TRƯỜNG ĐẠI HỌC ĐIỆN LỰC

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO CHUYÊN ĐỀ HỌC PHẦN**

**MÔN NGÔN NGỮ LẬP TRÌNH PYTHON**

**ĐỀ TÀI:**

**LẬP TRÌNH TRỢ LÝ ẢO FRIDAY TRONG IRON MAN VỚI PYTHON**

**Sinh viên thực hiện: TRỊNH NHẬT HUY**

**NGUYỄN TUẤN ANH**

**Giảng viên hướng dẫn: NGUYỄN THỊ HỒNG KHÁNH**

**Ngành: CÔNG NGHỆ THÔNG TIN**

**Chuyên ngành: CÔNG NGHỆ PHẦN MỀM**

**Lớp: D13CNPM6**

**Khóa: 2018 – 2023**

**PHIẾU CHẤM ĐIỂM**

Sinh viên thực hiện:

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và tên** | **Điểm** | **Chữ ký** |
| **1** | TRỊNH NHẬT HUY |  |  |
| **2** | NGUYỄN TUẤN ANH |  |  |

|  |  |  |
| --- | --- | --- |
| **Họ và tên giảng viên** | **Chữ ký** | **Ghi chú** |
| Giảng viên chấm 1: |  |  |
| Giảng viên chấm 2: |  |  |

**MỤC LỤC**

[**LỜI NÓI ĐẦU** 1](#_Toc87482029)

[**CHƯƠNG 1: TỔNG QUAN VỀ NGÔN NGỮ PYTHON** 2](#_Toc87482030)

[**1.1. Khái niệm** 2](#_Toc87482031)

[**1.2. Đặc điểm** 2](#_Toc87482032)

[**1.3. Ứng dụng Python trong thực tế** 4](#_Toc87482033)

[**1.3.1. Phân tích dữ liệu** 4](#_Toc87482034)

[**1.3.2. Khoa học dữ liệu** 4](#_Toc87482035)

[**1.3.3. Phát triển Website** 5](#_Toc87482036)

[**1.3.4. Phát triển trò chơi** 5](#_Toc87482037)

[**1.3.5. Phát triển phần mềm** 5](#_Toc87482038)

[**1.3.6. Kỹ thuật dữ liệu** 6](#_Toc87482039)

[**1.3.7. Kỹ thuật robot** 6](#_Toc87482040)

[**1.3.8. Tự động hóa** 7](#_Toc87482041)

[**1.3.9. Giao diện và điều khiển phần cứng** 7](#_Toc87482042)

[**1.3.10. Giáo dục và đào tạo** 7](#_Toc87482043)

[**CHƯƠNG 2: SỬ DỤNG VÀ ỨNG DỤNG CÁC THƯ VIỆN ĐỂ XÂY DỰNG TRỢ LÝ ẢO FRIDAY** 8](#_Toc87482044)

[**2.1. Pyttsx3** 8](#_Toc87482045)

[**2.1.1. Chạy thử** 8](#_Toc87482046)

[**2.1.2. Cài đặt nhịp điệu và âm lượng** 9](#_Toc87482047)

[**2.1.3. Giọng nói và ngôn ngữ** 9](#_Toc87482048)

[**2.2. PyAudio** 9](#_Toc87482049)

[**2.3. Speech recognition** 9](#_Toc87482050)

[**2.4. PyWhatKit** 11](#_Toc87482051)

[**2.5. Pyjokes** 11](#_Toc87482052)

[**2.6. Matplotlib** 11](#_Toc87482053)

[**CHƯƠNG 3: LẬP TRÌNH TRỢ LÝ ẢO FRIDAY** 13](#_Toc87482054)

[**3.1. Trợ lý ảo là gì?** 13](#_Toc87482055)

[**3.2. Các kiến thức thu được sau khi xây dựng và lập trình xong trợ lý ảo FRIDAY** 13](#_Toc87482056)

[**KẾT LUẬN** 16](#_Toc87482057)

[**TÀI LIỆU THAM KHẢO** 17](#_Toc87482058)

# **LỜI NÓI ĐẦU**

Trong cuộc sống của mỗi chúng ta khi nền kinh tế ngày càng phát triển, ngành công nghệ thông tin trở thành một trở thủ, điều không thể thiếu với hầu hết các ngành nghề. Có thể ban đầu nhiều người sẽ cảm thấy chưa cần thiết hoặc chưa sẵn sàng sử dụng nó vì nghĩ rằng phần mềm là một khái niệm gì đó rất mới mẻ, mới lạ, khó sử dụng.

Ngày nay trong cuộc sống 4.0 của chúng ta gần như không thể thiếu được những khái niệm liên quan đến công nghệ thông tin, từ các phương tiện truyền thông, xã hội cho đến lĩnh vực kinh doanh, quản lý… Mọi thứ đều cần những phần mềm cũng như sản phẩm của công nghệ thông tin để hỗ trợ giúp nâng cao phát triển vững mạnh hơn.

Cùng với sự phát triển của nhiều ngôn ngữ lập trình như PHP, Java thì Python là một cái tên đáng chú ý. Hiện nay ngôn ngữ Python là một ngôn ngữ được sử dụng rất phổ biến. Python là một ngôn ngữ có cấu trúc rõ ràng, thuận tiện cho người mới bắt đầu học lập trình. Cấu trúc của Python còn cho phép người sử dụng viết mã lệnh với số lần gõ phím tối thiểu nhất. Python là ngôn ngữ lập trình đơn giản nhưng lại rất hiệu quả. Bên cạnh đó, Python là ngôn ngữ có tính hướng đối tượng cao.

Trợ lý là một vai trò không thể thiếu trong công việc, giúp xử lý và điều hành tốt hơn, nhanh hơn, tiết kiệm thời gian. Song đôi khi chúng ta sẽ cần dùng đến công nghệ cao hơn, đó chính là trợ lý ảo, sử dụng trợ lý ảo từ AI sẽ tăng hiệu quả tốc độ làm việc lên gấp nhiều lần, độ chính xác cao và nhanh chóng. Vì vậy em chọn đề tài “Lập trình trợ lý ảo FRIDAY trong IRON MAN” với Python – một đề tài rất thú vị.

# **CHƯƠNG 1: TỔNG QUAN VỀ NGÔN NGỮ PYTHON**

## **1.1. Khái niệm**

Python là ngôn ngữ lập trình hướng đối tượng thông dụng dùng để viết các tiện ích hệ thống. Nó cũng được sử dụng như ngôn ngữ kết dính đóng vai trò tích hợp C và C++.

Được tạo ra bởi Guido van Rossum tại Amsterdam năm 1990, Python hoàn toàn tạo kiểu động và dùng cơ chế cấp phát bộ nhớ tự động. Python được phát triển trong một dự án mã mở, do tổ chức phi lợi nhuận Python Software Foundation quản lý.

Python là ngôn ngữ có hình thức khá đơn giản và rõ ràng, do đó tạo nên sự dễ dàng tiếp cânh cho những lập trình viên mới bắt đầu.

Ban đầu, Python được phát triển để chạy trên nền Unix, nhưng rồi theo thời gian, nó đã được mở rộng sang mọi hệ điều hành từ MS-DOS đến Mac OS, OS/2, Windows, Linux và các hệ điều hành khác thuộc họ Unix.

## **1.2. Đặc điểm**

* Cú pháp rất tường minh, dễ đọc.
* Các khả năng tự xét mạnh mẽ.
* Hướng đối tượng trực giác.
* Cách thể hiện tự nhiên mã thủ tục.
* Hoàn toàn mô-đun hóa, hỗ trợ các gói theo cấp bậc.
* Xử lý lỗi dựa theo ngoại lệ.
* Kiểu dữ liệu động ở mức rất cao.
* Các thư viện chuẩn và các mô-đun ngoài bao quát hầu như mọi việc.
* Phần mở rộng và mô-đun dễ dàng viết trong C, C++.
* Có thể nhúng trong ứng dụng như một giao diện kịch bản (scripting interface).
* Python mạnh mẽ và thực hiện rất nhanh.

- Python dễ dàng kết nối với các thành phần khác:

* Python có thể kết nối với các đối tượng COM, .NET (Ironpython, Python for .net), và CORBA, Java.. Python cũng được hỗ trợ bởi internet.
* Có thể viết các thư viện trên C/C++ để nhúng vào Python và ngược lại.

- Python là ngôn ngữ có khả năng chạy trên nhiều nền tảng:

* Python có cho mọi hệ điều hành: Windows, Linux/Unix, OS/2, Mac, Amiga, và những hệ điều hành khác. Thậm chí có cả những phiên bản chạy trên .NET, máy ảo Java, và điện thoại di động (Nokia Series 60). Với cùng một mã nguồn sẽ chạy giống nhau trên mọi nền tảng.

- Python rất đơn giản và dễ học:

* Python có cộng đồng lập trình rất lớn, hệ thống thư viện chuẩn, và cả các thư viện mã nguồn mở được chia sẻ trên mạng.

- Python là ngôn ngữ mã nguồn mở:

* Cài đặt Python dùng giấy phép nguồn mở nên được sử dụng và phân tối tự do, ngay cả trong việc thương mại. Giấy phép Python được quản lý bởi Python Software Foundation.

## **1.3. Ứng dụng Python trong thực tế**

### **1.3.1. Phân tích dữ liệu**

Khi các công ty trong mọi ngành thu thập ngày càng nhiều dữ liệu, họ cần những người có thể hiểu được dữ liệu đó. Và khi đó, các nhà phân tích dữ liệu có kỹ năng Python thực sự hữu ích.

Python phổ biến cho việc phân tích dữ liệu vì các thư viện mạnh mẽ như numpy và pandas, giúp các tác vụ phân tích và dọn dẹp dữ liệu tương đối đơn giản, ngay cả khi làm việc với các bộ dữ liệu lớn. Ngoài ra còn có các thư viện Python hỗ trợ nhiều tác vụ phân tích dữ liệu khác nhau, từ khai thác dữ liệu web với Beautiful Soup đến hiển thị dữ liệu bằng Matplotlib.

Các công cụ như Jupyter Notebook giúp các nhà phân tích dữ liệu dễ dàng tạo các phân tích dễ lặp lại hoặc thêm văn bản và hình ảnh để giúp công việc của họ dễ hiểu ngay cả với những người không chuyên về lập trình.

### **1.3.2. Khoa học dữ liệu**

Python cũng cực kỳ phổ biến cho các tác vụ liên quan đến dữ liệu nâng cao trong lĩnh vực máy học. Các thư viện mạnh mẽ như scikit-learning và TensorFlow giúp việc triển khai các thuật toán học máy phổ biến trở nên đơn giản. Đồng thời nhiều thư viện chuyên biệt giúp thực hiện nhiều tác vụ học máy từ nhận dạng hình ảnh đến tạo nội dung một cách dễ dàng

Hầu hết mọi thứ bạn thấy đang xãy ra xung quanh liên quan đến “trí tuệ nhận tạo” đều là một dạng triển khai máy học. Và rất nhiều chương trình học máy đang được thực hiện với Python.

### **1.3.3. Phát triển Website**

Rất nhiều công ty nổi tiếng hiện nay sử dụng Python là bằng chứng cho thấy Python là một ngôn ngữ rất phổ biến để phát triển ứng dụng web. Nhiều trang web bạn sử dụng hàng ngày được xây dựng bằng Python và các Python framework phổ biến như Django và Flask. Mặc dù bản thân các trang được hiển thị bằng HTML và CSS, Python làm nền tảng cho các yếu tố trực quan này trên nhiều trang web, điều khiển chức năng, quản lý cơ sở dữ liệu, tài khoản người dùng và hơn thế nữa.

### **1.3.4. Phát triển trò chơi**

Python được sử dụng trong việc phát triển các trò chơi điện tử độc lập, nhờ vào sự tồn tại của các thư viện tiện lợi như PyGame.

Python không được sử dụng thường xuyên trong việc phát triển các trò chơi phức tạp. Nếu mục tiêu của bạn là xây dựng một thế giới 3D chân thực, thì tốc độ tương đối chậm của Python và mức sử dụng bộ nhớ tương đối cao khiến nó không phải là ngôn ngữ lý tưởng cho trường hợp này. Tuy nhiên, Python đôi khi được sử dụng để xây dựng các hệ thống làm nền tảng cho các trò chơi này. Các trò chơi bao gồm Battlefield 2, Eve Online, The Sims 3, Civilization IV và World of Tanks sử dụng Python, mặc dù không có trò chơi nào được viết hoàn toàn bằng Python.

### **1.3.5. Phát triển phần mềm**

Python được sử dụng rộng rãi trong phát triển phần mềm, trên nhiều ứng dụng trong thực tế. Ngày nay ranh giới giữa phát triển phần mềm và phát triển web hơi mờ nhạt vì hầu như tất cả phần mềm đều được xây dựng để hoạt động trên web ngay cả khi có cả ứng dụng dành cho máy tính để bàn. Dropbox là một ví dụ điển hình về một công ty phát triển phần mềm hiện đại làm cả hai và Python được sử dụng để xây dựng ứng dụng dành cho máy tính để bàn của Dropbox. Tương tự, Spotify có cả ứng dụng web và máy tính để bàn và Python được sử dụng để xây dựng một số dịch vụ nền giúp chúng hoạt động.

Tất nhiên, Python cũng được sử dụng tại nhiều công ty để phát triển các phần mềm nội bộ.

### **1.3.6. Kỹ thuật dữ liệu**

Nhiều thư viện Python làm cho nó trở thành một lựa chọn hàng đầu cho các nhà phân tích dữ liệu và các nhà khoa học dữ liệu cũng làm cho Python trở thành một ngôn ngữ quan trọng đối với các kỹ sư dữ liệu. Các kỹ sư dữ liệu sử dụng Python cho các nhiệm vụ như xây dựng pipelines, kết hợp bộ dữ liệu, làm sạch dữ liệu, làm việc với API, tự động hóa các quy trình dữ liệu khác nhau, v.v.

### **1.3.7. Kỹ thuật robot**

Python là một ngôn ngữ phổ biến trong lĩnh vực chế tạo người máy, cả chuyên nghiệp và không chuyên. Với những người chỉ làm vì sở thích, Python thường được sử dụng cùng với nền tảng phần cứng Raspberry Pi, cho phép thử nghiệm linh hoạt và giá cả phải chăng. Trong kinh doanh, Python là một trong những ngôn ngữ thường được sử dụng để tự động hóa quy trình bằng rô bốt (robotic process automation – RPA) và nó được sử dụng để làm những việc như các cánh tay rô bốt công nghiệp được lập trình để có thể hoạt động song song với nhau.

### **1.3.8. Tự động hóa**

Python tuyệt vời để tự động hóa các tác vụ lặp đi lặp lại và có gần như vô tận các trường hợp sử dụng trong thực tế để tự động hóa Python. Ví dụ: Python là một công cụ phổ biến trong DevOps vì nó giúp tự động hóa các hệ thống và quy trình hiệu quả và minh bạch. Nhưng bên ngoài lĩnh vực phát triển phần mềm, nó cũng được sử dụng rộng rãi để tự động hóa mọi thứ, từ các hệ thống phức tạp đến các quy trình đơn giản, cá nhân như điền vào bảng tính hoặc trả lời email.

### **1.3.9. Giao diện và điều khiển phần cứng**

Khả năng điều khiển phần cứng của Python vượt ra ngoài lĩnh vực robotics. Trên thực tế, Python được sử dụng trong tất cả các loại ứng dụng điều khiển phần cứng trong thế giới thực. Ví dụ: thư viện Python có thể sử dụng Python cho nhiều ứng dụng điều khiển công nghiệp.

### **1.3.10. Giáo dục và đào tạo**

Bởi vì Python là một ngôn ngữ có thể đọc được (readable) rất cao và cũng có nhiều mục đích sử dụng thực tế khác nhau, nên Python là ngôn ngữ được lựa chọn đầu tiên cho những người muốn học lập trình. Nhiều hướng dẫn Python, video, các khóa học tương tác và các tài liệu giáo dục khác có sẵn cho Python khiến nó được cho là ngôn ngữ lập trình dễ học nhất.

# **CHƯƠNG 2: SỬ DỤNG VÀ ỨNG DỤNG CÁC THƯ VIỆN ĐỂ XÂY DỰNG TRỢ LÝ ẢO FRIDAY**

## **2.1. Pyttsx3**

- Là một thư viện chuyển đổi văn bản thành giọng nói bằng Python. Không giống như các thư viện thay thế, nó hoạt động ngoại tuyến và tương thích với cả Python 2 và 3.

**Cài đặt**

pip install pyttsx3

Nếu bạn nhận được các lỗi như  No module named win32com.client, No module named win32, hoặc No module named win32api, bạn sẽ cần phải cài đặt thêm pypiwin32 .

### **2.1.1. Chạy thử**

import pyttsx3

import os

engine = pyttsx3.init() #hay pyttsx3.init(driverName='sapi5')

engine.say("Good morning, How are you")

engine.say("Today is 12 February") engine.save\_to\_file("Hello" , "chao.mp3")

# Flush the say() queue and play the audio

engine.runAndWait()

os.system("chao.mp3")

### **2.1.2. Cài đặt nhịp điệu và âm lượng**

engine.setProperty('rate', 50)

engine.setProperty('volume', 0.25)

### **2.1.3. Giọng nói và ngôn ngữ**

- Vào Settings Time & Language Language cài đặt ngôn ngữ English, Vietnames…

- Vào Settings Time & Language Speech cài đặt giọng nói Voices Add Voices chọn Vietnamese

## **2.2. PyAudio**

- Cung cấp các liên kết Python cho PortAudio, thư viện I / O âm thanh đa nền tảng. Với - PyAudio, bạn có thể dễ dàng sử dụng Python để phát và ghi âm thanh trên nhiều nền tảng khác nhau. PyAudio được lấy cảm hứng từ:

* pyPortAudio / fastaudio: Các liên kết Python cho API PortAudio v18.
* tkSnack: bộ công cụ âm thanh đa nền tảng cho Tcl / Tk và Python.

## **2.3. Speech recognition**

**-** Là một quá trình nhận dạng mẫu, với mục đích là phân lớp (classify) thông tin đầu vào là tín hiệu tiếng nói thành một dãy tuần tự các mẫu đã được học trước đó và lưu trữ trong bộ nhớ. Các mẫu là các đơn vị nhận dạng, chúng có thể là các từ, hoặc các [âm vị](https://vi.wikipedia.org/wiki/%C3%82m_v%E1%BB%8B). Nếu các mẫu này là bất biến và không thay đổi thì công việc nhận dạng tiếng nói trở nên đơn giản bằng cách so sánh dữ liệu tiếng nói cần nhận dạng với các mẫu đã được học và lưu trữ trong bộ nhớ. Khó khăn cơ bản của nhận dạng tiếng nói đó là tiếng nói luôn biến thiên theo thời gian và có sự khác biệt lớn giữa tiếng nói của những người nói khác nhau, tốc độ nói, ngữ cảnh và môi trường âm học khác nhau. Xác định những thông tin biến thiên nào của tiếng nói là có ích và những thông tin nào là không có ích đối với nhận dạng tiếng nói là rất quan trọng. Đây là một nhiệm vụ rất khó khăn mà ngay cả với các kỹ thuật xác suất thống kê mạnh cũng khó khăn trong việc tổng quát hoá từ các mẫu tiếng nói những biến thiên quan trọng cần thiết trong nhận dạng tiếng nói.

- Các nghiên cứu về nhận dạng tiếng nói dựa trên ba nguyên tắc cơ bản:

* Tín hiệu tiếng nói được biểu diễn chính xác bởi các giá trị phổ trong một khung thời gian ngắn (short-term amplitude spectrum). Nhờ vậy ta có thể trích ra các đặc điểm tiếng nói từ những khoảng thời gian ngắn và dùng các đặc điểm này làm dữ liệu để nhận dạng tiếng nói.
* Nội dung của tiếng nói được biểu diễn dưới dạng chữ viết, là một dãy các ký hiệu ngữ âm. Do đó ý nghĩa của một phát âm được bảo toàn khi chúng ta phiên âm phát âm thành dãy các ký hiệu ngữ âm.
* Nhận dạng tiếng nói là một quá trình nhận thức. Thông tin về ngữ nghĩa (semantics) và suy đoán (pragmatics) có giá trị trong quá trình nhận dạng tiếng nói, nhất là khi thông tin về [âm học](https://vi.wikipedia.org/wiki/%C3%82m_h%E1%BB%8Dc) là không rõ ràng.

**Cài đặt**

Trước tiên, hãy đảm bảo rằng bạn có tất cả các yêu cầu được liệt kê trong phần "Yêu cầu".

Cách dễ nhất để cài đặt điều này là sử dụng pip install SpeechRecognition .

Nếu không, hãy tải xuống bản phân phối nguồn từ [PyPI](https://pypi.python.org/pypi/SpeechRecognition/) và giải nén kho lưu trữ.

Trong thư mục, hãy chạy install python setup.py .

**Yêu cầu**

Để sử dụng tất cả các chức năng của thư viện, bạn nên có:

* **Python** 2.6, 2.7 hoặc 3.3+ (bắt buộc)
* **PyAudio** 0.2.11+ (chỉ bắt buộc nếu bạn cần sử dụng đầu vào micrô, Micrô )
* **PocketSphinx** (bắt buộc nếu bạn cần phải sử dụng Sphinx recognizer, recognizer\_instance.recognize\_sphinx )
* **Thư viện ứng dụng Google API cho Python** (chỉ bắt buộc nếu bạn cần sử dụng API Google Cloud Speech, Recogzer\_instance.recognize\_google\_cloud )
* **Bộ mã hóa FLAC** (chỉ bắt buộc nếu hệ thống không phải là Windows / Linux / OS X dựa trên x86)

Các yêu cầu sau là tùy chọn, nhưng có thể cải thiện hoặc mở rộng chức năng trong một số trường hợp:

* Trên Python 2 và chỉ trên Python 2, một số hàm (như Recogzer\_instance.recognize\_bing ) sẽ chạy chậm hơn nếu bạn chưa cài đặt **Monotonic cho Python 2** .
* Nếu sử dụng CMU Sphinx, bạn có thể muốn [cài đặt các gói ngôn ngữ bổ sung](https://github.com/Uberi/speech_recognition/blob/master/reference/pocketsphinx.rst#installing-other-languages) để hỗ trợ các ngôn ngữ như tiếng Pháp quốc tế hoặc tiếng Quan Thoại.

## **2.4. PyWhatKit**

- Là một thư viện Python với nhiều tính năng hữu ích khác nhau. Đây là một thư viện dễ sử dụng mà không yêu cầu bạn thực hiện một số thiết lập bổ sung.

**Cài đặt**

Thư viện này có thể được cài đặt bằng lệnh pip, hãy mở dấu nhắc lệnh của bạn và nhập lệnh sau ...

pip3 install pywhatkit

## **2.5. Pyjokes**

**-** Một câu chuyện cười cho các lập trình viên (đùa như một dịch vụ).

Cài đặt

pip3 install pyjokes

hoặc

pip install pyjokes

## **2.6. Matplotlib**

- Một Matplotlib figure có thể được phân loại thành nhiều phần như dưới đây:

* **Figure:** Như một cái cửa sổ chứa tất cả những gì bạn sẽ vẽ trên đó.
* **Axes:** Thành phần chính của một figure là các axes (những khung nhỏ hơn để vẽ hình lên đó). Một figure có thể chứa một hoặc nhiều axes. Nói cách khác, figure chỉ là khung chứa, chính các axes mới thật sự là nơi các hình vẽ được vẽ lên.
* **Axis:** Chúng là dòng số giống như các đối tượng và đảm nhiệm việc tạo các giới hạn biểu đồ.
* **Artist:** Mọi thứ mà bạn có thể nhìn thấy trên figure là một artist như Text objects, Line2D objects, collection objects. Hầu hết các Artists được gắn với Axes.

- Để cài đặt Matplotlib nếu bạn có Anaconda chỉ cần gõ

conda install matplotlib

hoặc sử dụng tools pip

pip install matplotlib

# **CHƯƠNG 3: LẬP TRÌNH TRỢ LÝ ẢO FRIDAY**

## **3.1. Trợ lý ảo là gì?**

Trợ lý ảo (có thể được gọi là trợ lý kỹ thuật số, trợ lý giọng nói hay là trợ lý AI) là một ứng dụng lập trình hướng nhiệm vụ, nhận dạng giọng nói của con người và thực hiện các lệnh được phát âm bởi người dùng. Nền tảng của nó là AI và năng suất của nó dựa vào việc lưu trữ hàng triệu từ và hàng triệu cụm từ. Không giống như các thiết bị nhận dạng giọng nói đầu tiên mà các nhà khoa học đang nghiên cứu vào những năm 40-50 của thế kỷ trước, các trợ lý kỹ thuật số hiện đại không bị hạn chế bởi một mẫu ngôn ngữ hoặc từ vựng nhất định.

## **3.2. Các kiến thức thu được sau khi xây dựng và lập trình xong trợ lý ảo FRIDAY**

- Cài đặt thư viện ngoài (third-party module) trong Python với pip install

- Phân tích, xây dựng chương trình trợ lý ảo bằng Python

- Cách sử dụng google, youtube hiệu quả

- Xử lý ngôn ngữ tự nhiên - Natural Language Processing (NLP)

- Nhận dạng giọng nói - Speech Recognition

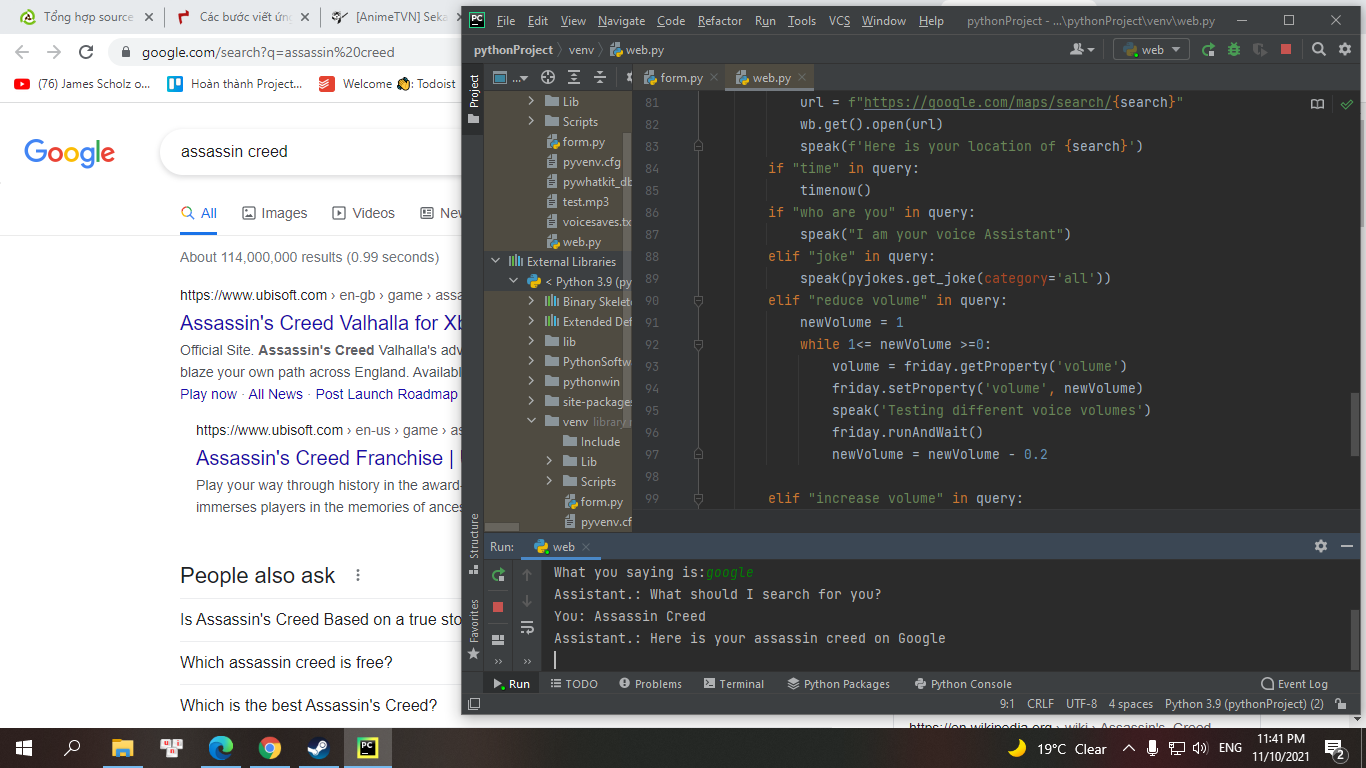
- Biến đổi thành giọng nói - Text to Speech

- Lấy ngày, thời gian hiện tại bằng thư viện Date Time

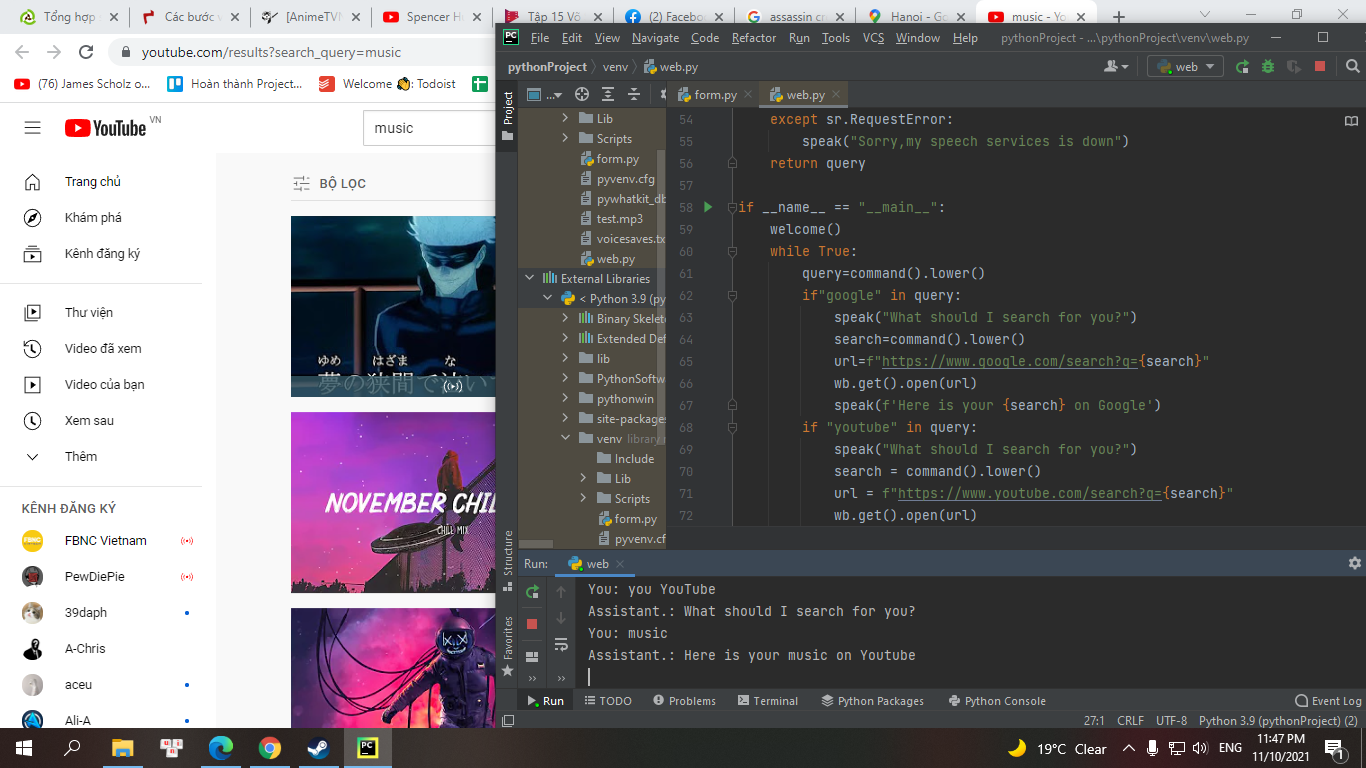
- Câu lệnh điều kiện - if else

- Xử lý lỗi - error handling, error catching với cấu trúc try catch

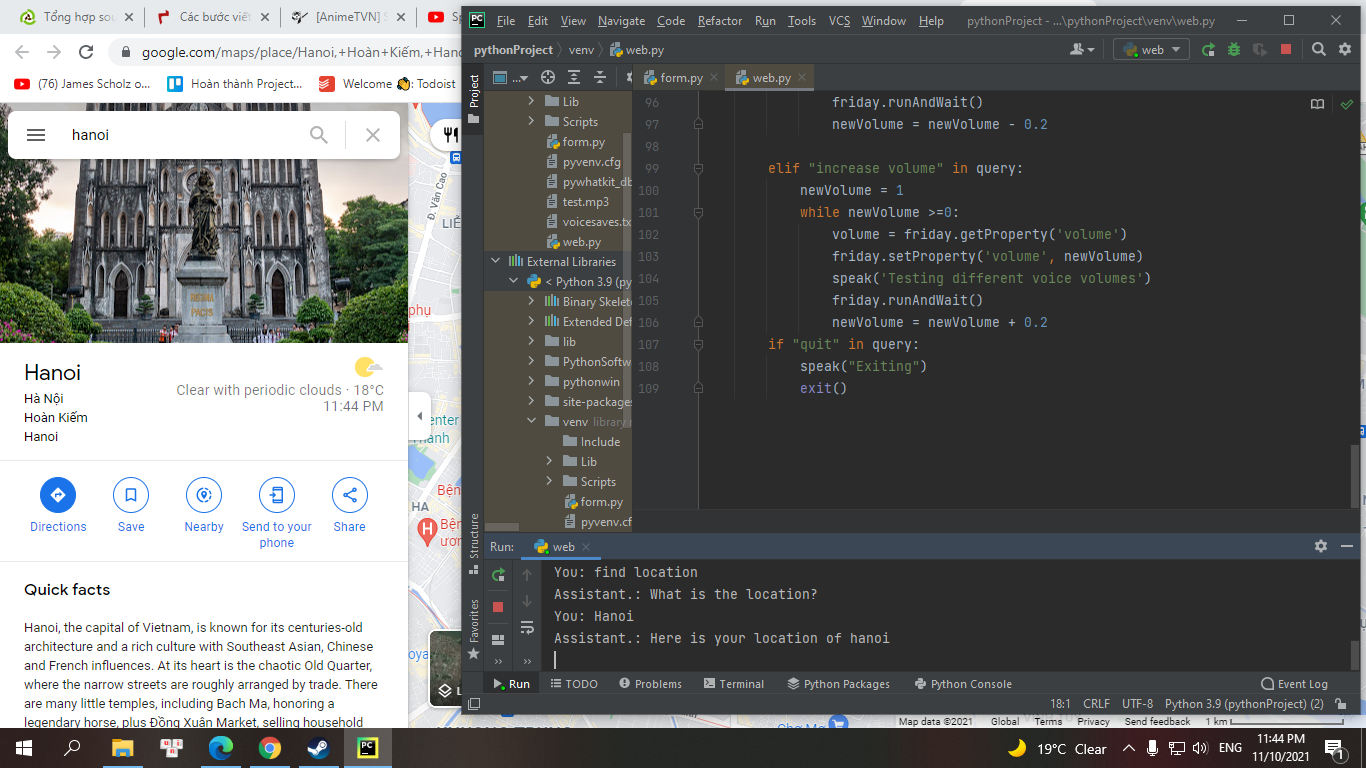
**3.3. Một số hình ảnh kết quả chạy chương trình**



*Hình 3.3.1: Thử tìm kiếm trên Google thông qua FRIDAY*



*Hình 3.3.2: Thử tìm kiếm âm nhạc thông qua FRIDAY*

**

*Hình 3.3.3: Thử tìm kiếm địa điểm thông qua FRIDAY*

# **KẾT LUẬN**

Sau thời gian nghiên cứu và bắt tay vào thực hiện, với mong muốn hiểu được bản chất và nắm rõ cách hoạt động của trợ lý ảo, chúng em đã hoàn thành đề tài “*Sử dụng ngôn ngữ Python để lập trình và xây dựng trợ lý ảo FRIDAY trong IRON MAN”.*

Kết quả đạt được:

* Hiểu và nắm rõ được cách hoạt động của trợ lý ảo
* Xây dựng được chương trình trợ lý ảo FRIDAY trong IRON MAN

Mặc dù đã rất cố gắng hoàn thiện báo cáo với tất cả sự nỗ lực, tuy nhiên do bước đầu tìm hiểu và xây dựng báo cáo môn học trong thời gian có hạn, với lượng kiến thức còn hạn chế, chúng em rất mong nhận được sự quan tâm, thông cảm và những đóng góp quý báu của các thầy cô và các bạn để báo cáo này ngày càng hoàn thiện hơn.

# **TÀI LIỆU THAM KHẢO**

- Lập Trình Trợ Lý Ảo FRIDAY trong Iron Man với Python bởi CodeXplore.