

Educação Profissional Paulista

Técnico em
**Desenvolvimento
de Sistemas**

Estruturas de seleção

Atividades práticas – Criação de programas

Aula 1

Código da aula: [SIS]ANO1C1B2S9A1

Exposição



Objetivos da aula

- Compreender a aplicação prática das estruturas de seleção na lógica de programação;
- Conhecer exemplos de aplicações no dia a dia utilizando *Python*.



Competências da unidade (técnicas e socioemocionais)

- Desenvolver sistemas computacionais utilizando um ambiente de desenvolvimento;
- Explorar a criatividade na resolução de problemas computacionais.



Recursos didáticos

- Recursos audiovisuais para exibição de imagens;
- Caderno, canetas e lápis;
- Computador com acesso à Internet.



Duração da aula

50 minutos.

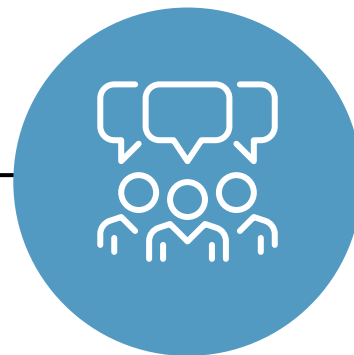
Exposição

Desenvolvimento da aula

- ✓ Aplicação prática de algoritmos;
- ✓ Programação aplicada à resolução de problemas;
- ✓ Utilização de linguagens de programação para o aprendizado.

Exposição

Vamos programar



Atividade

Vamos criar um programa em **Python**.



Meta do programa

Lê uma letra do **alfabeto** e **exibe um objeto** cujo nome começa com a mesma letra.



Instruções

Utilizaremos a **estrutura if, elif e else** para realizar a comparação e a exibição do objeto correspondente.

Vamos programar

1. O programa deverá ler primeiro a letra indicada pelo usuário e convertê-la em minúscula;
2. Em seguida, precisa comparar a letra com cada uma das possíveis letras do alfabeto usando a estrutura *if*, *elif* e *else*.



Atenção!

Se a letra digitada estiver entre 'a' e 'z', o programa exibirá o objeto correspondente. Caso contrário, exibirá uma mensagem de erro.

Exposição



© Getty Images

Exemplo de uso em *python* (Parte 1)

Recebe a entrada do usuário

```
letra = input("Digite uma letra do alfabeto: ")
```

Converte a letra em minúscula para evitar problemas

```
letra = letra.lower()
```

Verifica a letra digitada e mostra o objeto correspondente

```
if letra == 'a':  
    print("Avião")  
elif letra == 'b':  
    print("Bola")  
elif letra == 'c':  
    print("Carro")  
elif letra == 'd':  
    print("Dado")  
elif letra == 'e':  
    print("Elefante")  
elif letra == 'f':  
    print("Foca")  
elif letra == 'g':  
    print("Gato")  
elif letra == 'h':  
    print("Hipopótamo")
```


Exposição



© Getty Images

Exemplo de uso em *python* (Parte 2)

```
elif letra == 'i':  
    print("Igreja")  
elif letra == 'j':  
    print("Janela")  
elif letra == 'u':  
    print("Uva")  
elif letra == 'k':  
    Print("Kalunga")
```

fazer o mesmo com as outras letras do alfabeto até...

```
else:  
    print("Não conheço uma palavra que comece com essa letra.")
```


Exposição

Maiúsculas e minúsculas

Em *Python*, ***lower()*** e ***upper()*** são dois **métodos de *strings*** usados para manipular e transformar o conteúdo de uma *string*.



Tome nota

Ambos os métodos não alteram a *string* original, mas retornam uma nova ***string*** com as alterações aplicadas.

Exposição

Métodos *lower()* e *upper()*

	<i>String</i> para minúsculas	<i>String</i> para maiúsculas
<i>lower()</i>	É usado para converter todos os caracteres alfabéticos de uma <i>string</i> em minúsculas. Ele não afeta outros caracteres que não são letras.	X
<i>upper()</i>	X	É usado para converter todos os caracteres alfabéticos de uma <i>string</i> em maiúsculas. Assim como o <i>lower()</i> , ele não afeta os caracteres que não são letras.

Exemplos:

Método *lower()*

```
texto = "Olá, Mundo!"  
texto_min = texto.lower()  
print(texto_min)
```

Saída:
olá, mundo!

Método *upper()*

```
texto = "Olá, Mundo!"  
texto_maiusculo = texto.upper()  
print(texto_maiusculo)
```

Saída:
OLÁ, MUNDO!

Vamos
fazer uma
atividade

Construir um programa em *Python* que permita a interpretação, em código, de letras do alfabeto.



30 minutos



Esta atividade deve ser entregue no AVA.

Exercícios práticos – Criação de programas



Materiais necessários

- Acesso a um computador com internet.



Passo a passo

1. Abertura do programa de desenvolvimento em *Python*;
2. Introdução sobre o enunciado dos exercícios e contextualização do que é solicitado;
3. Resolução e testes dos códigos produzidos.



Faça download da explicação da atividade



O que nós
**aprendemos
hoje?**

Hoje desenvolvemos:

- 1 Conhecimento de como usar a estrutura condicional *if*, *elif* e *else* para fazer seleções em nosso código, permitindo executar diferentes blocos de código com base em condições específicas;
- 2 Aprendizado sobre os métodos *lower()* e *upper()* em *Python*, que nos permitem transformar *strings* em letras minúsculas ou maiúsculas, respectivamente, tornando mais fácil fazer comparações de *strings* ou padronizar o seu formato;
- 3 Compreensão de como usar a função *input()* para receber entradas de dados do usuário no console. Essa função permite que o usuário insira informações no programa que podem ser processadas e utilizadas para diferentes finalidades, como realizar cálculos, exibir resultados ou interagir com o usuário de maneira dinâmica.

© Getty Images

Saiba mais

No vídeo a seguir, é possível saber mais sobre a **profissão de programador** e sobre como iniciar a carreira.

CURSO EM VÍDEO. *Curso Python #01* – Seja um programador. Disponível em:
https://www.youtube.com/watch?v=S9uPNp_pGsGo&source_ve_path=OTY3MTQ&feature=emb_imp_woyt Acesso em: 13 fev. 2024.

Referências da aula

MACHADO, V. *Estruturas de Seleção*. Rio de Janeiro: IBMEC, [s.d.]. Disponível em: https://victor0machado.github.io/prog/notas_aula/006-estruturas-selecao.html. Acesso em: 13 fev. 2024.

CURSO EM VÍDEO. *Curso Python #01 – Seja um programador*. Disponível em: https://www.youtube.com/watch?v=S9uPNppGsGo&source_ve_path=OTY3MTQ&feature=emb_imp_woyt. Acesso em: 13 fev. 2024.

Identidade visual: Imagens © Getty Images

Educação Profissional Paulista

Técnico em
**Desenvolvimento
de Sistemas**