



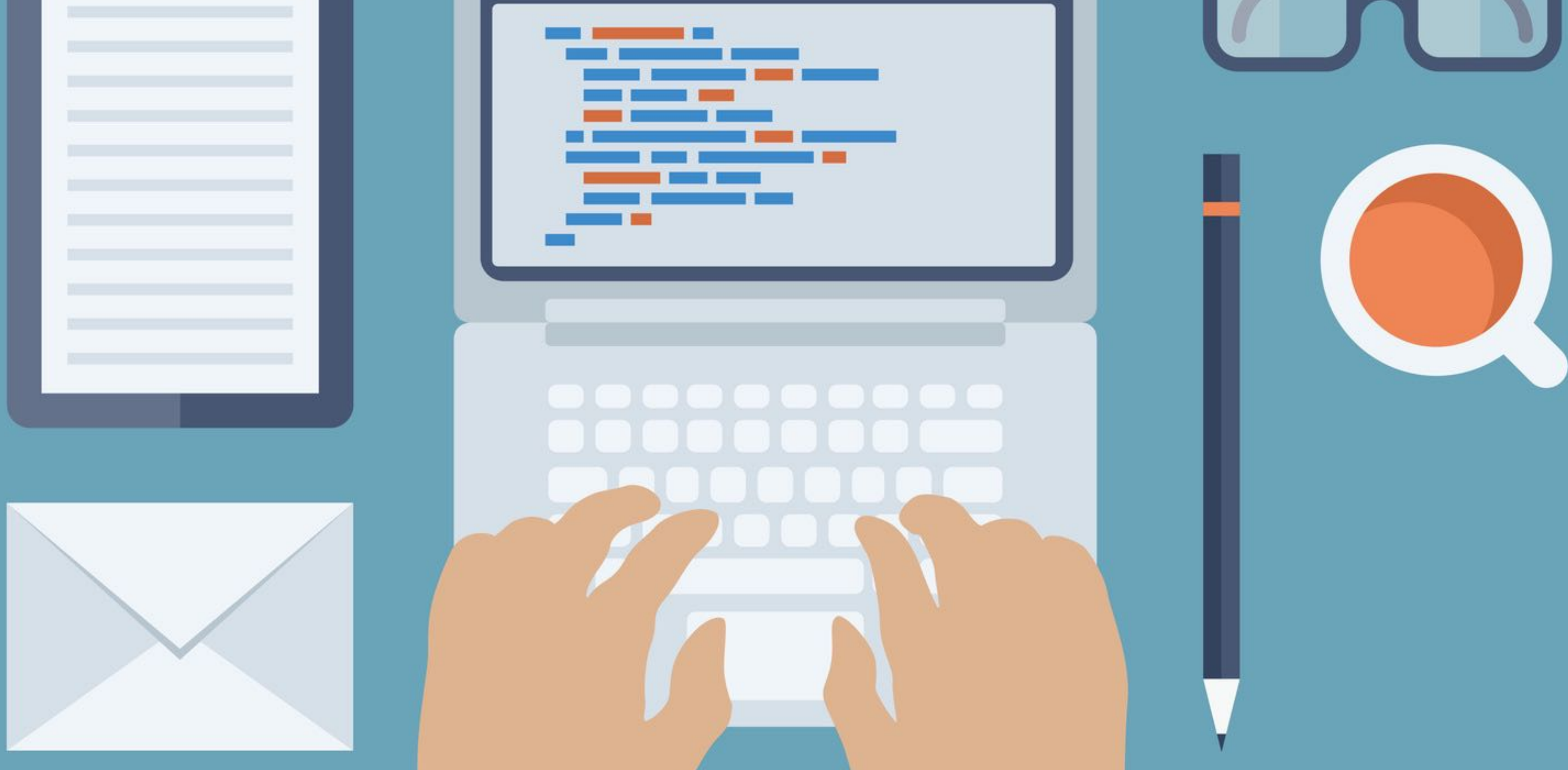
# CK0211 - Fundamentos de Programação: Registros

Emanuele Santos

Bibliografia: Ascencio, Cap. 10

# Objetivos

- Aprender a criar e manipular conjuntos de tipos de dados heterogêneos em algoritmos e em Python



# INTRODUÇÃO

# Introdução

- As **variáveis** armazenam apenas um tipo de dado
- Algumas vezes precisamos armazenar, dentro de uma mesma variável, **diferentes tipos de dados**
- Exemplo: programa para gerenciar as notas de uma disciplina
  - Cada aluno possui nome, número de matrícula, nota AP1, nota AP2
- **Registros**: agregam vários dados acerca de uma mesma entidade

# Registros

- Tipos de dados heterogêneos:
  - Tipos de dados cujos elementos podem ser de tipos diferentes
- Variável composta por **campos**
  - Cada **campo** possui um **nome** e pode ser de um **tipo** diferente
  - Exemplo: registro de passagem de ônibus:

```
Número da passagem: _____ Data: _____  
Origem: _____ Destino: _____  
Horário: _____ Poltrona: _____ Idade: _____  
Nome do passageiro: _____
```

# Registros

- A maior funcionalidade dos registros é poder gerar novos tipos de dados, assim o programador não fica limitado aos tipos de dados primitivos



# REGISTROS EM ALGORITMOS

# Declaração de registros em algoritmos

- Para declarar uma variável do tipo REGISTRO, precisamos especificar:
  - Nome da variável
  - Os nomes e tipos dos seus campos



# Declaração de registros em algoritmos

```
DECLARE nome_da_variavel REGISTRO (nome_campo1
TIPO_DO_CAMP01, nome_campo2 TIPO_DO_CAMP02,..., nome_campoN
TIPO_DO_CAMPON)
```

- Exemplo:

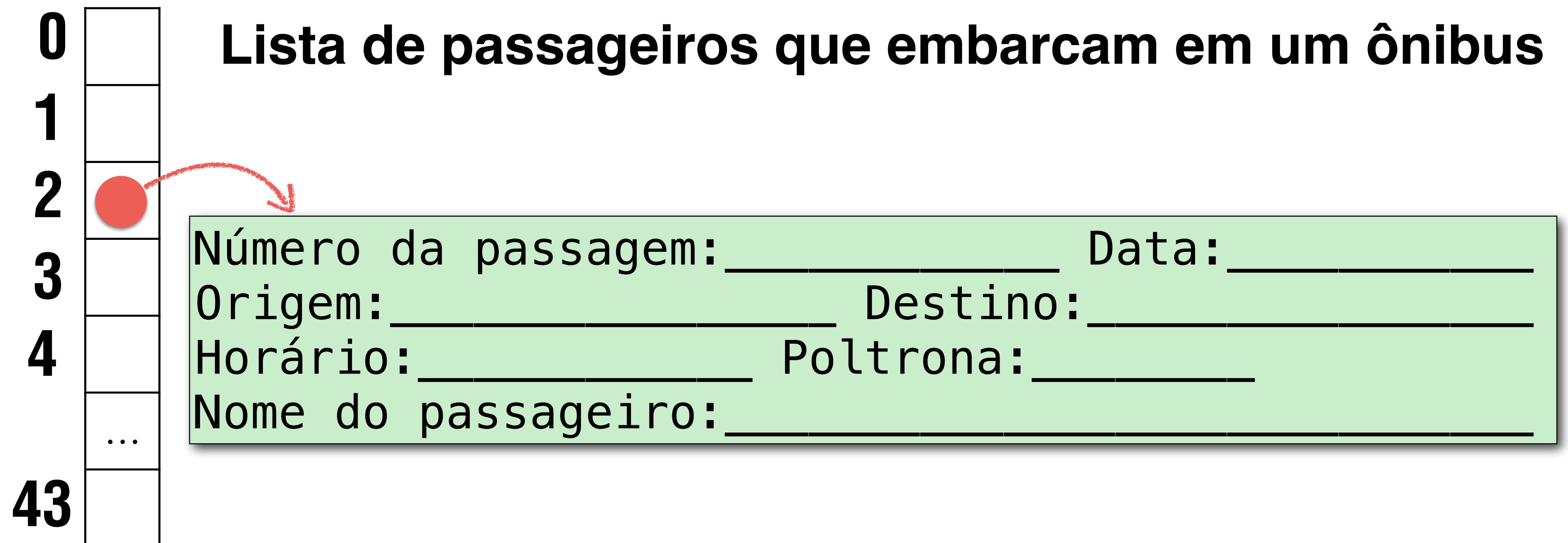
```
DECLARE conta REGISTRO (num, saldo NUMÉRICO, nome LITERAL)
```

**Variável conta**

```
num: _____
saldo: _____
nome: _____
```

# Declaração de registros em algoritmos

- Assim como nos outros tipos de dados, uma variável registro pode ser simples (exemplo anterior), um vetor ou uma matriz



# Declaração de registros em algoritmos

```
DECLARE conta[3] REGISTRO (num, saldo NUMÉRICO, nome LITERAL)
```

## Vetor conta

1	2	3
num: _____ saldo: _____ nome: _____	num: _____ saldo: _____ nome: _____	num: _____ saldo: _____ nome: _____

# Acesso aos campos de um registro

- Podemos acessar cada campo de um registro, seja para gravar ou para recuperar um dado
- O acesso a um determinado campo de registro é feito informando-se o nome da variável, seguido por um ponto e pelo nome do campo desejado.
- Forma geral:

```
nome_da_variavel_registro.nome_do_campo
```

# Exemplo 1: preenchendo registro simples

```
DECLARE conta REGISTRO (numero, saldo NUMÉRICO, titular  
LITERAL)  
ESCREVA "Digite o número da conta: "  
LEIA conta.numero  
ESCREVA "Digite o nome do titular da conta: "  
LEIA conta.titular  
ESCREVA "Digite o saldo da conta: "  
LEIA conta.saldo
```

# Exemplo 2: preenchendo vetor de registros

```

DECLARE funcionario[4] REGISTRO (nome LITERAL, salario
NUMÉRICO)
        i NUMÉRICO
PARA i ← 1 ATÉ 4 FAÇA
INÍCIO
    ESCREVA "Digite o nome do funcionário ", i, ": "
    LEIA funcionario[i].nome
    ESCREVA "Digite o salário do funcionário ", i, ": "
    LEIA funcionario[i].salario
FIM

```

# Exemplo 2

## Vetor funcionario

1	2	3	4
nome: João salario: 1000,00	nome: Maria salario: 5000,00	nome: Pedro salario: 1800,00	nome: Lúcia salario: 2700,00

# Exemplo 3: mostrando registros em vetor

**PARA**  $i \leftarrow 1$  **ATÉ** 4 **FAÇA**

**INÍCIO**

**ESCREVA** “Funcionário que ocupa a posição ”,  $i$ , “no vetor: ”

**ESCREVA** “Nome: ”, `funcionario[i].nome`

**ESCREVA** “Salário: ”, `funcionario[i].salario`

**FIM**

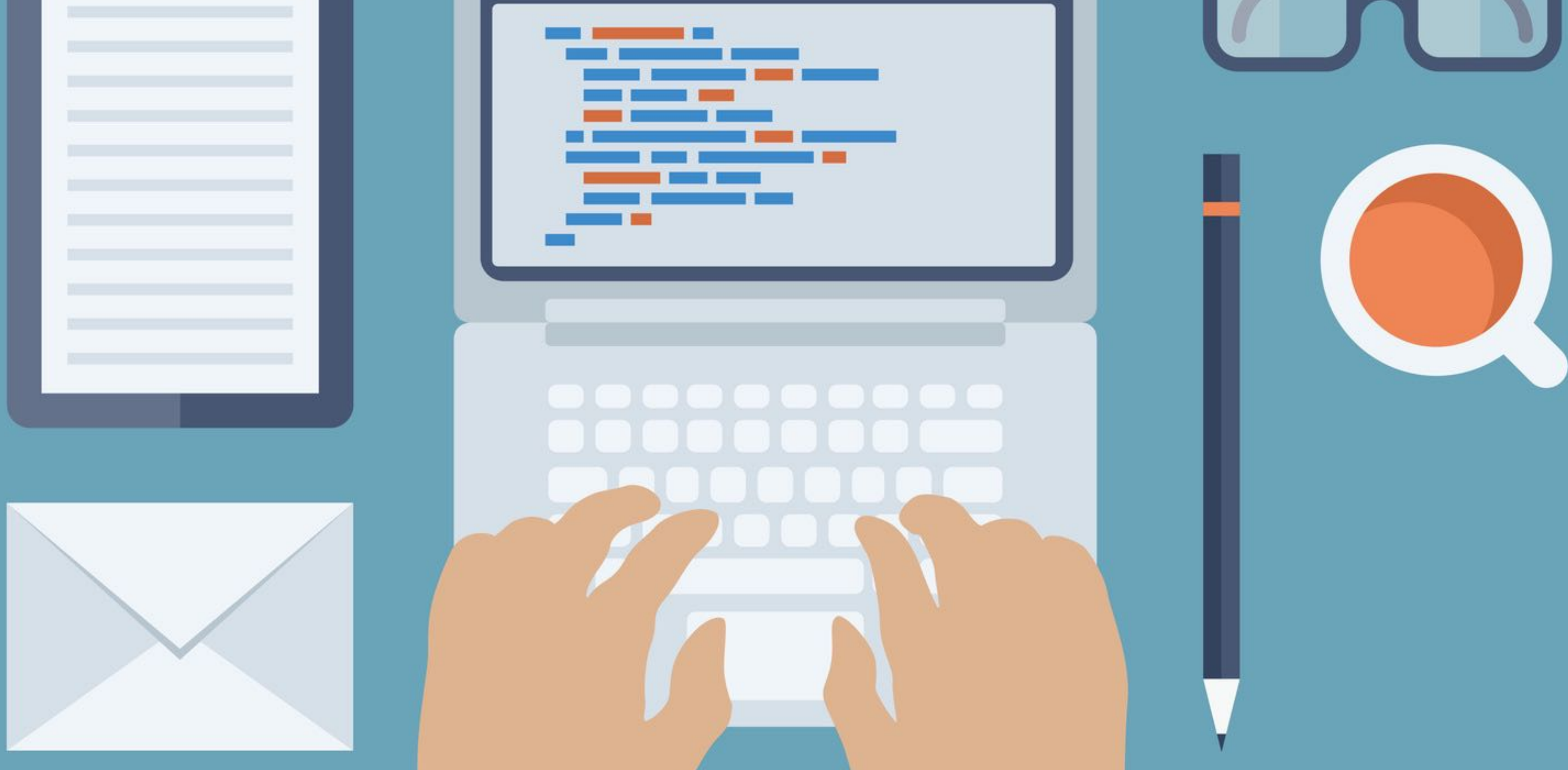


# Exercício 1:

- Faça um programa para criar um estoque de produtos. Cada produto possui um código, um preço e uma quantidade. O estoque pode armazenar 10 produtos. Depois de receber os produtos do estoque, mostre um relatório com os produtos, a quantidade total de itens e o valor total do estoque.
- Use uma função `cria_produto` para criar um registro com código, preço e quantidade e outra função `mostra_produto` para mostrar os valores dos campos de um produto

# Exercício 2:

- Incremente o programa anterior, acrescentando uma função que busca um determinado produto por código. A função deve retornar o registro do produto. O programa deve pedir para o usuário digitar um código, buscar o produto, e se encontrá-lo mostrar os dados do produto.



# REGISTROS EM PYTHON

# Registros em Python

- Em Python, representamos registros através de **dicionários**
- Podemos criar dicionários de duas maneiras:
  - Já com todos os campos e valores:

```
nome_registro = { 'nome_campo1' : conteudo_campo1,  
                  'nome_campo2' : conteudo_campo2,  
                  ...,  
                  'nome_campoN' : conteudo_campoN }
```

# Registros em Python

- Inicializa um dicionário vazio e insere os campos e valores depois:

```
nome_registro = {}  
nome_registro['nome_campo1'] = conteudo_campo1  
nome_registro['nome_campo2'] = conteudo_campo2  
...  
nome_registro['nome_campoN'] = conteudo_campoN
```

# Registros em Python

**Variável conta**

```
num: _____
saldo: _____
nome: _____
```

```
conta = {'num' : 123456,
        'saldo' : 1000.00,
        'nome' : 'João Silva'}
```

OU

```
conta = {}
conta['num'] = int(input("Digite o número da conta: "))
conta['nome'] = input("Digite o nome do titular da conta: ")
conta['saldo'] = float(input("Digite o saldo da conta: "))
```

# Acesso aos campos de um dicionário

- Para acessar os campos de um dicionário, usamos a seguinte notação:

```
nome_registro['nome_campo']
```

# Exemplo 1:

```
DECLARE conta REGISTRO (numero, saldo NUMÉRICO, titular  
LITERAL)  
ESCREVA "Digite o número da conta: "  
LEIA conta.numero  
ESCREVA "Digite o nome do titular da conta: "  
LEIA conta.titular  
ESCREVA "Digite o saldo da conta: "  
LEIA conta.saldo
```

```
conta = {}  
conta['numero'] = int(input("Digite o número da conta: "))  
conta['titular'] = input("Digite o nome do titular da conta: ")  
conta['saldo'] = float(input("Digite o saldo da conta: "))
```



# Exemplo 2: Preenchendo lista de dicionários

```
funcionarios = [] # lista vazia
for i in range(4): # quatro funcionarios
    funcionario = {} # registro funcionario
    funcionario['nome'] = input("Digite o nome do funcionário %d: "%i)
    funcionario['salario'] = float(input("Digite o salário do \
funcionário %d: " % i))
    funcionarios.append(funcionario) # adiciona o dicionário à lista
```

# Exemplo 3: Mostrando dicionários em lista

```
for i in range(4):  
    print("Funcionário que ocupa a posição %d no vetor:" % i)  
    print("Nome: %s" % funcionarios[i]['nome'])  
    print("Salário: R$ %.2f" % funcionarios[i]['salario'])
```

# Exercício 1:

- Faça um programa em Python para criar um estoque de produtos. Cada produto possui um código, um preço e uma quantidade. O estoque pode armazenar 10 produtos. Depois de receber os produtos do estoque, mostre um relatório com os produtos, a quantidade total de itens e o valor total do estoque.
- Use uma função `cria_produto` para criar um registro com código, preço e quantidade e outra função `mostra_produto` para mostrar os valores dos campos de um produto

# Exercício 2:

- Incremente o programa anterior, acrescentando uma função que busca um determinado produto por código. A função deve retornar o registro do produto. O programa deve pedir para o usuário digitar um código, buscar o produto, e se encontrá-lo mostrar os dados do produto.