



Ciencia Latina
Internacional

Ciencia Latina Revista Científica Multidisciplinar, Ciudad de México, México.
ISSN 2707-2207 / ISSN 2707-2215 (en línea), julio-agosto 2024,
Volumen 8, Número 4.

https://doi.org/10.37811/cl_rcm.v8i4

IMPLEMENTACIÓN DE PYTHON EN LA FORMACIÓN DE INGENIEROS DEL ÁREA DE SISTEMAS Y COMPUTACIÓN EN EL TECN VILLAHERMOSA

**PROPOSAL FOR THE IMPLEMENTATION OF PYTHON IN THE
TRAINING OF COMPUTER SYSTEMS ENGINEERS AT TECN
VILLAHERMOSA**

Javier Enrique Gonzalez Jimenez

Instituto Tecnológico de Villahermosa, México

Jorge Cein Villanueva Guzmán

Instituto Tecnológico de Villahermosa, México

Ezequiel Gómez Domínguez

Instituto Tecnológico de Villahermosa, México

Jesús Junior Canales Obeso

Instituto Tecnológico de Villahermosa, México

Mario José Romellón Cerino

Instituto Tecnológico de Villahermosa, México

DOI: https://doi.org/10.37811/cl_rcm.v8i4.12878

Implementación de Python en la formación de Ingenieros del área de Sistemas y Computación en el TecNM Villahermosa

Javier Enrique Gonzalez Jimenez¹

114300942@villahermosa.tecnm.mx

<https://orcid.org/0009-0007-9909-6258>

Tecnológico Nacional de México / Instituto
Tecnológico de Villahermosa
México

Jorge Cein Villanueva Guzmán

jorge.vg@villahermosa.tecnm.mx

<https://orcid.org/0000-0003-1307-0801>

Tecnológico Nacional de México / Instituto
Tecnológico de Villahermosa
México

Ezequiel Gómez Domínguez

ezequiel.gd@zolmeca.tecnm.mx

<https://orcid.org/0009-0008-3996-951X>

Tecnológico Nacional de México / Instituto
Tecnológico de la Zona Olmeca
México

Jesús Junior Canales Obeso

jesus.canaleso@villahermosa.tecnm.mx

<https://orcid.org/0009-0008-2708-2567>

Tecnológico Nacional de México / Instituto
Tecnológico de Villahermosa
México

Mario José Romellón Cerino

mario.rc@villahermosa.tecnm.mx

<https://orcid.org/0000-0002-8579-1280>

Tecnológico Nacional de México / Instituto
Tecnológico de Villahermosa
México

RESUMEN

Este trabajo de investigación tiene como objetivo presentar una propuesta para la implementación de un lenguaje de programación multiparadigma en el Instituto Tecnológico de Villahermosa, con un enfoque particular en la adopción de Python. Se llevó a cabo una encuesta utilizando la metodología cuantitativa-descriptiva entre los estudiantes de las carreras pertenecientes al Departamento de Sistemas y Computación, específicamente en la asignatura "Fundamentos de Programación". El propósito de la encuesta fue determinar qué lenguaje de programación les gustaría aprender, adicional al que se enseña actualmente en la asignatura. Cabe destacar que la encuesta se realizó sin intervenciones externas que pudieran afectar los resultados proporcionados por los estudiantes. Tras analizar los datos de la encuesta, se observó que la mayoría de los encuestados expresaron su interés en aprender Python. Posteriormente, se elaboró una propuesta detallada para la implementación de Python como lenguaje de programación en el instituto como alternativa para la asignatura "Fundamentos de Programación".

Palabras clave: python, programación, sistemas computacionales, enseñanza

¹ Autor principal.

Correspondencia: jorge.vg@villahermosa.tecnm.mx

Proposal for the implementation of Python in the training of Computer Systems Engineers at TecNM Villahermosa

ABSTRACT

The objective of this research work is to present a proposal for the implementation of a multi-paradigm programming language at the Instituto Tecnológico de Villahermosa, with a particular focus on the adoption of Python. A survey was conducted using the quantitative-descriptive methodology among the students of the courses belonging to the Department of Systems and Computing, specifically in the subject "Programming Fundamentals". The purpose of the survey was to determine which programming language they would like to learn, in addition to the one currently taught in the subject. It should be noted that the survey was conducted without external interventions that could affect the results provided by the students. After analyzing the survey data, it was observed that most of the respondents expressed interest in learning Python. Subsequently, a detailed proposal was developed for the implementation of Python as a programming language at the institute as an alternative for the subject "Programming Fundamentals".

Keywords: python, programming, computer systems, teaching

Artículo recibido 20 agosto 2024

Aceptado para publicación: 05 septiembre 2024



INTRODUCCIÓN

La enseñanza de un lenguaje de programación en el ámbito laboral y educativo difiere significativamente en la forma en que se imparte en ambos sectores. En el área de programación existen diversos lenguajes con diferentes enfoques de uso: académico y profesional, en los cuales la facilidad de aprendizaje de cada uno de ellos es distinta. Sin embargo, algunos, aunque concebidos para el desarrollo de software, fueron diseñados teniendo en cuenta la facilidad de aprendizaje, permitiendo a cualquier persona incluso sin conocimientos previos en programación, utilizarlos gradualmente en menos tiempo, dependiendo del esfuerzo personal en aprender, a diferencia de otros lenguajes de programación.

La interrogante sobre cuál lenguaje de programación es apropiada para introducir a los estudiantes universitarios al mundo de la programación ha sido objeto de debate entre varios autores, quienes han buscado respuestas a través de investigaciones que se adapten a los requisitos específicos del momento y del lugar. Dos de los requisitos comúnmente considerados en este contexto son el nivel de dificultad para aprender el lenguaje de programación y la demanda en el mercado laboral para dicho lenguaje. Actualmente, la tendencia predominante en internet se centra en los datos. Empresas como Google, Facebook, Amazon, entre otras, buscan constantemente conocer tus preferencias para ofrecerte servicios, productos o incluso vender tus datos a terceros. Los datos pueden ser utilizados para llevar a cabo diversas acciones. Otra tendencia en alza es la proliferación de Inteligencias Artificiales (IA) de diversa índole, siendo la empresa OpenAI pionera en este ámbito con su chatbot llamado ChatGPT (OpenAI, s.f.). En este escenario, el lenguaje de programación que actualmente tiene amplio auge para el manejo de grandes cantidades de datos y destacarse en inteligencia artificial es Python.

Muchos líderes del mercado utilizan Python para diversas actividades, como segmentación de clientes, desarrollo web, automatización de procesos, desarrollo de aplicaciones de software y muchas más. Python es un lenguaje de programación poderoso (Mehare, Anilkumar, & Usmani, 2023).

Esta investigación es crucial para proporcionar pautas que justifiquen el por qué Python sería el lenguaje de programación más adecuado para impartir en el Instituto Tecnológico de Villahermosa, específicamente en la asignatura "Fundamentos de Programación" de las tres carreras del Departamento de Sistemas y Computación. Este trabajo de investigación proporcionará dichas pautas, destacando las razones para elegir a Python como lenguaje introductorio para los estudiantes.



MARCO REFERENCIAL

En el contexto de esta investigación, se llevó a cabo una revisión exhaustiva de artículos y estudios relacionados con la implementación de Python en la enseñanza de la programación en instituciones educativas. Uno de los destacados en este marco de referencia es el artículo titulado "Python Programming in an IS Curriculum: Perceived Relevance and Outcomes", elaborado por los profesores (Xu & Frydenberg, 2021) de Bentley University ubicada en Waltham, Massachusetts.

El estudio aborda la implementación de Python en los programas de estudios de Sistemas de Información (IS) y carreras de negocios. La motivación detrás de esta implementación radica en la creciente demanda de habilidades en análisis de datos, un área donde Python se destaca significativamente. La investigación incluyó la aplicación de encuestas a 64 estudiantes universitarios de diversas carreras relacionadas con los negocios que participaron en un curso de introducción a la programación.

Los resultados se analizaron mediante el método de análisis descriptivo, revelando que la mayoría de los estudiantes que recibieron instrucción en Python expresaron un interés significativo en aprender este lenguaje. Este interés se atribuyó a los beneficios que Python ofrece en el ámbito del análisis de datos, así como a las oportunidades laborales que esta habilidad podría generar. Sin embargo, es importante señalar que algunos estudiantes encontraron dificultades al programar en Python, citando la falta de familiaridad con la lógica de programación como un desafío.

En resumen, el estudio concluyó que Python podría considerarse un lenguaje alternativo valioso para ser utilizado en cursos introductorios de programación. Además, sugiere una implementación gradual de Python en los planes de estudio de Sistemas de Información, reconociendo tanto los beneficios potenciales como los posibles desafíos asociados con la enseñanza de este lenguaje.

Otro artículo que respalda esta línea de pensamiento es "C++ or Python? Which One to Begin with: A Learner's Perspective" por los autores (Ateeq, Habib, Umer, & Ul Rehman, 2014), proveniente del Departamento de Ciencias de la Computación e Informática de la Universidad Islámica de Bahawalpur, Pakistán. En este departamento, se ha enseñado el lenguaje de programación C++ como el primer lenguaje introductorio a los estudiantes. Sin embargo, han observado que los alumnos encuentran dificultades para aprender los conceptos básicos de la programación, y una de las razones identificadas es la complejidad inherente al lenguaje C++.



Por este motivo, investigaron un lenguaje alternativo que tuviera características como facilidad de aprendizaje, versatilidad, disponibilidad en todas las plataformas y enfoque no limitado a una sola área de la programación. Python fue identificado como el lenguaje que cumplía con estas características. Convencidos por estas ideas, llevaron a cabo un estudio en su departamento, organizaron en dos grupos paralelos.

En un grupo, se enseñó C++ a estudiantes que ya habían tenido experiencia previa con C++ y Python, mientras que el otro grupo no tenía conocimientos previos de ningún lenguaje y se les enseñó Python. Se aplicó una encuesta a 45 estudiantes dividida en dos partes: la primera, cuantitativa, se centró en las dificultades percibidas en diversas características de ambos lenguajes. La segunda parte consistió en preguntas abiertas que buscaban obtener información sobre el nivel de satisfacción y percepción del primer lenguaje.

Los resultados de la primera parte indicaron que el grupo que interactuó con Python sin conocimiento previo de programación tuvo más dificultades en ciertos aspectos, como algoritmos, funciones y listas/arrays, en comparación con el grupo que interactuó con C++.

Sin embargo, se justificó considerando que el grupo de Python no tenía conocimientos previos de programación. Además, en el resto de los aspectos, Python recibió menos votos de dificultad, y ambos grupos destacaron que es más fácil programar con Python en comparación con C++.

En la segunda parte, sobre la satisfacción de los alumnos, llegaron a la conclusión de que la práctica lleva a la perfección y, por ende, a la satisfacción. La perspectiva sobre el primer lenguaje resultó en la conclusión de que Python es un buen lenguaje introductorio. No obstante, reconocieron la importancia de C++ en su departamento debido a su arraigo y los buenos resultados que ha brindado. Concluyeron que es crucial elegir cuidadosamente el primer lenguaje introductorio para los alumnos y que Python es un buen candidato para este propósito, aunque no se debe pasar por alto a otros lenguajes de programación.

El artículo mexicano titulado "Python como primer lenguaje de programación" (Ramírez, 2010) proveniente del Tecnológico de Monterrey, Campus Estado de México, aborda la crucial decisión de seleccionar un lenguaje introductorio adecuado para los estudiantes universitarios. El autor comparte la perspectiva de que la elección del primer lenguaje de programación es esencial para el éxito de los



estudiantes en su aprendizaje.

Desde la sede en el Departamento de Tecnologías de Información y Computación de la División de Ingeniería y Arquitectura, el artículo destaca la complejidad de lenguajes tradicionales como Java, C, C++, y C#. Estos lenguajes, originalmente diseñados para el desarrollo de software, son descritos como detallados y menos amigables para la enseñanza. En contraste, Python se presenta como un lenguaje simplificado, diseñado no solo para el desarrollo de software, sino también para facilitar la comprensión y el uso, especialmente para principiantes en la programación.

El autor respalda su argumento mediante la comparación de ejercicios de programación en Python con aquellos en C y Java. Destaca la concisión de Python al realizar tareas similares con menos líneas de código, subrayando la simplicidad y la legibilidad del lenguaje. Además, señala la experiencia de otras universidades internacionales que han optado por Python como su primer lenguaje de programación, respaldando así la efectividad de esta elección.

A lo largo del artículo, se exploran las características distintivas de Python y se proporcionan ejemplos prácticos para ilustrar la ventaja de su enfoque simplificado. El autor también destaca la presencia de Python en la industria, enumerando empresas que utilizan activamente este lenguaje.

No obstante, el artículo no pasa por alto los posibles aspectos negativos de Python y aborda críticamente ciertos puntos. A pesar de ello, concluye que Python se presenta como una opción sólida y efectiva como primer lenguaje de programación introductorio para los estudiantes universitarios. La experiencia positiva del autor y la aceptación internacional respaldan la afirmación de que Python puede facilitar la introducción exitosa de los estudiantes al mundo de la programación.

El último trabajo que concuerda con nuestras ideas es el artículo "Python como primera aproximación a la programación", elaborado por el autor (Peña, 2015), desde la Universidad de Alcalá en la ubicación Alcalá de Henares, España . En este documento, se abordan los problemas que han afectado el rendimiento de los alumnos en su plan de estudios. El autor propone explorar un lenguaje alternativo que pueda modificar los resultados obtenidos previamente. Python se presenta como la elección adecuada para abordar estos desafíos.



METODOLOGÍA

La metodología seleccionada para este estudio fue la de investigación cuantitativa, la cual se enfoca en la recopilación de datos numéricos y su análisis estadístico para describir, explicar o predecir fenómenos (Álvarez, 2011). En este estudio, se trabaja con números, en concreto, para conocer los gustos de los estudiantes en el área de los lenguajes de programación.

En esta metodología, se recopilan y se analizan los datos censados, además de trabajar estrechamente con la investigación descriptiva, que tiene como objetivo principal describir de manera detallada las características, propiedades o condiciones de un fenómeno o situación, sin manipular variables ni establecer relaciones causales. En esta investigación, solo analizamos los datos, sin crear el porqué o el motivo de la elección de los estudiantes. Además, se trabajó en la recopilación de los datos mediante aplicación de encuestas que se trabajan tanto en la metodología cuantitativa como en la investigación descriptiva, sin manipular las variables ni sesgar los gustos de unos hacia los encuestados.

En el Instituto Tecnológico de Villahermosa (ITVH) se imparten al momento de la investigación las carreras de Ingeniería en Sistemas Computacionales, Ingeniería Informática e Ingeniería en Tecnología de la Información y Comunicación correspondiente al departamento de sistemas y computación. En la asignatura de Fundamento de Programación de estas carreras, se enseñan los fundamentos básicos para aprender a resolver problemas empleando la lógica en un lenguaje de programación.

Actualmente los docentes que imparten dicha asignatura utilizan como lenguaje principal Java por su versatilidad y la experiencia de estos en este lenguaje.

El lenguaje de programación Java ha sido una de las herramientas más poderosas a disposición de los programadores informáticos desde su creación en 1995. El lenguaje de programación Java sigue en su mayoría una sintaxis similar a la de C/C++ para escribir declaraciones y expresiones. (Samanta & Sarma, 2023).

Para identificar los intereses de los estudiantes, se diseñó y aplicó una encuesta a los alumnos de primer semestre que cursaban la asignatura de Fundamentos de Programación durante el periodo agosto – diciembre 2023 de las carreras de Ingeniería en Sistemas Computacionales, Ingeniería Informática e Ingeniería en Tecnología de la Información y Comunicación. La población encuestada fue de 75 estudiantes.



Dicha encuesta fue creada mediante Google Forms®, el cual permite crear y compartir fácilmente formularios online, y analiza las respuestas en tiempo real (Google LLC, s.f.). Este formulario se distribuyó a los estudiantes mediante medios electrónicos como código QR y mensajes de WhatsApp a través de los docentes titulares de dicha asignatura de las carreras antes mencionadas.

La encuesta fue la siguiente:

1.- ¿Te gusta programar?

- Si
- No

2.- ¿En dónde te quieres enfocar en la programación?

- Programación móvil
- Programación web
- Programación de videojuegos
- No me gusta programar

3.- ¿Que lenguaje de programación te están enseñando actualmente?

- | | | |
|--------------|--------------|-----------|
| • Java | • C++ | • Spring |
| • PHP | • TypeScript | Framework |
| • Python | • Laravel | • Flask |
| • JavaScript | • Django | • NodeJs |
| • C# | | • Angular |

4.- ¿Utilizas otro lenguaje de programación para programar?

- | | | |
|--------------|--------------|--------------------|
| • NO | • C# | • Spring Framework |
| • Java | • C++ | • Flask |
| • PHP | • TypeScript | • NodeJs |
| • Python | • Laravel | • Angular |
| • JavaScript | • Django | |

5.- ¿Con el lenguaje de programación seleccionado en la cuarta pregunta puedes programar una pequeña aplicación web?

- | | | |
|--|--------------|--------------------|
| • Todavía no tengo el conocimiento necesario | • C# | • Spring Framework |
| • Java | • C++ | • Flask |
| • PHP | • TypeScript | • NodeJs |
| • Python | • Laravel | • Angular |
| • JavaScript | • Django | |

6.- ¿Que otro lenguaje te gustaría aprender además de los que te enseñando en la escuela?

- | | | |
|--------------|--------------|--------------------|
| • Java | • C++ | • Spring Framework |
| • PHP | • TypeScript | • Flask |
| • Python | • Laravel | • NodeJs |
| • JavaScript | • Django | • Angular |
| • C# | | |

7.- ¿La curva de aprendizaje del lenguaje de programación que están enseñando actualmente es alta?

- SI: su curva de aprendizaje es difícil
- NO: su curva de aprendizaje es fácil



8.- ¿crees que llegarás a dominar ese lenguaje de programación que están enseñando actualmente?

- Si
- No

9.- ¿Deseas que te enseñen una variedad de lenguajes de programación?

- Si
- No

10.- ¿Te hubiera gustado haber comenzado con un lenguaje de programación con una curva de aprendizaje más fácil?

- Si
- No

RESULTADOS Y DISCUSIÓN

Derivado de la aplicación y análisis de los datos obtenidos mediante la encuesta a los estudiantes de la asignatura de Fundamento de programación, se identificó el interés de los jóvenes por aprender un lenguaje multiparadigma con una curva de aprendizaje más plana y que les permita acceder a mayores oportunidades laborales en el ámbito del desarrollo del software y sistemas embebidos.

Resultados de la encuesta

En las siguientes imágenes se presentan los resultados de la encuesta así como su interpretación.

Los resultados muestran que 91% (68) de los estudiantes tienen la idea de que les gusta programar, y el 9% (7) mencionan que no les gusta programar (Figura. 1); el 51% (38) tienen interés en enfocarse en la programación web, el 41% (31) programación de videojuegos, 4% (3) programación móvil, y, por último, un 4% (3) de los alumnos no les gusta programar (Figura. 2).

Figura. 1. Respuestas de la pregunta 1

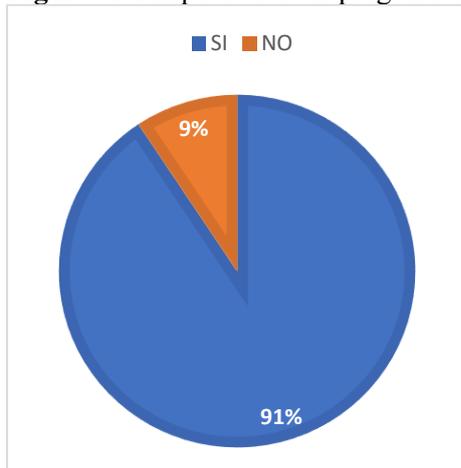
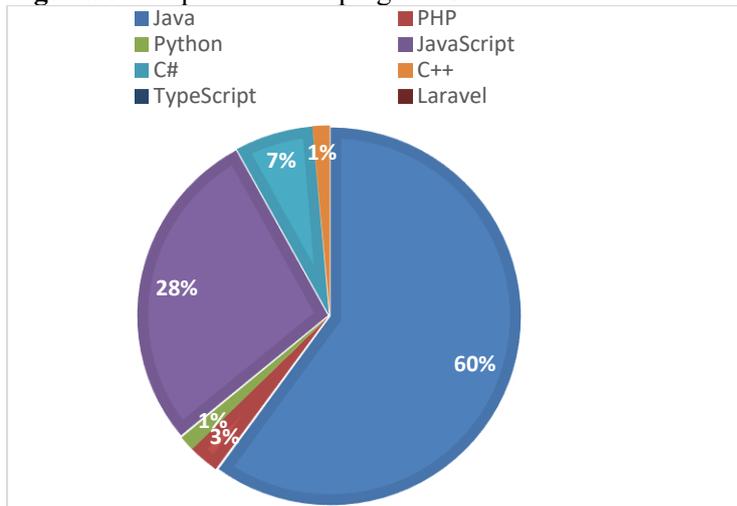


Figura. 2. Respuestas de la pregunta 2



En la Figura. 3, se muestra que un 60% (45) de los estudiantes recibe enseñanza del lenguaje de programación Java, 21% (28) aprenden JavaScript, mientras que un 7% (5) se educa en C#. Solo un 3% (2) recibe enseñanza de PHP, y un 1% (1) aprende Python. Por último, también un 1% (1) de los estudiantes se les imparte C++.

Figura. 3. Respuestas de la pregunta 3



El 52% (39) de los encuestados no muestran disposición para aprender o investigar otro lenguaje de programación además del que se les enseña en la asignatura, 16% (12) de los alumnos están estudiando o practicando por su cuenta el lenguaje de JavaScript, 9% (5) manifiestan inclinación por aprender el lenguaje de PHP, 7% (5) tienen interés en aprender C++, 6% (5) demuestra voluntad para aprender Java, 7% (5) tiene un grado de curiosidad por aprender Python y un 3% (2) manifiesta interés en aprender C# (**¡Error! No se encuentra el origen de la referencia.**).

En la Figura. 5, los resultados indican que el 68% (51) de los estudiantes carecen del conocimiento y habilidades para desarrollar una aplicación web básica. Un 15% (11) menciona poseer la capacidad de crearla utilizando JavaScript, mientras que el 11% (8) afirma tener las habilidades para desarrollar con Java.

Figura. 4. Respuestas de la pregunta 4

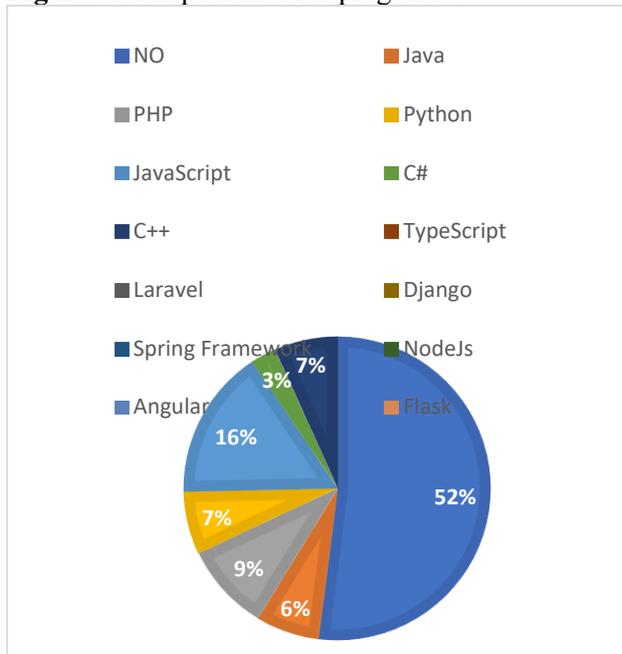
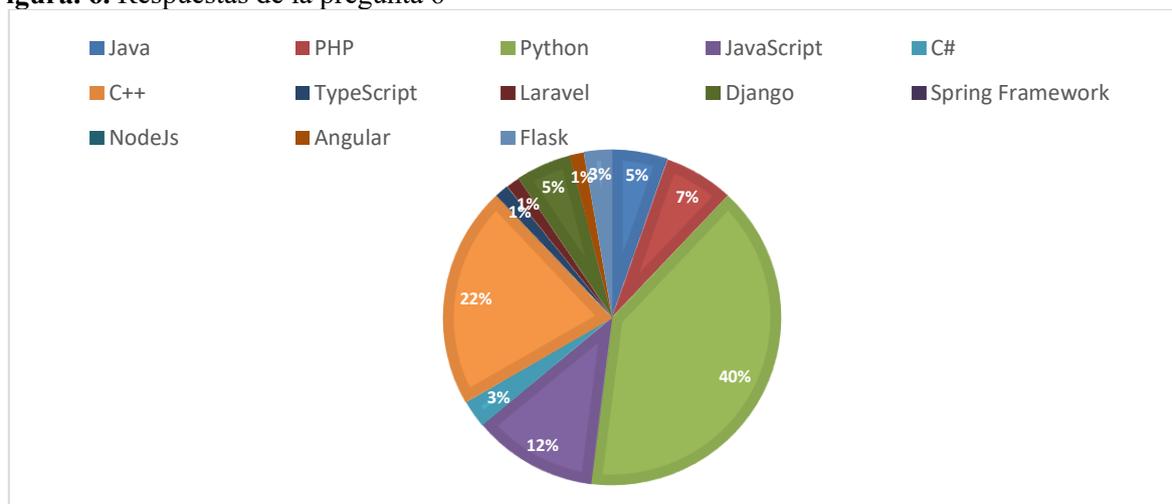


Figura. 5. Respuestas de la pregunta 5

Los resultados de la **Figura. 6** indican que un 48% de los estudiantes, desea aprender el lenguaje de programación Python o algunos de los frameworks relacionados a este.

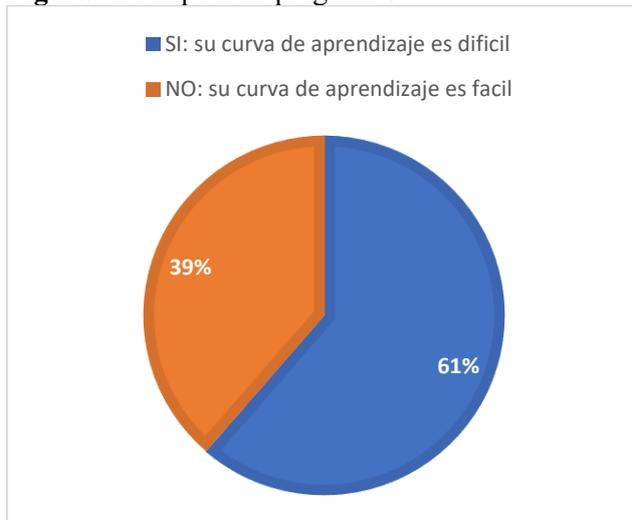
Figura. 6. Respuestas de la pregunta 6



En cuanto a la curva de aprendizaje, el 61% (46) encuentra difícil aprender el lenguaje que se le está impartiendo en la asignatura (actualmente Java), mientras que el 39% (29) de los alumnos no

experimenta dificultades para aprender el lenguaje de programación que se les imparte (Ver **Figura. 7**).

Figura. 7. Respuestas pregunta 7



En la pregunta 8, el 92% (69) de los estudiantes tiene la intención de conocer y manejar el lenguaje que se les imparte en la asignatura de "Fundamentos de Programación", mientras que el 8% (6) tiene la percepción de no dominar el lenguaje que se les está enseñando.

En relación a la enseñanza de varios lenguajes de programación por parte de la institución, el 96% (72) de los estudiantes aprecia la idea de que se les enseñen múltiples lenguajes, mientras que el 4% (3) no muestra interés en aprender varios lenguajes.

Y finalmente, en cuanto a la pregunta si les hubiera gustado iniciar programación con un lenguaje de programación con una curva de aprendizaje mas plana, el 72% (54) de los estudiantes aprecia la idea, mientras que el 28% (21) no está de acuerdo.

Se determina la importancia de incorporar la enseñanza de un lenguaje de programación alterno al que actualmente se está empleado en la asignatura de fundamento de programación con visión a estandarizar su uso en las asignaturas subsecuentes de plan de estudios de cada uno de las carreras del departamento de sistemas y computación. La elección de dicho lenguaje se hizo considerando su curva de aprendizaje, oportunidad laboral y versatilidad para aplicarse en distintos proyectos del desarrollo de software, aplicaciones móviles y sistemas embebidos. Por lo anterior, se propone incorporar el lenguaje multiparadigma Python dado que actualmente está posicionado a nivel mundial y nacional entre los más utilizados por la industria del desarrollo de software debido a su versatilidad en diferentes campos del desarrollo tecnológico y una curva de aprendizaje más plana que otros lenguajes.

Las fuentes consultadas que se tomaron como referencia para determinar, la factibilidad de aprender Python por su utilización en el mundo y posteriormente en México son las siguientes:

Índice TIOBE: mide la influencia y la utilización de los lenguajes de programación más utilizado del mundo. Obtienen sus datos mediante motores de búsqueda. (<https://www.tiobe.com/tiobe-index/>) (Ver

Tabla 1)

Tabla 1. Lenguaje de programación más usado del mundo por (TIOBE, 2023).

Octubre 2023	Lenguaje de programación	% utilización
1	Python	14,82%
2	C	12,08%
3	C++	10,67%
4	Java	8,92%
5	C#	7,71%

Stack Overflow: es un sitio web de preguntas y respuestas donde se postean sus dudas o problemas de programación. Cada año realizan una encuesta a sus usuarios para conocer qué tecnologías utilizan. (<https://survey.stackoverflow.co/2023/#technology>) (Ver *Tabla 2*)

Tabla 2. Lenguaje de programación más usado del mundo por Stack Overflow (Developer Survey, 2023)

Mayo 2023	Lenguaje de programación	% utilización
1	JavaScript	63.61%
2	HTML/CSS	52.97%
3	Python	49.28%
4	SQL	48.66%
5	TypeScript	38.87%

Devjobsscanner: portal de trabajo enfocado en el ámbito de Tecnologías de la Información (TI). Sus resultados lo obtienen analizando las ofertas de trabajo en todo el mundo que requiera conocimiento de un lenguaje. (<https://www.devjobsscanner.com/blog/top-8-most-demanded-programming-languages/>).

(Ver *Tabla 3*)

Tabla 3. Lenguaje de programación con más oferta del mundo por devjobsscanner (dev, 2023)

De ene/2022- may/2023	Lenguaje de programación	trabajos
1	Javascript/TypeScript	915k jobs
2	Python	603k jobs
3	Java	546k jobs
4	C#	375k Jobs
5	PHP	288k jobs



Hireline: es un portal de búsqueda de empleo enfocada a la TI, que realiza encuestas periódicas a sus usuarios en México para conocer las tecnologías que utilizan. (<https://hireline.io/mx/estudio-mercado-laboral-y-empleos-de-ti-mexico?year=2022#slide-skills-tecnologias-mas-demandadas>) (Ver **Tabla 4**)

Tabla 4. Lenguaje de programación más utilizados en México por Hireline (Olvera, 2022)

2022	Lenguaje de programación	% utilización
1	Java	15,22%
2	.Net	5,42%
3	Python	4,12%
4	Microsoft Azure	3,84%
5	AWS	3,68%

Coderslink: es un portal de trabajo enfocado en TI que recopila sus datos analizando los perfiles de un millón de profesionales de la industria tecnológica de Mexico y genera un Ranking de cuales son las tecnologías mejor remuneradas del país. (<https://coderslink.com/talento/blog/frameworks-lenguajes-de-programacion-mejor-pagados-2022/>) (Ver **Tabla 5**)

Tabla 5. Lenguaje de programación con Mejor sueldo en México por Coderslink (HRJ, 2022).

2022	Lenguaje de programación	Sueldos
1	HTML/CSS	\$63,911
2	C	\$48,247
3	C#/.NET	\$48,111
4	Python	\$47,230
5	SQL	\$45,952

Para implementar el lenguaje de programación en la asignatura de Fundamentos de Programación, es necesario determinar qué versión de Python se debe aprender. En este contexto, se recomienda utilizar la versión 3.12 de Python, lanzada en diciembre de 2023 o superior. La elección de Python 3.12 se basa en su continua evolución, mejoras en rendimiento, características adicionales y el soporte activo de la comunidad de desarrollo.

Ademas, se proponen a continuación temas de aprendizaje del lenguaje Python apegados al temario vigente de dicha asignatura (Ver **Tabla 6**)

Tabla 6. Temario de la asignatura Fundamentos de programación (*Tecnológico Nacional de México, 2023*)

No	Temas de la asignatura de Fundamento de programación	Temas recomendados de Python
1.	<p>Tema: Introducción a la Programación</p> <p>Subtema:</p> <p>1.1 Conceptos básicos.</p> <p>1.2 Características del lenguaje de programación.</p> <p>1.3 Estructura básica de un programa.</p> <p>1.4 Elementos del lenguaje: tipos de datos, literales, constantes, variables, identificadores, parámetros, operadores y salida de datos.</p> <p>1.5 Traducción de un programa: compilación, enlace, ejecución y errores.</p>	<p>Tema: Introducción a la programación en Python</p> <p>Subtema: Conceptos básicos</p> <p>En este apartado, se abordarán los fundamentos esenciales de la programación utilizando Python. Los estudiantes aprenderán sobre la sintaxis básica, la estructura de un programa en Python y la ejecución de instrucciones simples.</p> <p>Subtema: Características del lenguaje de programación</p> <p>Se explorarán las características distintivas de Python que lo hacen único y versátil. Se destacarán aspectos como su legibilidad, simplicidad y capacidad para abordar una amplia gama de aplicaciones.</p> <p>Subtema: Estructura básica de un programa</p> <p>Este subtema se centrará en la estructura fundamental que debe seguir un programa en Python. Se analizará la importancia de la indentación, la definición de funciones y la organización lógica del código.</p> <p>Subtema: Elementos del lenguaje</p> <p>Se estudiarán los elementos esenciales de Python, como tipos de datos (enteros, cadenas, listas, etc.), literales, constantes, variables, identificadores, parámetros de funciones, operadores y la salida de datos mediante instrucciones de impresión.</p> <p>Subtema: Traducción de un programa</p> <p>Este segmento se enfocará en el proceso de traducción de un programa Python, incluyendo la compilación, enlace, ejecución y la gestión de errores. Se discutirán las peculiaridades del entorno de ejecución de Python.</p>
2..	<p>Tema: Control de Flujo</p> <p>Subtemas:</p> <p>2.1 Estructuras secuenciales.</p> <p>2.2 Estructuras selectivas: simple, doble y múltiple.</p> <p>2.3 Estructuras iterativas: repetir mientras, hasta, desde.</p>	<p>Tema: Control de Flujo en Python</p> <p>Subtema: Estructuras secuenciales</p> <p>En este subtema, se explorará cómo Python gestiona las estructuras secuenciales de código. Los estudiantes aprenderán a ejecutar instrucciones en secuencia y comprenderán la importancia de la estructura de un programa en Python.</p> <p>Subtema: Estructuras selectivas: simple, doble y múltiple</p> <p>Se abordarán las estructuras selectivas en Python, incluyendo las instrucciones if, if-else y if-elif-else. Los estudiantes comprenderán cómo implementar decisiones en el código, adaptándose a las condiciones específicas.</p> <p>Subtema: Estructuras iterativas: repetir mientras, hasta, desde</p> <p>Este segmento se enfocará en las estructuras iterativas de Python, como los bucles while y for. Los estudiantes aprenderán a implementar repeticiones basadas en condiciones y a iterar sobre secuencias de datos utilizando estas estructuras.</p>

<p>3. Tema: Organización de datos</p> <p>Subtemas:</p> <p>3.1 Arreglos</p> <p>3.2 Unidimensionales: conceptos básicos, operaciones y aplicaciones.</p> <p>3.3 Multidimensionales: conceptos básicos, operaciones y aplicaciones.</p> <p>3.4 Estructuras o registros.</p>	<p>Tema: Organización de datos en python</p> <p>Subtema: Arreglos</p> <p>En este subtema, se explorará cómo Python maneja los arreglos, estructuras de datos fundamentales para organizar información. Se abordarán los conceptos básicos de los arreglos, las operaciones disponibles y sus diversas aplicaciones en el contexto de Python.</p> <p>Subtema: Arreglos Unidimensionales: conceptos básicos, operaciones y aplicaciones</p> <p>Los estudiantes aprenderán sobre los arreglos unidimensionales en Python, comprendiendo los fundamentos, las operaciones disponibles y cómo aplicar eficientemente esta estructura de datos en distintos contextos de programación.</p> <p>Subtema: Arreglos Multidimensionales: conceptos básicos, operaciones y aplicaciones</p> <p>Este segmento se enfocará en los arreglos multidimensionales en Python. Se explorarán los conceptos esenciales, las operaciones específicas y las diversas aplicaciones de estas estructuras, proporcionando a los estudiantes herramientas para manipular datos más complejos y organizados.</p> <p>Subtema: Estructuras o registros</p> <p>En este subtema, se introducirán las estructuras de registros en Python. Los estudiantes aprenderán cómo organizar y acceder a datos mediante esta estructura, que permite agrupar diferentes tipos de información bajo una única entidad. Declaración, operaciones, membresía, eliminación y conversión con listas.</p> <p>Aplicaciones prácticas y eficiencia.</p>
<p>4. Tema: Modularidad</p> <p>Subtema:</p> <p>4.1 Declaración y uso de módulos.</p> <p>4.2 Paso de parámetros o argumentos.</p> <p>4.3 Implementación.</p>	<p>Tema: Modularidad en Python</p> <p>Subtema: Declaración y Uso de Módulos</p> <p>En este subtema, se debe explorar cómo Python facilita el modularidad, permitiendo la creación y uso de módulos para organizar y reutilizar código. Se deben abordar las mejores prácticas para declarar módulos y cómo importarlos en otros scripts.</p> <p>Subtema: Paso de Parámetros o Argumentos</p> <p>Los estudiantes se sumergirán en la flexibilidad de Python en la comunicación entre funciones y módulos mediante el paso de parámetros y argumentos. Se abordarán las estrategias para definir funciones con parámetros y cómo adaptarlas a diferentes contextos.</p> <p>Subtema: Implementación Modular</p> <p>En este segmento, se deben enfocar en la implementación efectiva de funciones dentro de módulos. Explorar estrategias para organizar funciones y variables de manera clara y eficiente, fomentando una implementación modular coherente.</p>

Actividades de aprendizaje sugeridas para python

Para que los estudiantes pongan en práctica los conocimientos teóricos adquiridos, se propone que



desarrollen actividades de aprendizaje en las cuales puedan plasmar su creatividad y habilidades para la resolución de problemas mediante la implementación de Python.

Tabla 7. Actividades sugeridas por tema.

1.Introducción a la programación	
Actividades de Aprendizaje	Prácticas
<ul style="list-style-type: none"> • Participar en una discusión sobre la importancia de los primeros programas en el aprendizaje. • Realizar ejercicios prácticos para imprimir mensajes con diferentes variables. • Participar en discusiones sobre las características distintivas de Python. • Realizar ejercicios prácticos para aplicar conceptos de legibilidad y simplicidad en el código. • Practicar la organización del código mediante ejercicios de reestructuración. • Desarrollar programas simples con funciones definidas. • Realizar prácticas de manipulación de tipos de datos mediante ejercicios. • Desarrollar programas que utilicen diferentes elementos del lenguaje. • Participar en discusiones sobre errores comunes y su resolución. • Resolver problemas prácticos con errores y discutir las soluciones. 	<p>Práctica de Conceptos Básicos: "Saludo Personalizado" Descripción: Crear un programa que solicite el nombre y la edad del usuario, luego imprima un mensaje de bienvenida.</p> <p>Práctica de Características del Lenguaje: "Python en Foco" Descripción: Investigar y presentar en clase tres características clave de Python que destaquen su legibilidad y simplicidad.</p> <p>Práctica de Estructura Básica de un Programa: "Organización de Código" Descripción: Reorganizar un código desordenado utilizando indentación y definición de funciones, enfocándose en mejorar la legibilidad.</p> <p>Práctica de Elementos del Lenguaje: "Operaciones con Datos" Descripción: Crear un programa que solicite dos números al usuario, realice operaciones básicas y presente los resultados de manera clara.</p> <p>Práctica de Traducción de un Programa: "Corrección de Errores" Descripción: Identificar y corregir errores en un programa proporcionado, explicando el proceso de compilación y ejecución.</p>
2.Control de Flujo	
Actividades de Aprendizaje	Prácticas
<ul style="list-style-type: none"> • Abordar la importancia de los primeros programas en el aprendizaje. • Compartir experiencias y reflexionar sobre cómo los primeros pasos impactan en el desarrollo de habilidades de programación. • Realizar ejercicios prácticos para imprimir mensajes utilizando diferentes variables. • Fomentar la experimentación con diferentes tipos de datos y formatos de salida. • Explorar temas como la legibilidad del código, la sintaxis clara y la versatilidad del lenguaje. • Realizar ejercicios prácticos para aplicar conceptos de legibilidad y simplicidad en el código. • Practicar la organización del código mediante ejercicios de reestructuración. 	<p>Práctica de Estructuras Secuenciales: "Registro de Actividades" Descripción: Crear un programa que registre y muestre en pantalla una secuencia de actividades. Cada actividad debe ejecutarse en orden, demostrando el concepto de estructuras secuenciales.</p> <p>Práctica de Estructuras Selectivas: "Tomando Decisiones" Descripción: Desarrollar un programa que tome decisiones basadas en la entrada del usuario. Se utilizarán estructuras selectivas simples, dobles y múltiples para adaptarse a diferentes situaciones</p> <p>Práctica de Estructuras Iterativas: "Explorando Bucles" Descripción: Crear un programa que utilice tanto bucles while como for para iterar sobre</p>



- Guiar a los estudiantes en la creación de programas simples que involucren funciones definidas.
- Realizar ejercicios prácticos de manipulación de tipos de datos.
- Practicar la conversión entre tipos y operaciones básicas con diferentes tipos de datos.

una lista de elementos. Esto permitirá a los estudiantes practicar repeticiones basadas en condiciones y la iteración sobre secuencias de datos.

3.Organización de datos

Actividades de aprendizaje	Prácticas
<ul style="list-style-type: none"> • Explicar los conceptos básicos de los arreglos unidimensionales. • Describir las operaciones disponibles, como la creación, acceso a elementos y modificación. • Solicitar a los estudiantes crear un arreglo unidimensional que represente una lista de calificaciones. • Realizar operaciones simples, como mostrar la calificación promedio y la calificación más alta. • Compartir experiencias y desafíos encontrados durante la práctica. • Destacar la eficiencia y simplicidad de los arreglos unidimensionales en comparación con otras estructuras. • Presentar conceptos esenciales de los arreglos multidimensionales. • Proporcionar ejemplos prácticos de matrices bidimensionales, como una matriz que representa un tablero de ajedrez. • Implementar operaciones, como encontrar el elemento máximo en una fila o columna. • Introducir la declaración de estructuras de registros en Python. • Explicar operaciones como agregar, acceder y eliminar elementos de una estructura de registro. • Asignar la tarea de crear una estructura de registro que almacene información sobre estudiantes (nombre, edad, calificaciones, etc.). • Realizar operaciones como agregar nuevos registros, acceder a información específica y eliminar registros. • Discutir las ventajas y desventajas de utilizar estructuras de registros en comparación con otras formas de organizar datos. • Reflexionar sobre situaciones específicas en las que las estructuras de registros son más eficientes. 	<p>Práctica de Estructuras Secuenciales: "Registro de Actividades" Descripción: Crear un programa que registre y muestre en pantalla una secuencia de actividades. Cada actividad debe ejecutarse en orden, demostrando el concepto de estructuras secuenciales.</p> <p>Práctica de Estructuras Selectivas: "Tomando Decisiones" Descripción: Desarrollar un programa que tome decisiones basadas en la entrada del usuario. Se utilizarán estructuras selectivas simples, dobles y múltiples para adaptarse a diferentes situaciones.</p> <p>Práctica de Estructuras Iterativas: "Explorando Bucles" Descripción: Crear un programa que utilice tanto bucles while como for para iterar sobre una lista de elementos. Esto permitirá a los estudiantes practicar repeticiones basadas en condiciones y la iteración sobre secuencias de datos.</p>



4.Modularidad

Actividades de aprendizaje	Practicas
<ul style="list-style-type: none">• Crear un módulo simple que contenga funciones básicas (por ejemplo, una función que imprima un mensaje).• Explicar las mejores prácticas para la declaración de módulos.• Crear un script independiente que importe y utilice el módulo creado.• Demostrar cómo acceder a las funciones del módulo y utilizarlas en el nuevo script.• Sugerir posibles mejoras en la estructura del módulo para una mayor claridad y reutilización.• Crear funciones dentro de un módulo que acepten parámetros.• Mostrar ejemplos de diferentes tipos de parámetros (posicionales, por palabra clave, valores predeterminados).• Crear un script que importe el módulo y utilice las funciones con diferentes configuraciones de parámetros.• Demostrar la versatilidad del paso de argumentos.• Crear un módulo que contenga varias funciones y variables relacionadas.• Aplicar estrategias de organización clara, como agrupar funciones relacionadas y utilizar nombres significativos.• Simular la implementación del módulo en un proyecto más grande.• Realizar una revisión del módulo y sugerir mejoras para una implementación modular coherente y eficiente.	<p>Práctica de Modularidad: "Creación y Uso Básico de Módulos" Descripción: Familiarizarse con la creación y uso de módulos en Python.</p> <p>Práctica de Modularidad: "Paso de Parámetros en Funciones y Módulos" Descripción: Experimentar con el paso de parámetros y argumentos entre funciones y módulos.</p> <p>Práctica de Modularidad: "Implementación Modular Eficiente" Descripción: Enfocarse en la organización efectiva de funciones y variables dentro de un módulo.</p>

Material de apoyo educativo sugeridos de acceso libre

1. Libros:

- Aprenda a Pensar Como un Programador con Python (Downey, Elkner, & Meyers, 2022).
- Aprende Python (Delgado Quintero, 2022)
- Python para todos (González Duque).

2. Cursos de plataforma de aprendizaje:

- Curso gratuito proporcionado por la plataforma de aprendizaje Edutin Academy "Curso de Python" por el autor Gianluigi García: <https://edutin.com/curso-de-python>
- Curso gratuito proporcionado por la plataforma de aprendizaje UdeMy "Introducción a la Programación con Python" por el autor Juan Pablo de la Torre Valdez <https://www.udemy.com/course/introduccion-a-la-programacion-con-python-juan-de-la-torre/>



CONCLUSIONES

El lenguaje de programación Python es ampliamente reconocido en todo el mundo por su influencia y sus contribuciones significativas al ámbito de la informática. Su prominencia es tan notable que se ha adoptado como lenguaje introductorio en numerosas universidades de todo el mundo. Las discusiones en el ámbito educativo sobre cuál es el lenguaje de programación más adecuado para que los estudiantes lo aprendan primero son comunes y Python destaca como una opción líder. Las características únicas de Python, así como su facilidad de aprendizaje en comparación con otros lenguajes en el mercado, lo convierten en una elección atractiva; por este motivo fue uno de los más seleccionado por los estudiantes en la encuesta aplicada tanto como resultado la propuesta de implantación de su enseñanza

La propuesta de implementación se adaptó al temario de Fundamentos de Programación ofrecido por el Instituto Tecnológico de Villahermosa, garantizando una aplicación adecuada de los temas y proporcionando un aprendizaje secuencial para los estudiantes de dicha asignatura.

Aunque los lenguajes que se imparten en el Instituto Tecnológico de Villahermosa en la asignatura de Fundamentos de Programación son considerados apropiados y altamente demandados en el mundo laboral, no debemos pasar por alto la importancia de brindarle a Python una oportunidad en este contexto educativo. La inclusión de Python como lenguaje alternativo a los que son impartidos actualmente, enriquecerá la experiencia de aprendizaje de los estudiantes y prepararlos aún más para los desafíos del mundo profesional.

REFERENCIAS BIBLIOGRÁFICAS

- Álvarez, C. A. (2011). *METODOLOGIA DE LA INVESTIGACION CUANTITATIVA Y CUALITATIVA Guia didactica*. UNIVERSIDAD SURCOLOMBIANA. Obtenido de <https://www.uv.mx/rmipe/files/2017/02/Guia-didactica-metodologia-de-la-investigacion.pdf>
- Ateeq, M., Habib, H., Umer, A., & Ul Rehman, M. (abril de 2014). C++ or Python? Which One to Begin with: A Learner's Perspective. *Conference: 2014 International Conference on Teaching and Learning in Computing and Engineering (LaTiCE)*, 64-69. Obtenido de ResearchGate: https://www.researchgate.net/publication/271425337_C_or_Python_Which_One_to_Begin_with_A_Learner's_Perspective#:~:text=Results%20indicate%20that%20due%20to,language%20compared%20to%20C%2B%2B.



- Delgado Quintero, S. (2022). *Aprende Python*. GNU General Public License v3.0. Obtenido de <https://uneweb.edu.ve/tuto-docs/libro-python.pdf>
- dev, L. (21 de Junio de 2023). *Top 8 Most Demanded Programming Languages in 2023*. Obtenido de devjobsscanner: <https://www.devjobsscanner.com/blog/top-8-most-demanded-programming-languages/>
- Developer Survey*. (Mayo de 2023). Obtenido de Stack Overflow: <https://survey.stackoverflow.co/2023/#technology>
- Downey, A., Elkner, J., & Meyers, C. (2022). *Aprenda a Pensar Como un Programador con Python*. Wellesley, Massachusetts: Green Tea Press. Obtenido de <https://argentinaenpython.com/quiero-aprender-python/aprenda-a-pensar-como-un-programador-con-python.pdf>
- González Duque, R. (s.f.). *Python para todos*. Creative Commons. Obtenido de <http://edge.launchpad.net/improve-python-spanish-doc/0.4/0.4.0/+download/Python%20para%20todos.pdf>
- Google LLC. (s.f.). *Google Forms*. Obtenido de Google: <https://www.google.com/forms/about/>
- HRJ. (31 de Mayo de 2022). *Tecnologías más rentables para programadores: lenguajes de programación mejor pagados en 2022 (México)*. Obtenido de CodersLink: <https://coderslink.com/talento/blog/frameworks-lenguajes-de-programacion-mejor-pagados-2022/>
- Mehare, H. B., Anilkumar, J. P., & Usmani, N. A. (2023). The Python Programming Language. En *A Guide to Applied Machine Learning for Biologists* (págs. 27-60). Springer International Publishing.
- Olvera, E. (2022). *Reporte del Mercado*. Obtenido de Hireline: <https://hireline.io/mx/estudio-mercado-laboral-y-empleos-de-ti-mexico?year=2022#slide-skills-tecnologias-mas-demandadas>
- OpenAI. (s.f.). *ChatGPT For Everyone*. Obtenido de OpenAI: <https://openai.com/chatgpt/>
- Peña, R. (2015). Python como primera aproximación a la programación. 17-29. Obtenido de <https://aenui.org/revision/>
- Ramírez, A. O. (2010). Python como primer lenguaje de programación. 1-16. Obtenido de https://arielortiz.info/apps/s201711/tc1014/actividad_primer_lenguaje/



Samanta, D., & Sarma, M. (2023). Java Language Basics. En *Joy with Java* (págs. 41-145). Cambridge University Press.

Tecnológico Nacional de México. (Agosto de 2023). *Oferta educativa: Ingeniería en Sistemas Computacionales*. Obtenido de TNM - Instituto Tecnológico de villahermosa: <https://villahermosa.tecnm.mx/docs/oferta/ingsistemas/temario2010/1semestre/FundamentosdeProgramacion-AED-1285.pdf>

TIOBE. (Octubre de 2023). *TIOBE Index for October 2023*. Obtenido de TIOBE: <https://www.tiobe.com/tiobe-index/>

Xu, J., & Frydenberg, M. (Aug de 2021). Python Programming in an IS Curriculum: Perceived Relevance and Outcomes. *nformation Systems Education Journal*, v19 n4, 37-54. Obtenido de <https://eric.ed.gov/?id=EJ1310052>

