

UNIVERSIDAD DE LAS FUERZAS  
ARMADAS  
ESPE



**HERRAMIENTAS DE SOFTWARE  
PARA LA INGENIERIA**

**Maquina dispensadora con lenguaje  
python**

INGENIERÍA EN ELECTRÓNICA Y TELECOMINICACIONES

**Docente:** ING. Darwin Omar Alulema Flores

**Nombre:** Jhony Borja, Kevin Casagallo, David Campoverde

**NRC:** 5035

SANGOLQUI 10 DE JUNIO DEL 2019



## 1 Introducción

Al hablar de programación nos referimos al proceso por el cual podemos diseñar, codificar, limpiar y proteger el código fuente de programas computacionales. Donde se tiene el objetivo de que lo que fue creado se pueda desarrollar en una computadora de manera directa o a través de otros programas creados previamente. En los últimos años el rol de la programación en nuestro mundo se ha hecho vital para los procesos de sistematización de tareas y el manejo de información.

Python es un lenguaje de programación de uso general, versátil y popular. Es excelente como primer lenguaje porque es conciso y fácil de leer, y también es un buen lenguaje para tener muy en cuenta, ya que puede usarse para todo, desde desarrollo web hasta aplicaciones de desarrollo de software y ciencia de datos.

Python es un lenguaje de programación que permite varios estilos como programación orientada a objetos, programación imperativa y programación funcional. Otros paradigmas están soportados mediante el uso de extensiones.





## 2 Índice

1. Planteamiento del problema
2. Objetivos
3. Estado del arte
4. Marco teórico
  - (a) Definición
  - (b) Clases
  - (c) Herencias
5. Diagramas
6. Lista de componentes
7. Mapa de variables
8. Explicación del código fuente
9. Descripción de prerequisites y configuraciones
10. Conclusiones
11. Recomendaciones
12. Bibliografía
13. Anexos
14. Manual de usuario
15. Hojas técnicas





### 3 Planteamiento del problema

En la actualidad aprender un lenguaje de programación tiene tanta importancia en el campo laboral como aprender un nuevo idioma, y el saber programar no sólo es de útil para ingenieros de Sistemas o Técnicos en Computación e Informática. También puede llegar a ser para cualquier persona que se interese en programar.

Los lenguajes de programación más conocidos C, C ++ o Java, son los lenguajes más llamativos para la mayoría de personas que inician a programar y los escogen como el primer lenguaje de aprendizaje, que son efectivos para diseñar aplicaciones reales y, por lo tanto, son populares en la industria. Sin embargo, la compleja sintaxis de estos idiomas es un reto para los principiantes, lo que se convierte en un obstáculo para el aprendizaje de los novatos. Python tiene una sintaxis más simple y estructuras de datos de alto nivel para habilitar programas sucintos. Los múltiples paradigmas de Python también brindan oportunidades para que los principiantes aprendan varias características de los lenguajes de programación. Por lo tanto, Python se convierte gradualmente en una nueva opción del primer idioma para el aprendizaje.







## 4 Objetivos

El objetivo general del proyecto es desarrollar una programa con las características de una maquina expendedora enfocada a mejorar el aprendizaje de conceptos teóricos y prácticos en el marco educativo. También se presenta como objetivo inherente al proyecto investigar la versatilidad del lenguaje de programación Python. A continuación se presentan los objetivos específicos que se pretenden alcanzar:

- Observar y estudiar el funcionamiento del lenguaje python en el desarrollo de este proyecto
- Investigar sobre los diferentes tipos de datos, condicionales, clases, funciones y generadores de python que permiten la realización de este proyecto.
- Conocer las diferentes formas en las cuales implementar el lenguaje python.





## 5 Estado del arte

Bilina, R., Lawford, S. (2012). Python for Unified Research in Econometrics and Statistics. *Econometric Reviews*, 31(5), 558–591. <https://doi.org/10.1080/07474938.2011.553573>

Python es un potente lenguaje de programación de código abierto de alto nivel, que está disponible para múltiples plataformas. Es compatible con la programación orientada a objetos, y recientemente se ha convertido en una seria alternativa a los lenguajes compilados de bajo nivel como C. Es fácil de aprender y usar, y es reconocido por tiempos de desarrollo muy rápidos, lo que lo hace adecuado para la creación rápida de prototipos de software, así como fines docentes. Motivamos el uso de Python y sus módulos de extensión gratuitos para aplicaciones independientes de alto rendimiento en econometría y estadísticas, y como una herramienta para unir diferentes aplicaciones. Brindamos detalles sobre las funciones básicas del idioma, que permitirán al usuario comenzar a trabajar de inmediato y luego brindarán ejemplos prácticos de los usos avanzados de Python

Misra, A., Blank, D., y Kumar, D. (2009). Un contexto musical para la enseñanza de la computación introductoria. *ACM SIGCSE Bulletin*, 41 (3), 248. <https://doi.org/10.1145/1595496.1562955>

Describimos myro.chuck, un módulo de Python para controlar la síntesis de música, y sus aplicaciones para la enseñanza de introducción a la informática. El módulo se creó dentro del marco Myro usando el lenguaje de programación ChucK, y se usó en un curso introductorio de ciencias de la computación que combina robots, gráficos y música. Los resultados apoyaron el valor de la música al involucrar a los estudiantes y ampliar su visión de la ciencia de la computación.

Neidhardt, A., Klein, F., Knoop, N., y Köllmer, T. (2017). Herramienta flexible de Python para aplicaciones dinámicas de síntesis binaural. En la 142a Convención de la Sociedad de Ingeniería de Audio (pp. 1–5).

En este informe, presentamos una herramienta de código abierto para la síntesis binaural dinámica en tiempo real implementada en Python en la parte superior de pyAudio. El núcleo es una implementación eficiente de la convolución dividida uniformemente con el enfoque de superposición-guardar. La reproducción dinámica e interactiva de escenas de audio espacial se ha convertido en un requisito común en la ciencia y la industria. Los casos de uso son diversos, desde exámenes de audición que consideran la rotación de cabezales hasta escenarios de reproducción complejos para realidad aumentada o virtual, por ejemplo, en combinación con pantallas montadas en cabezales. Con Python como un lenguaje de programación flexible y fácil de aprender, pyBinSim ofrece un gran valor en la investigación y la enseñanza de la síntesis binaural. El código fuente, ejemplos y documentación están disponibles en línea.

Marowka, A. (2018). Aceleradores Python para computación de alto rendimiento. *Diario de Supercomputación*, 74 (4), 1449-1460. <https://doi.org/10.1007/s11227-017-2213-5>

Python se convirtió en el lenguaje preferido para la enseñanza en el mundo académico, y es uno de los lenguajes de programación más populares para la computación científica. Esta gran popularidad se produce a pesar del débil desempeño del lenguaje. Esta debilidad es la motivación



que impulsa los esfuerzos dedicados por la comunidad de Python para mejorar el rendimiento del idioma. En este artículo, estamos siguiendo estos esfuerzos mientras nos centramos en una solución prometida específica que tiene como objetivo proporcionar un alto rendimiento y portabilidad de rendimiento para las aplicaciones de Python.

Ortiz, A. (2012). Desarrollo web con python y django (solo resumen) (p. 686). Asociación de Maquinaria de Computación (ACM). <https://doi.org/10.1145/2157136.2157353>

Muchos instructores ya han descubierto la alegría de enseñar programación utilizando el lenguaje de programación Python. Ahora es el momento de llevar a Python al siguiente nivel. Este taller presentará Django, un marco web de Python de código abierto que le ahorra tiempo y hace que el desarrollo web sea divertido. Está dirigido a instructores de ciencias de la computación que desean enseñar cómo construir aplicaciones web elegantes con un mínimo esfuerzo. Django sigue el patrón arquitectónico Modelo-Vista-Controlador (MVC). Su objetivo es facilitar la creación de sitios web complejos, basados en bases de datos. Django hace hincapié en la reutilización y la conectividad de los componentes, el desarrollo rápido y el principio de DRY (no se repita). Python se utiliza en todo, incluso para configuraciones, archivos y modelos de datos. Los temas que se tratarán durante el taller incluyen: configuración y configuración, lenguaje de plantilla e integración de bases de datos a través de mapeo objeto-relacional. Los participantes deben estar familiarizados con Python, HTML y SQL. Se requiere computadora portátil.



## 6 Merco teorico

### 6.1 Definicion

Python es un lenguaje de programación interpretado cuya sintaxis favorece a código legible. Y define este como un lenguaje multiparadigma, Suponiendo paradigma como modelo de desarrollo que cada lenguaje tiene. Debido a que soporta orientación a objetos, programación imperativa y en menor medida programación funcional. Es interpretado de tipado dinámico y multiplataforma.

- Interpretado debido a que el intérprete va a traducir nuestro código a medida que lo necesitamos
- Es de tipado dinámico porque permite la transformación de variables. Windows, Linux, Mac, Android, Web
- Multiplataforma porque se puede ejecutar en cualquier sistema operativo como

#### 6.1.1 Clases

Las clases proveen una forma de empaquetar datos y funcionalidad juntos. Al crear una nueva clase, se crea un nuevo tipo de objeto, permitiendo crear nuevas instancias de ese tipo. Cada instancia de clase puede tener atributos adjuntos para mantener su estado. Las instancias de clase también pueden tener métodos (definidos por su clase) para modificar su estado.

Las clases de Python proveen todas las características normales de la Programación Orientada a Objetos el mecanismo de la herencia de clases permite múltiples clases base, una clase derivada puede sobre escribir cualquier método de su clase base. Los objetos pueden tener una cantidad arbitraria de datos de cualquier tipo. Igual que con los módulos, las clases participan de la naturaleza dinámica de Python, se crean en tiempo de ejecución, y pueden modificarse luego de la creación.





## 7 Diagramas

## 8 Lista de componentes

- Computador
- Progrma python
- Acceso a internet si no desea descargar el programa se lo puede ejecutar de forma online
- 







## 9 Mapa de variables

## 10 Explicacion del codigo fuente

```
main.py  saved
1  class Product:
2      def __init__(self, _name, _price):
3          self.name = _name
4          self.price = _price
5
6      def __repr__(self):
7          return "{0} : {1}".format(self.name, self.price)
8
9
10 products = {
11     1: Product("Papas", 0.50),
12     2: Product("Galletas", 0.45),
13     3: Product("Agua", 1.35),
14     4: Product("Sanduche", 1.65),
15     5: Product("V220", 0.65)
16 }
17
18 coins = {
19     "1": 10,
20     "50": 5,
21     "25": 10,
22     "10": 10,
23     "5": 5
24 }
25
26 # coin_translator = {
27 #     "5": 5,
28 #     "10": 10,
29 #     "25": 25,
30 #     "50": 50,
31 #     "1": 100
32 # }
```



```
main.py  saved
33
34
35 class Machine:
36
37     def __init__(self, coins, products):
38         self.coins = coins
39         self.products = products
40         self.coin_translator = coin_translator = {
41             "5": 5,
42             "10": 10,
43             "25": 25,
44             "50": 50,
45             "1": 100
46         }
47
48     def display_products(self):
49         for key, value in self.products.items():
50             print("{0}. {1}".format(key, value))
51
52     def calculate_return(self, price, payed):
53         returned = payed-price
54         returned_coins = []
55         for coin, coin_ammount in self.coins.items():
56             coin_value = self.coin_translator[coin]
57             if returned < coin_value:
58                 continue
59             elif returned == coin_value and coin_ammount > 0:
60                 returned_coins.append(coin)
61                 self.coins[coin] = self.coins[coin]-1
62                 return returned_coins
63             elif coin_ammount > 0:
```



```
main.py  saved
62         return returned_coins
63     elif coin_ammount > 0:
64         while returned > 0 and coin_ammount > 0:
65             returned -= coin_value
66             if returned < 0:
67                 returned += coin_value
68                 break
69             self.coins[coin] = self.coins[coin]-1
70             returned_coins.append(coin)
71             if returned == 0:
72                 return returned_coins
73
74         print("Maquina no tiene cambio disponible")
75         return []
76
77
78 if __name__ == "__main__":
79
80     machine = Machine(coins, products)
81     while True:
82         machine.display_products()
83         option = input("-> Elija un producto, q para salir: ")
84         if option == "q":
85             break
86         if option.isdigit():
87             option = int(option)
88         else:
89             break
90         if option not in machine.products:
91             print("Producto no existe")
92         else:
93             print("Producto elegido: {0} : {1}".format(
```



```
80 machine = Machine(coins, products)
81 while True:
82     machine.display_products()
83     option = input("-> Elija un producto, q para salir: ")
84     if option == "q":
85         break
86     if option.isdigit():
87         option = int(option)
88     else:
89         break
90     if option not in machine.products:
91         print("Producto no existe")
92     else:
93         print("Producto elegido: {0} : {1}".format(
94             machine.products[option].name, machine.products
95             [option].price))
96         payment = input("Ingrese su pago separado por
97         comas -> ")
98         payment = payment.split(",")
99         for coin in payment:
100             if coin not in machine.coins:
101                 print("No se ha ingresado un pago valido")
102                 pass
103             machine.coins[coin] = machine.coins[coin]+1
104         ammount = sum([machine.coin_translator[x] for x in
105             payment])
106         product_price = machine.products[option].price*100
107         if ammount == product_price:
108             input("Muchas gracias por su compra ... ")
109         elif ammount > product_price:
110             change = machine.calculate_return
111             (product_price, ammount)
```



## 11 Descripción de prerequisites y configuraciones

- Descargar e instalar python
- También se lo puede hacer funcionar de forma online ingresando en la pagina <https://repl.it/languages>





## 12 Aportaciones

## 13 Conclusiones

Python es una buena opción como lenguaje como lenguaje de programación para comenzar que además muy versátil y con multitud de aplicaciones.

Es muy utilizado para crear aplicaciones web e incluso para desarrollar algunos juegos.

Python, utilizan un intérprete el cual puede ejecutar programas de forma instantánea sin la necesidad de un compilador.







## 14 Recomendaciones

Aprende los conceptos básicos del lenguaje dominar estos conceptos desde un comienzo facilitará resolver problemas en nuestro código y Asia lograr que sea eficiente.

Cambiando el texto puedes aprender la forma en la que el programa maneja los datos básicos.

Examina la sintaxis de forma en la que se escribe el lenguaje para que el compilador pueda entenderlo.

Comenta todo tu código, esto permite dejar una explicación corta, pero clara, acerca de lo que hace el código.





## 15 Cronograma

## 16 Bibliografía

<http://docs.python.org.ar/tutorial/3/classes.html>

<http://do1.dr-chuck.net/py4inf/ES-es/book.pdf>

<https://www.digitaltechinstitute.com/phyton-lenguaje-programacion/>





## 17 Anexos

## 18 Manual de usuario

- Al iniciar el programa
- Se despliega un listado con los productos enumerada, con el nombre de cada producto y su respectivo valor
- Para seleccionar un producto solo se escribe el numero del producto que se desea
- luego aparece un mensaje confirmando la elección del producto y un mensaje ingrese su pago
- Para pagar el producto ingresamos separado por comas. teniendo en cuenta que 1 representa un dolar, 50 representa 0.50 centavos, 25 representa 0.25 centavos, 10 representa 0.10 centavos y 5 representa 0.05 centavos
- Luego la maquina nos dirá cual es nuestro cambio, la cantidad de monedas y la denominación de cada una, • luego la maquina regresa al listado inicial en el cual escoceremos otro producto o presionamos q para salir del programa • •

