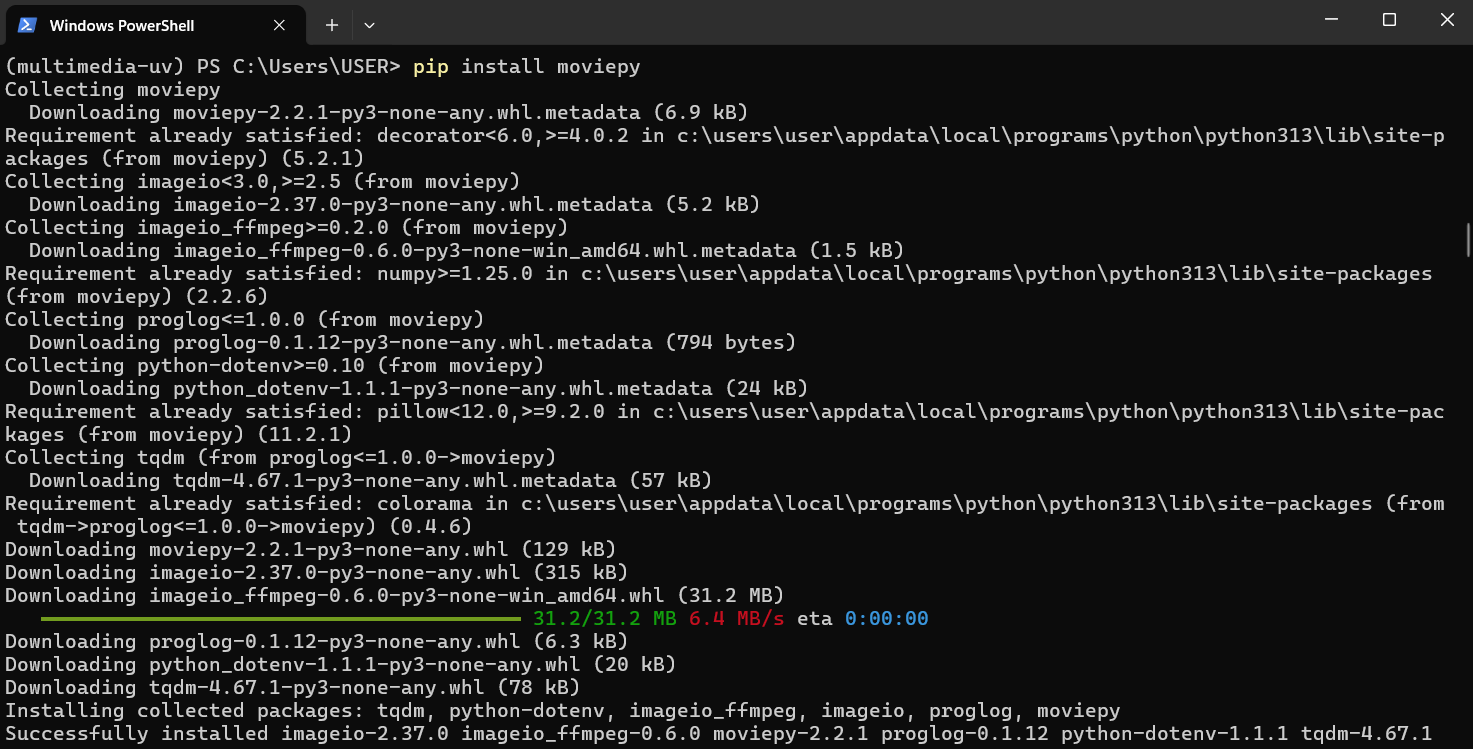


1. Install image processing

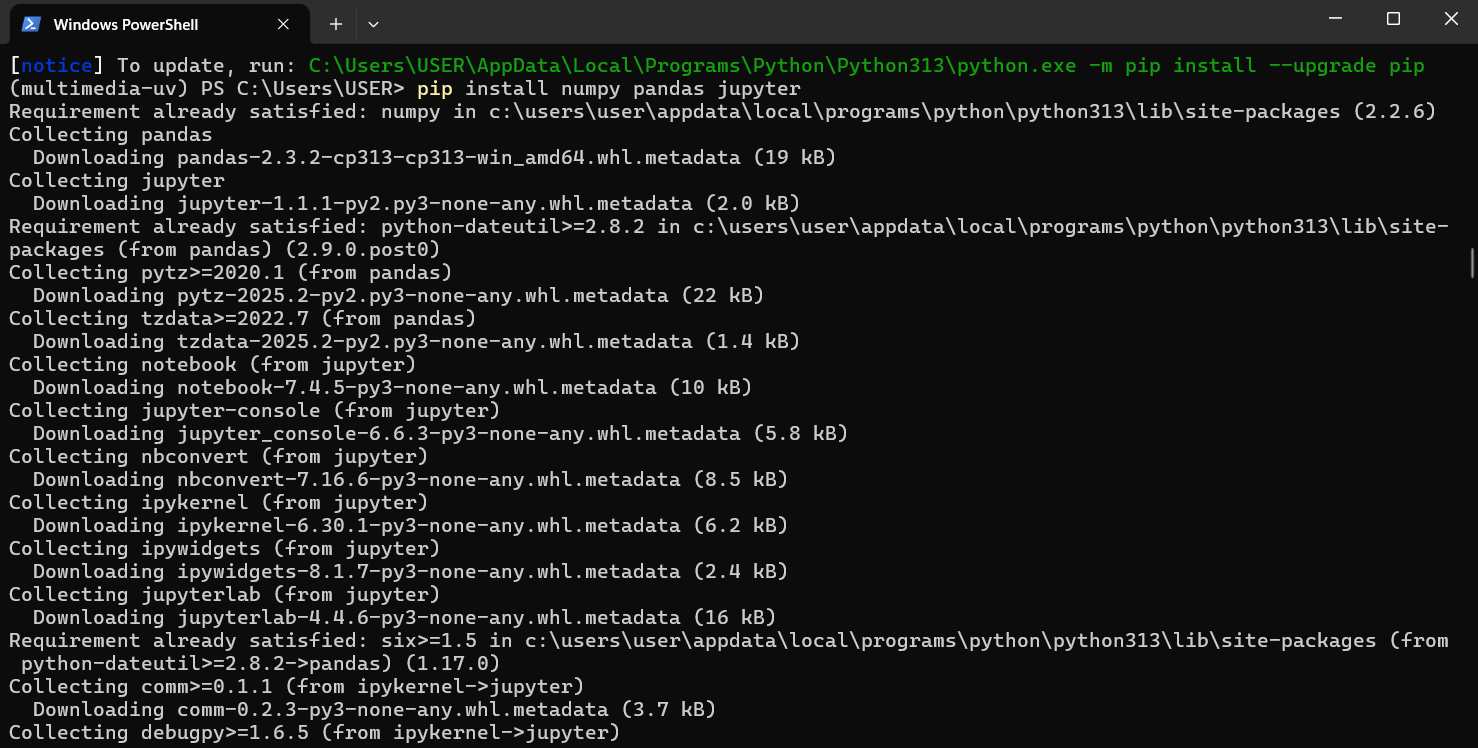


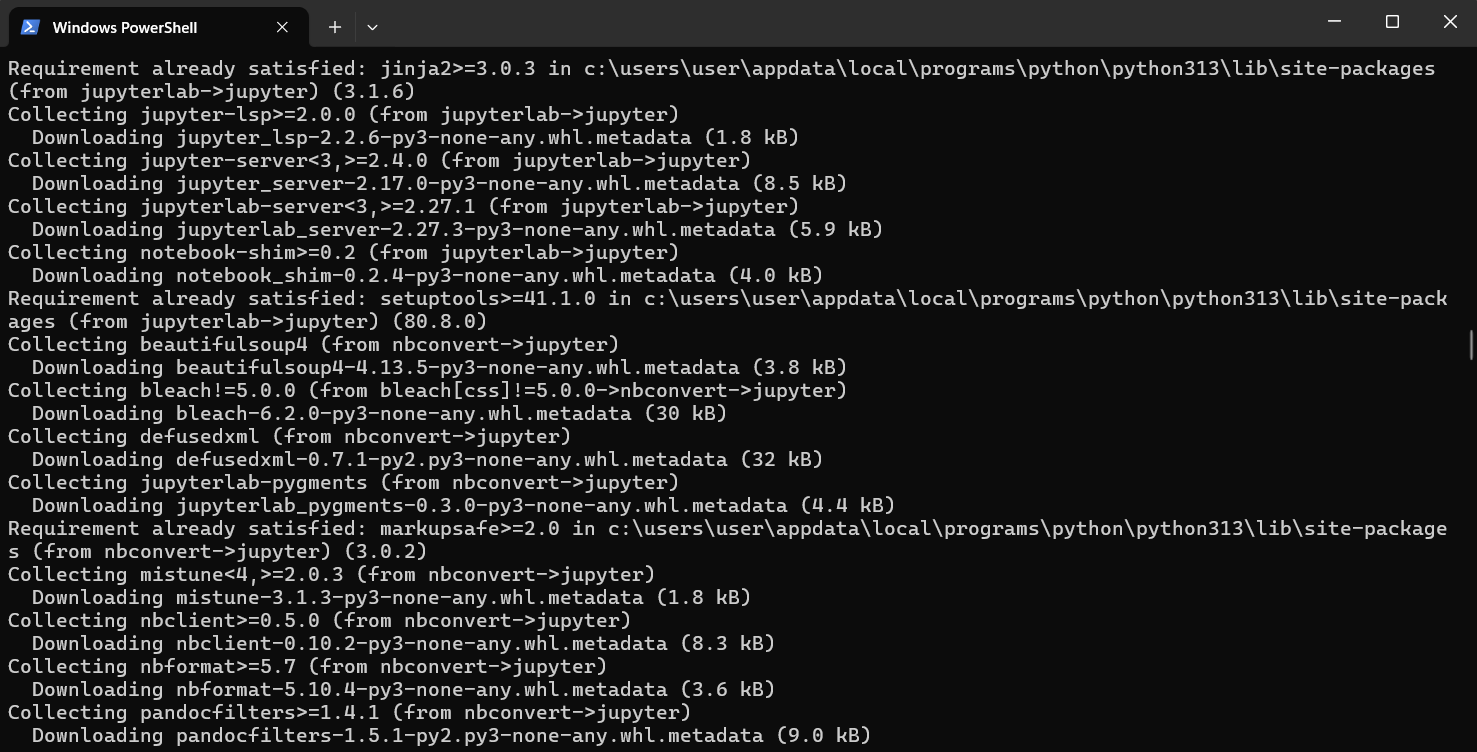


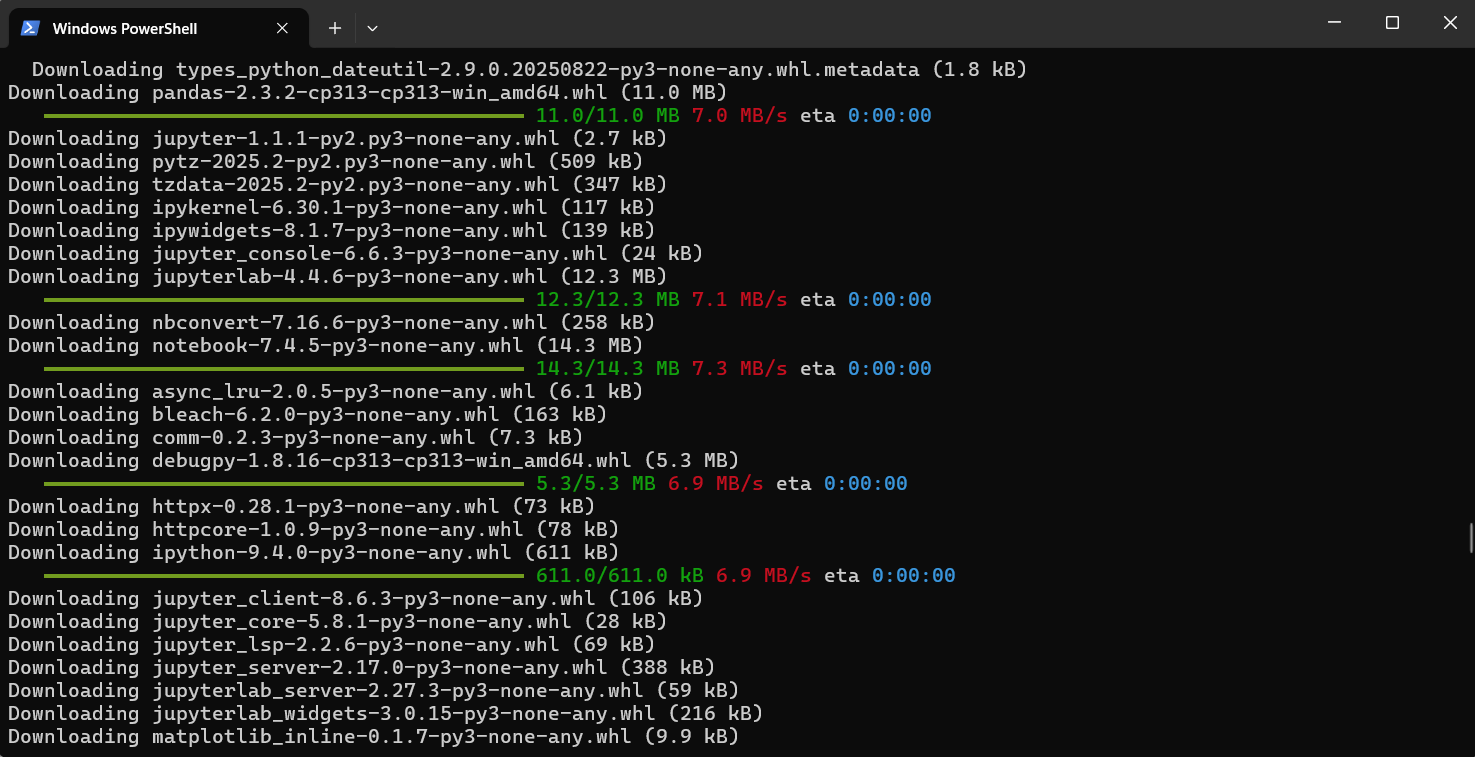
1. Install video processing

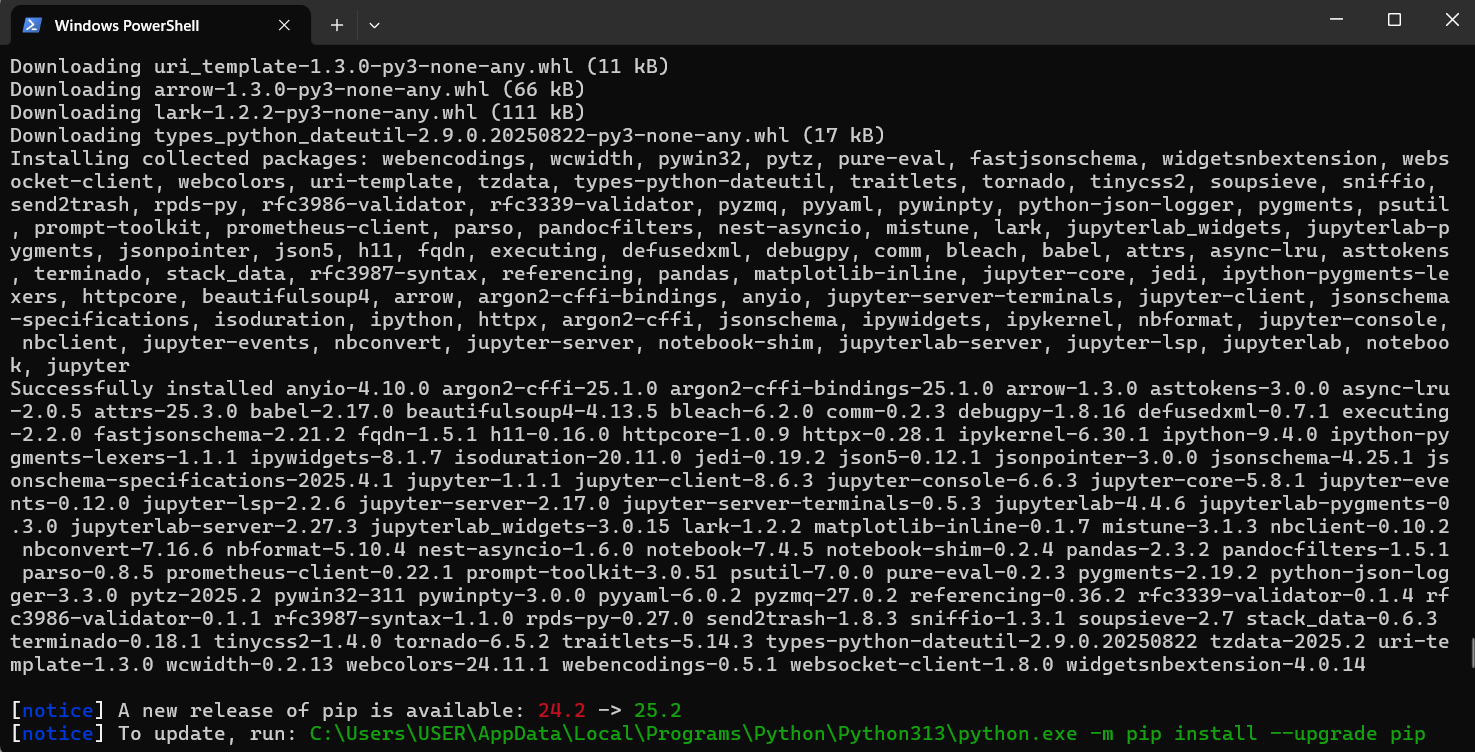


1. Install General Purpose









## Bagian 3: Verifikasi Instalasi

Buat file Python sederhana untuk menguji semua library yang telah diinstall: **Jalankan script dan dokumentasikan hasilnya:**

## Bagian 4: Simple Test dengan Sample Code

Buat dan jalankan contoh sederhana untuk setiap kategori multimedia:

**3.5.1 Test Audio Processing**

|  |
| --- |
| import numpy as np import matplotlib.pyplot as plt  # Generate simple sine wave duration = 2 # seconds sample\_rate = 44100 frequency = 440 # A4 note  t = np.linspace(0, duration, int(sample\_rate \* duration)) audio\_signal = np.sin(2 \* np.pi \* frequency \* t)  # Plot waveform plt.figure(figsize=(10, 4)) plt.plot(t[:1000], audio\_signal[:1000]) # Plot first 1000 samples plt.title('Sine Wave (440 Hz)') plt.xlabel('Time (s)') plt.ylabel('Amplitude') plt.grid(True) plt.savefig('sine\_wave\_test.png', dpi=150, bbox\_inches='tight') plt.show()  print(f"Generated {duration}s sine wave at {frequency}Hz") print(f"Sample rate: {sample\_rate}Hz") print(f"Total samples: {len(audio\_signal)}") |

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

Kode 8: Test audio processing sederhana

**3.5.2 Test Image Processing**

|  |
| --- |
| import numpy as np  import matplotlib.pyplot as plt from PIL import Image  # Create a simple test image width, height = 400, 300  image = np.zeros((height, width, 3), dtype=np.uint8)  # Add some patterns image[:, :width//3, 0] = 255 # Red section image[:, width//3:2\*width//3, 1] = 255 # Green section image[:, 2\*width//3:, 2] = 255 # Blue section  # Add a white circle in the center center\_x, center\_y = width//2, height//2 radius = 50  Y, X = np.ogrid[:height, :width]  mask = (X - center\_x)\*\*2 + (Y - center\_y)\*\*2 <= radius\*\*2 image[mask] = [255, 255, 255]  # Display and save plt.figure(figsize=(8, 6)) plt.imshow(image) plt.title('Test Image with RGB Stripes and White Circle') plt.axis('off') plt.savefig('test\_image.png', dpi=150, bbox\_inches='tight') |

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

|  |
| --- |
| plt.show()  print(f"Created test image: {width}x{height} pixels") print(f"Image shape: {image.shape}") print(f"Image dtype: {image.dtype}") |

27

28

29

30

31

Kode 9: Test image processing sederhana

**Dokumentasikan hasil eksekusi:**

* Screenshot output dari kedua script di atas
* Gambar yang dihasilkan (sine\_wave\_test.png dan test\_image.png)
* Error message jika ada dan cara mengatasinya

# Bagian Laporan

## Output Verifikasi Instalasi

**Copy-paste output lengkap dari script test\_multimedia.py di sini:**

|  |
| --- |
| [PASTE OUTPUT DI SINI] |

1

Kode 10: Output verifikasi instalasi

## Screenshot Hasil Test

**Sisipkan screenshot atau gambar hasil dari:**

* Terminal/command prompt yang menunjukkan environment aktif
* Output dari script test audio (sine wave plot)
* Output dari script test image (RGB stripes dengan circle)

*Gunakan perintah \includegraphics untuk menyisipkan gambar*

## Analisis dan Refleksi

**Jawab pertanyaan berikut:**

1. **Mengapa penting menggunakan environment terpisah untuk project multimedia?** *[Jawaban Anda di sini]*
2. **Apa perbedaan utama antara conda, venv, dan uv? Mengapa Anda memilih tool yang Anda gunakan?**

*[Jawaban Anda di sini]*

1. **Library mana yang paling sulit diinstall dan mengapa?**

*[Jawaban Anda di sini]*

1. **Bagaimana cara mengatasi masalah dependency conflict jika terjadi?** *[Jawaban Anda di sini]*
2. **Jelaskan fungsi dari masing-masing library yang berhasil Anda install!**

*[Jawaban Anda di sini]*

## Troubleshooting

**Dokumentasikan masalah yang Anda hadapi (jika ada) dan cara mengatasinya:**

* **Masalah 1:** *[Deskripsi masalah]*

**Solusi:** *[Cara mengatasi]*

* **Masalah 2:** *[Deskripsi masalah]*

**Solusi:** *[Cara mengatasi]*

# Export Environment untuk Reproduksi

Sebagai langkah terakhir, export environment Anda agar dapat direproduksi:

## Untuk Conda

|  |
| --- |
| conda env export > environment.yml |

1

Kode 11: Export conda environment

## Untuk venv/uv

|  |
| --- |
| pip freeze > requirements.txt |

1

Kode 12: Export pip requirements

**Copy-paste isi file environment.yml atau requirements.txt di sini:**

|  |
| --- |
| [PASTE ISI FILE DI SINI] |

1

Kode 13: Environment/Requirements file

# Kesimpulan

**Tuliskan kesimpulan Anda mengenai:**

* Pengalaman setup Python environment untuk multimedia
* Persiapan untuk project multimedia selanjutnya
* Saran untuk mahasiswa lain yang akan melakukan setup serupa

*[Kesimpulan Anda di sini]*

**7 Referensi**

Sertakan referensi yang Anda gunakan selama proses setup dan troubleshooting.

# References