

Capítulo 3

Mini - Projeto para 2024-01-12-SA

Resolva os exercícios das duas secções seguintes. Entregar ficheiro zip com:

- **Nome:** <Número>__<PrimeiroNome>__<UltimoNome>__Ementas__Avaliacoeszip
- **Conteúdo:** dados, programa e relatório word/pdf
- **Local:** <https://moodle.ipg.pt/mod/assign/view.php?id=75326>

3.1 Exercício 1 - Ementas Semana: 12-01-2024-SA

Elabore um programa em *Python* para gerar o texto a que se referem as ementas. Deve gerar o texto em português e inglês, de acordo com a Figura 3.1.

Como entrada para o programa é fornecida uma data. O programa deve determinar a data da segunda-feira e sexta-feira e de seguida gerar o texto tendo em consideração quer a mudança de mês quer a mudança de ano. Repare que o mês e ano só se repetem quando são diferentes.

Teste o programa para os seguintes datas:

Tabela com diretivas de formatação para a função `datetime.strftime` <https://www.programiz.com/python-programming/datetime/strftime>

```
import locale
locale.setlocale(locale.LC_ALL, '')
from datetime import datetime, timedelta
data_atual = datetime.now()
data_atual = datetime.strptime('2024-01-10', '%Y-%m-%d')
data_seg = data_atual
while True:
    ds = int(datetime.strftime(data_seg, '%w')) + 1    # 0..6 - 1
    if ds == 2:
        break
    else:
```

| Data | Dia da semana |
|------------|---------------|
| 2023-11-23 | quinta-feira |
| 2024-01-22 | segunda-feira |
| 2024-03-22 | sexta-feira |
| 2024-05-21 | terça-feira |
| 2024-06-30 | domingo |
| 2024-08-29 | quinta-feira |
| 2024-10-28 | segunda-feira |
| 2024-12-27 | sexta-feira |
| 2025-02-25 | terça-feira |
| 2025-04-26 | sábado |

Tabela 3.1: Casos de Teste Ementas.

```
data_seg = data_seg - timedelta(days=1)
data_sex = data_seg + timedelta(days=4)
print(data_seg, datetime.strftime(data_seg, '%w %A'))
print(data_atual, datetime.strftime(data_atual, '%w %A'))
print(data_sex, datetime.strftime(data_sex, '%w %A'))
# 2024-01-08 00:00:00 1 segunda-feira
# 2024-01-10 00:00:00 3 quarta-feira
# 2024-01-12 00:00:00 5 sexta-feira
# Semana de 08 a 12 de janeiro de 2024
```

Listagem 3.1: Exemplo com datas: datetime, locale, timedelta e strftime

| Directive | Meaning | Example |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------|
| %a | Abbreviated weekday name. | Sun, Mon, ... |
| %A | Full weekday name. | Sunday, Monday, ... |
| %w | Weekday as a decimal number. | 0, 1, ..., 6 |
| %d | Day of the month as a zero-padded decimal. | 01, 02, ..., 31 |
| %-d | Day of the month as a decimal number. | 1, 2, ..., 30 |
| %b | Abbreviated month name. | Jan, Feb, ..., Dec |
| %B | Full month name. | January, February, ... |
| %m | Month as a zero-padded decimal number. | 01, 02, ..., 12 |
| %-m | Month as a decimal number. | 1, 2, ..., 12 |
| %y | Year without century as a zero-padded decimal number. | 00, 01, ..., 99 |
| %-y | Year without century as a decimal number. | 0, 1, ..., 99 |
| %Y | Year with century as a decimal number. | 2013, 2019 etc. |
| %H | Hour (24-hour clock) as a zero-padded decimal number. | 00, 01, ..., 23 |
| %-H | Hour (24-hour clock) as a decimal number. | 0, 1, ..., 23 |
| %I | Hour (12-hour clock) as a zero-padded decimal number. | 01, 02, ..., 12 |
| %-I | Hour (12-hour clock) as a decimal number. | 1, 2, ... 12 |
| %p | Locale's AM or PM. | AM, PM |
| %M | Minute as a zero-padded decimal number. | 00, 01, ..., 59 |
| %-M | Minute as a decimal number. | 0, 1, ..., 59 |
| %S | Second as a zero-padded decimal number. | 00, 01, ..., 59 |
| %-S | Second as a decimal number. | 0, 1, ..., 59 |
| %f | Microsecond as a decimal number, zero-padded on the left. | 000000 - 999999 |
| %z | UTC offset in the form +HHMM or -HHMM. | |
| %Z | Time zone name. | |
| %j | Day of the year as a zero-padded decimal number. | 001, 002, ..., 366 |
| %-j | Day of the year as a decimal number. | 1, 2, ..., 366 |
| %U | Week number of the year (Sunday as the first day of the week). All days in a new year preceding the first Sunday are considered to be in week 0. | 00, 01, ..., 53 |
| %W | Week number of the year (Monday as the first day of the week). All days in a new year preceding the first Monday are considered to be in week 0. | 00, 01, ..., 53 |
| %c | Locale's appropriate date and time representation. | Mon Sep 30 07:06:05 2013 |
| %x | Locale's appropriate date representation. | 09/30/13 |
| %X | Locale's appropriate time representation. | 0,295891203703704 |


Tabela 3.2: Diretivas de formatos para criar datas e horas a partir de strings.

POLI
TÉCNICO
GUARDA



SAS-IPG | Ementas (Menus)


Semana de 8 a 12 de janeiro de 2024
Week from 8th to 12th January, 2024

Cantina 1 - IPG - Campus



<https://ipg.iuope.com>
Reservas

|  | Almoço |  | Lunch |
|-----------------------------------------------------------------------------------|-------------------------------------------|-----------------------------------------------------------------------------------|------------------------------------------------------------------|
| 2024-01-08 (segunda-feira) | | Monday, January 08, 2024 | |
| Sopa | Feijão verde | Soup | Green beans soup |
| Carne | Costeletas grelhadas com arroz primavera | Meat | Grilled pork chops with rice |
| Peixe | Abrótea estufada com batata cozida | Fish | Stewed whiting with boiled potatoes |
| Vegetariano | Ratatouille | Vegetarian | Ratatouille |
| Sobremesa | Fruta | Dessert | Fruit |
| 2024-01-09 (terça-feira) | | Tuesday, January 09, 2024 | |
| Sopa | Ovo | Soup | Egg soup |
| Carne | Lombo assado com arroz de cenoura | Meat | Baked pork loin with carrot rice |
| Peixe | Pescada à marinheiro | Fish | "seaman" hake |
| Vegetariano | Paella vegetariana | Vegetarian | Vegetarian paella |
| Sobremesa | Gelatina | Dessert | Gelatin |
| 2024-01-10 (quarta-feira) | | Wednesday, January 10, 2024 | |
| Sopa | Caldo verde | Soup | Soup of shredded kale in potato broth |
| Carne | Espetadas de peru com esparguete salteado | Meat | Turkey skewers with spaghetti sauteed |
| Peixe | Solha frita com arroz de tomate | Fish | Fried plaice with tomato rice |
| Vegetariano | Arroz de ervilhas com cogumelos | Vegetarian | Rice peas with mushrooms |
| Sobremesa | Fruta | Dessert | Fruit |
| 2024-01-11 (quinta-feira) | | Thursday, January 11, 2024 | |
| Sopa | Abóbora com brócolos | Soup | Pumpkin soup with broccoli |
| Carne | Carne de porco à alentejana | Meat | Fried pork meat with clams and chips |
| Peixe | Massa de atum gratinada | Fish | Pasta with tuna au gratin |
| Vegetariano | Estufado de soja | Vegetarian | Soybean stew |
| Sobremesa | Iogurte | Dessert | Yogurt |
| 2024-01-12 (sexta-feira) | | Friday, January 12, 2024 | |
| Sopa | Canja de galinha | Soup | Chicken broth |
| Peixe | Bacalhau à gomes de sá | Fish | Flaked codfish w/ potatoes,hard boiled eggs,parsley,black olives |
| Vegetariano | Couscous com legumes salteados | Vegetarian | Couscous with sautéed vegetables |
| Sobremesa | Iogurte | Dessert | Yogurt |



<https://politecnicoguarda.pt/portal-do-estudante/ementas>
Consulta ementas

SAS-IPG, 04-01-2024 09:54:28

Figura 3.1: Ementas Cantina 1 - IPG - Campus - Semana de 8 a 12 de janeiro de 2024

3.2 Exercício 2 - Datas de Atendimento às Avaliações: 12-01-2024-SA

As avaliações na Escola Superior de Tecnologia e Gestão (ESTG) ocorrem no final da parte letiva de cada semestre. A Direção da ESTG publica, no portal do estudante¹, as datas das avaliações das disciplinas. Os docentes devem dar atendimentos dois dias úteis antes de cada avaliação. Para cada dia, o atendimento deve ter duração não inferior a três horas.

As datas das avaliações para o 1º semestre do ano letivo de 2023/2024 e para a ESTG, podem ser descarregadas do link:

- [Licenciaturas](#)
- [CTeSP](#)

Os docentes até à ultima semana letiva têm de enviar para o sistema de gestão documental as datas e respetivas horas de atendimento.

Considerando que lecionam em cada semestre cerca de três centenas de docentes seria de todo o interesse automatizar o processo de determinação das datas e horas de atendimento para todos os docente de disciplinas.

Elabore um programa em *Python* para determinar os dias e as horas de atendimento de um dado docente com base nas suas avaliações. Considere as horas de atendimentos 9:00-12:00 ou 15:00-17:00.

A Listagem 3.2 ilustra um exemplo para um docente com as seguintes avaliações:

- TESP CiberSegurança - Programação para a Internet - 18/01/2024 10:00 - * Frequência
- TESP CiberSegurança - Programação para a Internet - 29/01/2024 10:00 - * Exame
- TESP CiberSegurança - Programação para a Internet - 07/02/2024 14:00 - * Recurso
- Licenciatura em Engenharia Informática - Algoritmos e Estruturas de Dados - 19/01/2024 10:00 - * Frequência
- Licenciatura em Engenharia Informática - Algoritmos e Estruturas de Dados - 02/02/2024 14:00 - * Exame
- Licenciatura em Engenharia Informática - Algoritmos e Estruturas de Dados - 09/02/2024 10:00 - * Recurso

¹https://cloud.sysnovare.pt/ipg/web_base.gera_pagina?p_pagina=IPG_CALENDARIOS_AVALIACAO

```
# Dicionário: {'chave' : 'valor', ... }
cursos = {'TC' : 'TESP CiberSegurança', 'LEI' : 'Licenciatura em Engenharia Informática'}
disciplinas = {'PI' : 'Programação para a Internet', 'AED', 'Algoritmos e Estruturas de Dados'}
dic = {'F': '* Frequência', 'E': '* Exame', "R": '* Recurso', 'ATM': 'Atendimento'}
datasTexto = [
    ('TC', 'PI', '18/01/2024 10:00', 'F'),
    ('TC', 'PI', '29/01/2024 10:00', 'E'),
    ('TC', 'PI', '07/02/2024 14:00', 'R'),
    ('LEI', 'AED', '19/01/2024 10:00', 'F'),
    ('LEI', 'AED', '02/02/2024 14:00', 'E'),
    ('LEI', 'AED', '09/02/2024 10:00', 'R')
]

# resultado
Curso Disciplina Data          Das      às      Tipo
TC      PI      ter 16-01-2024    09:00    12:00    Atendimento
TC      PI      qua 17-01-2024    14:00    17:00    Atendimento
LEI      AED      qua 17-01-2024    14:00    17:00    Atendimento
LEI      AED      qui 18-01-2024    14:00    17:00    Atendimento
TC      PI      qui 18-01-2024    10:00    12:00    * Frequência
LEI      AED      sex 19-01-2024    10:00    12:00    * Frequência
TC      PI      qui 25-01-2024    09:00    12:00    Atendimento
TC      PI      sex 26-01-2024    09:00    12:00    Atendimento
TC      PI      seg 29-01-2024    10:00    12:00    * Exame
LEI      AED      qua 31-01-2024    09:00    12:00    Atendimento
LEI      AED      qui 01-02-2024    09:00    12:00    Atendimento
LEI      AED      sex 02-02-2024    14:00    16:00    * Exame
TC      PI      seg 05-02-2024    09:00    12:00    Atendimento
TC      PI      ter 06-02-2024    09:00    12:00    Atendimento
LEI      AED      qua 07-02-2024    09:00    12:00    Atendimento
TC      PI      qua 07-02-2024    14:00    16:00    * Recurso
LEI      AED      qui 08-02-2024    09:00    12:00    Atendimento
LEI      AED      sex 09-02-2024    10:00    12:00    * Recurso
```

Listagem 3.2: Datas e Horas para o Atendimento às avaliações 2º semestre ano letivo 2023/2024 para um docente de Informática.

A Listagem 3.3 contém um pedaço de código em *Python* que pode usar na resolução dos exercício.

```
# Dicionário: {'chave' : 'valor', ... }
cursos = {'TC': 'TESP CiberSegurança', 'LEI': 'Licenciatura em Engenharia Informática'}
disciplinas = {'PI': 'Programação para a Internet', 'AED': 'Algoritmos e Estruturas de Dados'}
tipos = {'F': '* Frequência', 'E': '* Exame', "R": '* Recurso', 'ATM': 'Atendimento'}
datasTexto = [
    ('TC', 'PI', '18/01/2024 10:00', 'F'),
    ('TC', 'PI', '29/01/2024 10:00', 'E'),
    ('TC', 'PI', '07/02/2024 14:00', 'R'),
    ('LEI', 'AED', '19/01/2024 10:00', 'F'),
    ('LEI', 'AED', '02/02/2024 14:00', 'E'),
    ('LEI', 'AED', '09/02/2024 10:00', 'R')
]

# listas as avaliações
for cur, dis, data, tipo in datasTexto:
    print(cursos[cur], disciplinas[dis], data, tipos[tipo], sep=' - ')

import locale
locale.setlocale(locale.LC_ALL, '')
from datetime import datetime, timedelta

def DataDiaAnterior(data, dias):
```

```

data_ant = data
data_ant = data_ant - timedelta(days=dias)
# ... sab, dom
return data_ant

print(disciplinas['PI'])
data = datetime.strptime('2024-01-18', '%Y-%m-%d')
datas_atm = []
da = DataDiaAnterior(data, 2)
datas_atm.append(da)
# datas_atm.append((cur, dis, da, tipo)
da = DataDiaAnterior(data, 1)
datas_atm.append(da)
datas_atm.append(data)

for d in datas_atm:
    print(d)

# fazer para todas as datas
# for cur, dis, data_s, tipo in datasTexto:
#     data = ..
# ordenar o vetor por data

```

Listagem 3.3: Exemplo com datas: dicionários, listas datetime, locale, timedelta e strftime

3.2.1 Casos de Teste

Teste o programa para os docentes 2 e 3 da Listagem 3.4.

```

Docente 2
Licenciatura em Ciência de Dados e Inteligência Artificial
  Aprendizagem Automática I
    02/02/2024 14:00 Exame
    08/02/2024 10:00 Exame Recurso

Licenciatura em Engenharia Informática
  Programação para a Internet
    05/02/2024 17:00 Exame
    10/02/2024 10:00 Exame Recuso

Docente 3
Licenciatura em Engenharia Informática
  Engenharia de Redes
    17/01/2024 14:00 Frequência
    03/02/2024 14:00 Exame
    14/02/2024 17:00 Exame Recurso

Mestrado em Computação Móvel
  Redes e Serviços de Comunicações
    24/01/2024 14:00 Frequência
    02/02/2024 14:00 Exame
    10/02/2024 11:00 Exame Recurso

TESP Cibersegurança
  Redes de Computadores II
    17/01/2024 14:00
    03/02/2024 14:00
    14/02/2024 14:00

```

Listagem 3.4: Exemplo com datas: dicionários, listas datetime, locale, timedelta e strftime