## **UTILIZAÇÃO DA LINGUAGEM PYTHON PARA O DESENVOLVIMENTO DE UM CONTROLADOR DE ESTOQUE: ESTUDO DE CASO EM UM PROVEDOR DE INTERNET**

## Andressa Hélen Gonçalves Ventura, (UFCG)

## andressa-sjt@hotmail.com

## Beatriz Silva dos Passos Oliveira, (UFCG)

## beattrizoliveira@hotmail.com

## José de Lima Henriques Júnior, (UFCG)

## jrhenrique01@gmail.com

**Resumo**:

Sabendo da importância de se utilizar softwares para a solução de problemas dentro de uma empresa, neste artigo será exposto a utilização da linguagem de programação Python para o desenvolvimento de um programa para um provedor de internet situado na cidade de Monteiro-PB que conta com um grande estoque de produtos para instalação e manutenção de internet, que com o qual não se tinha um controle, visando melhorias para a empresa foi proposto um programa básico de controle de estoque que atendesse as exigências e necessidades atuais da empresa, para a realização desse programa foram aplicados os conhecimentos da linguagem Python, que por ser uma linguagem básica, possibilita em uma perspectiva futura, ser alterado caso as necessidades do estoque mude. Após a implementação do programa de controle de estoque observou-se que este solucionou de forma fácil e rápida a otimização do estoque.

**Palavras-chave**: Python, linguagem de programação, controle de estoque.

## **Introdução**

Hoje em dia é de suma importância para as empresas utilizarem o máximo de informações e tecnologias disponíveis para manter vantagem competitiva. Sabendo disto, abordaremos no presente artigo a utilização de um software em linguagem de programação em python para a gestão do estoque de uma empresa provedora de internet e como este facilitará a manipulação dá entrada e saída dos produtos cadastrados.

## **Fundamentação teórica**

* 1. **Python**

Python é um exemplo de linguagem de programação de alto nível(DOWNEY, ELKNER, MEYERS,2009).Que foi desenvolvida por Guido Van Rossum em 1991. Uma linguagem de programação nada mais é do que um programa que traduz os comandos inseridos em linguagem de baixo nível - a linguagem que os computadores entendem - e há dois tipos de programas que executam esse tipo de linguagem, estes são: os interpretadores, que lê o programa aos poucos, alternando entre ler as linhas e as executar e os compiladores, lê o programa completamente e o executa.

De acordo com (DOWNEY, ELKNER, MEYERS,2009)Python é um interpretador, pois este tem uma linguagem interpretada, com aspectos de sintaxe considerada simples, que também apresenta características importantes das linguagens modernas (RAMALHO, 1998 -1999).

* + 1. **O que é um programa?**

Segundo SEVERANCE (2009-2013) um programa é uma sequência de comandos em uma ordem lógica que foram criadas para fazer alguma coisa. Onde na sua execução são coletados dados ou informações, denominados como entradas, afim de processá-los de acordo com as instruções contidas no programa e exibi-las ao usuário na tela, nomeado como saída.

Este processamento tem princípio no primeiro comando, seguido pelos demais comandos na sequência, sendo executados um por vez, esta orientação de comandos é chamada de fluxo de execução nível(DOWNEY, ELKNER, MEYERS,2009).

Os dados coletados durante a execução do programa, são armazenadas em variáveis, que são espaços lógicos na memória *RAM* do computador que serão utilizadas pelo programa, estas podem ser de tipos e tamanhos diferentes (siteforum).

* + 1. **Variáveis**

Uma variável, é um dos elementos básicos no qual um programa trabalha, podendo ser uma letra ou um número (SEVERANCE, 2009-2013),no Python há três tipos de variáveis, são elas:

* *Str* (string), chamada assim por conter uma “sequência” de letras, geralmente identificada por estarem entre aspas simples (SEVERANCE, 2009-2013).Podendo ainda compreender o conjunto de letras, números e caracteres;
* *Float***,** que é uma abreviação do termo *"floating point number"* ou número de ponto flutuante (RAMALHO, 1998 -1999), devido a suas casas decimais, ou seja, compreendem o conjunto de números Reais;
* *Int***,** que corresponde aos números inteiros.

Para haver um tipo de variável é necessário um comando de atribuição, que cria novas variáveis e dá a elas valores(DOWNEY, ELKNER, MEYERS,2009), ou seja, na inserção de dados para que o Python a reconheça e como variável trabalhe com elas é necessário determinar seu tipo por meio de um comando de atribuição, onde diz-se que a variável é igual ao tipo que deseja.

Porem algumas vezes para se obter uma variável é necessário perguntar ao usuário a informação desejada, para isso pode ser usado dois comandos:

* *Input;*
* *Raw\_Input*.

Onde ambos têm a mesma função, obtém o valor do teclado do usuário, quando este comando é executado, o programa para e espera o usuário digitar algo. Quando o usuário pressiona “Enter”, o programa continua (SEVERANCE, 2009-2013), mas para perguntar ao usuário as informações é fundamental imprimir na tela que informação deseja obter do mesmo, para isso é usado o comando *Print*, que sua principal função é imprimir informações na tela para o usuário, este serve também para imprimir os resultados finais do programa.

* + 1. **Comandos**

Tratando de comandos, que são as instruções que serão executadas pelo programa, será explorado apenas algumas especificações, visto que será de suma importância para compreender a proposta final do projeto.

* + 1. **Operadores**

Um símbolo especial que representa um cálculo simples (SEVERANCE, 2009-2013) estes comandos podem manipular as variáveis através de operadores matemáticos, operadores lógicos, operadores de comparação, são eles:

* Operadores matemáticos:São símbolos especiais como: +, -, \*, / e \*\* são de adição, subtração, multiplicação, divisão e exponenciação simples (SEVERANCE, 2009-2013), respectivamente, este tem aplicação igual na matemática;
* Operadores lógicos:Na programação há três operadores lógicos, and, or, not (e, ou, não) (DOWNEY, ELKNER, MEYERS,2009), que são usadas em condições onde para executa-los a sentença tem que ser atendida, geralmente utilizados juntos com os operadores de comparação;
* Operadores de comparação:Assim como os operadores matemáticos estes tem as mesmas funções que são aplicados na matemática, eles são: >, <, <=, >=, ==, != . maior, menor, maior igual, menor igual, igual e diferente, respectivamente.
	+ 1. **Condições e Sentenças**

Como foi mencionado anteriormente para os operadores de comparação estão diretamente ligadas as suas respectivas sentenças, que podem ser:

* *True* (verdadeiro), onde para que a condição seja executada a sentença tem que ser verdadeira;
* *False*(Falso), diferente de True, só haverá a execução da sentença se esta for falsa.

No o desenvolvimento de programas é natural que ao inserir dados em uma variável que pode percorrer caminhos distintos, para que estas percorram o caminho correto é necessário designar sentenças, para isto são usados os seguintes comandos:

* *If,* o mais simples e a expressão booleana após a sentença *If* é chamada de condição (SEVERANCE, 2009-2013)**,** que vem acompanhado de mais dois comandos *Elif* (se, se não), que é utilizado quando há 3 ou mais tipos de condições para a variável e *Else* (se não), que pode ser utilizado para finaliza o *If* junto com a cadeia de *Elif*, ou ainda utilizado quando se tem apenas duas condições;
* *While***,** utilizado para repetir tarefas idênticas ou similares sem gerar erros(DOWNEY, ELKNER, MEYERS,2009), através de uma condição determinada pelas sentenças. Que ao executar a primeira vez, o comando volta ao topo, verifica se a condição ainda é verdadeira e executa o While mais uma vez e assim sucessivamente. Este tipo de fluxo é chamado de um *loop* (ou laço) visto que o “*loop*” ou o laço de volta ao topo. Assim percebe-se que se for falsa na primeira vez que o While é verificado, os comandos dentro do loop jamais serão executados*(DOWNEY, ELKNER, MEYERS,2009).*

Apesar de ser só exposto dois tipos de condições há outros que não foram mencionados, demos foque nesses pois foram os utilizados no projeto.

* + 1. **Listas**

Quando se tem a necessidade de inserir vários dados em uma única variável, é utilizado o conceito que lista, onde está é um conjunto sequência ordenado de valores, que cada valor inserido é identificado por um índice. Os valores que compõem uma lista são chamados elementos*(DOWNEY, ELKNER, MEYERS,2009).*Na utilização de listas é possível aplicar várias operações e condições, além conter seus comandos específicos, como por exemplo:

* Max: Retorna o maior valor dentro de uma coluna*(CONTEMBITS,2010-2017);*
* Min**:** retorna o menor valor dentro de uma coluna*(CONTEMBITS,2010-2017);*
* Join: Retorna uma string com todos os elementos da sequência concatenados (slide de strings, site de Cecir).

Dentro das listas existem outros inúmeros comandos, porem para desenvolver nosso programa nos limitamos a esses acima citados.

* + 1. **Banco de dados**

No decorrer da execução de um programa, os dados inseridos pelos usuários ficam na guardados na memória *RAM*. Quando o programa encerra, ou o computador é desligado, os dados na memória se perdem. Para armazenar os dados permanentemente, você tem que colocá-los em um arquivo, que é identificado por um nome único, ou uma combinação de uns nomes(DOWNEY, ELKNER, MEYERS,2009).

No momento que desejamos executar o arquivo, é informando ao sistema o nome dado e a tarefa que queremos que executar nele. Mas antes de informar isto, devemos abri-lo, com o comando *open*, ele que vai receber os argumentos citados anteriormente, nome e o modo que deseja executa-lo, o Python executa dois tipos de modo:

* *Write*, “w’ (escrever), que abri o arquivo para gravação. Se não existir nenhum arquivo com o nome teste.dat, ele será criado. Se já existir um, ele será substituído pelo arquivo que estamos gravando (ou escrevendo)(DOWNEY, ELKNER, MEYERS,2009);
* *Reading***,** (ler), abre o arquivo para leitura, onde este lê os dados do arquivo, utilizando o argumento “r”. Se tentarmos abrir um arquivo que não existe, o programa indicara um erro, o comando também pode receber um argumento que indica quantos caracteres quer ler: Se não houver caracteres suficientes no arquivo, este retorna os caracteres restantes (DOWNEY, ELKNER, MEYERS,2009).

Uma observação quando se trata de arquivos, é que todos os dados devem estar no formato strings. Compreendendo todos estes conceitos podemos partir para definição de banco de dados:

Banco de dados, são como super arquivos, que armazenam dados em esquemas predefinidos, e mantém relações entre itens de dados que lhe permitem acessar os dados de várias formas *(DOWNEY, ELKNER, MEYERS,2009).*

Todas as definições, conceitos e comandos foram especificados para o projeto, porem o Python tem várias outras funções e comandos que não foram descritos no referencial teórico, visto que como mencionado, foi descrito apenas algumas funções e as especificações de cada uma, como: comandos, condições e etc.

1. **Metodologia**

A pesquisa adotou ao mesmo tempo as metodologias de campo e referencial bibliográfico onde através de visitas à empresa, conhecemos o estoque da mesma, seu funcionamento e percebemos a dificuldade que seus gestores tem em controlar o estoque.

Visto isso, encontramos na linguagem Python uma forma de solucionar tal problema criando um código de programação basico que gerenciasse os *inputs* e ***outputs*** da empresa, para tal solução tivemos que realizar visitas a empresa, analisar o estoque e como é feita a sua gestão, logo após aplicamos um questionário para o funcionário responsável pelo gerenciamento do estoque onde obtivemos as seguintes informações, capacidade máxima e mínima do estoque, quantidade atual contida no estoque e como se dá o gerenciamento do estoque, a parti disto.

1. **Resultados e discursão**

A aplicação do Programa de Controle do Estoque na empresa propõe os seguintes resultados: a empresa se manter com um estoque suficiente para suprir suas necessidades, sem que esta tenha muitos produtos parados por muito tempo, afinal estoque é capital imobilizado. Segue abaixo a tabela demonstrando como funciona o controle de estoque, e a quantidade máxima e mínima atribuída para cada produto.

Tabela 1 – Produtos contidos no estoque

|  |  |  |  |
| --- | --- | --- | --- |
| **ITENS** | **QUANTIDADE** | **MÍNIMO** | **MÁXIMO** |
| Roteador intelbras wrn 240 | 14 | 60 | 350 |
| Roteador intelbras wrn 150 | 8 | 60 | 350 |
| Roteador linkone rw 131 | 1 | 60 | 350 |
| Roteador linkone rw 141 | 3 | 60 | 350 |
| Roteador tp link wrn 740 | 40 | 60 | 350 |
| Roteador tp link wrn 720 | 6 | 60 | 350 |
| Roteador mercusys mw 355 | 333 | 60 | 350 |
| Rádio nano m5 | 33 | 10 | 50 |
| Rádio nano m2 | 9 | 5 | 15 |
| Rádio wom 5000 | 15 | 10 | 50 |
| Rádio airgrid | 6 | 5 | 15 |
| Swicht vlan | 28 | 10 | 100 |
| Swicht amadeus | 19 | 5 | 10 |
| Swicht 16 portas | 3 | 3 | 4 |
| Swicht gerenciável | 16 | 10 | 60 |
| Antena nanobeam | 8 | 2 | 5 |
| Bobina de cliente | 27 | 4 | 30 |
| Bobina de rota | 4 | 4 | 10 |
| Bobina de energia | 1 | 1 | 2 |
| Conector (pacote) | 60 | 40 | 100 |
| Abraçadeira (pacote) | 9 | 2 | 10 |
| Fonte 5 volts | 4 | 2 | 5 |
| Roteador kronnus | 22 | - | - |
| Roteador abocon | 8 | - | - |
| Roteador smart lan | 8 | - | - |
| Roteador mimax | 1 | - | - |
| Roteador aquario | 1 | - | - |
| Roteador tplink antigo | 3 | - | - |
| Roteador air live | 1 | - | - |
| Roteador multilaser | 1 | - | - |
| Roteador global | 2 | - | - |
| Roteador d link | 1 | - | - |
| Buchas (pacotes) | 8 | 4 | 10 |
| Presilhas (pacotes) | 3 | 2 | 4 |
| Antena sem fio | 8 | 2 | 5 |
| Parafusos (caixa com 200) | 3 | 1 | 4 |
| Alicate de clipar | 4 | 2 | 5 |
| Rocket m5 | 2 | 2 | 4 |
| Fita de aço | 15 | 10 | 20 |
| Rádio sxt 5hnd | 13 | 5 | 20 |
| Caixa de comando | 4 | 2 | 5 |
| Esticador (saco) | 2 | 1 | 4 |
| Baterias | 4 | 2 | 10 |
| Fita isolante | 21 | 10 | 40 |
| Hotspot 300 | 4 | 2 | 10 |
| Caixa plástica preta | 16 | 10 | 100 |

Fonte: Disponível pelo provedor de internet (2017)

Outro aspecto que o Programa visa melhorar no estoque da empresa é a otimização do tempo usado para o gerenciamento dos inputs e outputs, pois, como este controle é feito usando o Excel, há perda de tempo, e com o Programa em Python será utilizado menos tempo.

Uma das vantagens observadas da sua utilização como Controlador de Estoque é a facilidade de introduzir dados de entrada e a minimização de tempo na execução das tarefas realizadas, essa minimização do uso de tempo para desempenhar as atividades na execução do programa em python tornando ainda mais eficaz a escolha do método de resolução do problema enfrentado pela empresa.

1. **Conclusão**

A utilização de um programa elaborado em Python é um meio abrangente e funcional para aplicação do mesmo como controlador de estoque de uma empresa. Contudo, sua utilização vai além do objeto de estudo apresentado, e pode ser usado como ferramenta de trabalho para auxiliar diversas atividades.

Uma das maiores vantagens da sua utilização como Controlador de Estoque se dá com a facilidade de introduzir dados de entrada, sendo estes o cadastro de produtos informado pelo gerenciador do programa, a minimização de tempo na execução das tarefas realizadas, e a extrema facilidade de permutar informações já redigidas no programa. Sendo, portanto, essa minimização do uso de tempo para desempenhar as atividades na execução do programa em Python que torna ainda mais eficaz a escolha do método de resolução do problema enfrentado pelo provedor de internet. Com isso podemos concluir que o objetivo principal de pesquisa se encontra devidamente solucionado através das medidas e estratégias adotadas.

1. **Referências**

SEVERANCE, C.**Python for Informatics: Exploring Information.** Amazon CreateSpace 2009-2013**.**

RAMALHO, L.**Aprenda a programar.** São Paulo:RevistaMagnet, 1998 -1999.

DOWNEY, Allen; ELKNER, Jeff; MEYERS, Chris. **Aprenda Computação com Python 3.0**. [S.l.: s.n.], 2009.

Disponível em: < https://sites.google.com/site/ceciralmeidafarias/programacao-i/slides-aulas> Acesso em: 01 set. 2017

Disponível em: <http://contembits.com.br/minicursos/mysql/Funcao-Min-e-Max.aspx> Acesso em: 30 agost. 2017

Disponível em: <http://www.tiexpert.net/programacao/algoritmo/tipos-de-dados.php> Acesso em: 01 set. 2017

Disponível em: <http://www.estube.com.br/blog/?p=478>Acesso em: 01 set. 2017

Disponível em: <http://www.scielo.org/php/index.php>Acesso em: 30 set. 2017