PythonLab

Prof. Dr. Álvaro Campos Ferreira

idp

Comentários

Comentários

Existem duas formas de fazer comentários:

- # comentário de uma linha
- “”” Bloco de comentários “””

Variáveis

Declarando e nomeando variáveis

Podem conter apenas letras, números e sublinhado/underscore

- Espaços não são permitidos
- Evite palavras-chave Python
- Curto e descritivo
- Cuidado com l e O, parecem 1 e 0

idp

Declarando e nomeando variáveis

- Use nomes que revelem a intenção da variável
- Evite desinformação
- Use nomes pronunciáveis
- Use nomes pesquisáveis
- Não use trocadilhos

Tipos das variáveis

Tipos de variáveis

Para cada tipo de variável, as funções tem comportamento diferente.

- Texto (str)
- Número inteiro (int)
- Número real (float)
- Booleano (bool), True ou False

Tipos de variáveis

Ao declarar uma variável, o Python determina seu tipo dinamicamente, uma variável pode mudar de tipo apenas declarando-a novamente:

```
idade = “Trinta e dois” # tipo texto
idade = 32 # tipo inteiro
```

Listas

Declarando listas

Uma lista pode ter vários argumentos de diferentes tipos.

```
minhaLista = [‘eu’, ‘amo’, ‘Python’]
minhaLista[0]
```

Declarando listas

Modificando elementos de uma lista

```
minhaLista = [‘eu’, ‘amo’, ‘Python’]
minhaLista[2] = ‘software livre’
```

idp

Dicionários

Dicionários

Declarando um dicionário com duas entradas e acessando cada uma delas:

```
meuDict = {‘linguagem’: ’Python’, ’idade’: ‘28 anos’}
meuDict[‘linguagem’]
meuDict[‘idade’]
```

Controle de fluxo

Laços for ou “para cada”

Com **for**, o bloco irá executar para cada elemento do objeto.

```
for cliente in clientes:
    print(cliente)
```

idp

Laços while ou “enquanto”

O **while** executará o bloco “enquanto” a condição a direita dele resultar em verdadeiro, ou booleano True.

```
while nome == “Álvaro”:
    print(“Olá, professor!”)
```

idp

Blocos if... else ou “se... senão”

O **if** ou “se”, executa o bloco se a condição for satisfeita.

```
if nome == “Álvaro”:
    print(“Bom dia, professor”)
else:
    print(“Bom dia, querido colega”)
```

idp

Funções

Funções

Define-se funções com a palavra-chave **de**
da palavra definir:

```
def somar_numeros(a,b):
    return a + b
```

Documentando funções

A forma padrão de documentar uma função chama-se Docstring e nada mais é que um bloco de comentário.

```
def somar_numeros(a,b):
 """ Soma dois números.
 """
        return a + b
```

Retornando valores

Uma função pode retornar um ou mais valores após a execução.

```
def somar_numeros(a,b):
 """ Soma dois números.
 """

     return a + b
```

Módulos

Módulos de funções

É comum e boa prática gerar arquivos com funções relacionadas para que possam ser importados em outros scripts.

O funcionamento é semelhante a importar bibliotecas.

Módulos de funções

Nesses casos, é útil saber quando um módulo está sendo importado e quando está sendo executado como script e definir comportamentos diferentes dependendo do caso.

Módulos de funções

```
def main():
    print(“Meu módulo é o máximo.”)
if __name__ == ‘__main__’:
    main()
```

Bibliotecas

Planejamento e especificação

INSTITUTO BRASILEIRO DE ENSINO,
DESENVOLVIMENTO E PESQUISA