

FICHA TÉCNICA DA AÇÃO DE FORMAÇÃO

Criar aplicações em linguagem de programação Python

UC00618 · 50 horas · Formação a Distância (Microsoft Teams)

Ação de formação	UC00618 — Criar aplicações em linguagem de programação Python (50 horas · 4,5 pontos de crédito)
Modalidade / Regime	Formação Modular — Ativos Empregados · Formação a Distância (Microsoft Teams)
Datas	08/10/2026 a 30/11/2026 · 2.ª e 5.ª feiras, 18:00–21:00
Formador	Luís Ferreira
Entidade formadora	IEFP, I.P. — Centro de Emprego da Maia
Contactos	Rua Dr. Carlos Felgueiras, 418 · 4470-157 Maia · Tel.: 220 989 190 · Email: ce.maia@iefp.pt
Referencial	https://catalogo.snq.gov.pt/ucDetalhe/355650

Ficha Técnica da Ação de Formação

Designação	UC00618 — Criar aplicações em linguagem de programação Python
Referencial	Catálogo Nacional de Qualificações — https://catalogo.snq.gov.pt/ucDetalhe/355650
Pontos de crédito	4,5
Modalidade	Formação Modular — Ativos Empregados
Regime	Formação a Distância (Microsoft Teams) — conta gratuita de acesso ao Portal Office facultada pelo IEFP
Duração	50 horas — 16 sessões (14 × 3 h + 2 × 4 h)
Datas e horário	08/10/2026 a 30/11/2026 · 2.ª e 5.ª feiras, 18:00–21:00 (16/11 e 23/11: 18:00–22:00)
Destinatários	Ativos empregados. Não é exigida experiência prévia de programação.
Formador	Luís Ferreira
Entidade formadora	IEFP, I.P. — Centro de Emprego da Maia
Contactos	Rua Dr. Carlos Felgueiras, 418 · 4470-157 Maia · Tel.: 220 989 190 · Email: ce.maia@iefp.pt · Coordenação de formação: carlos.r.goncalves@iefp.pt

Realizações (resultados de aprendizagem)

No final da UC, os formandos deverão ser capazes de:

- Planear e estruturar os programas.
- Criar scripts em Python.
- Testar e depurar programas.

Critérios de desempenho

Criar aplicações em linguagem de programação Python:

- Cumprindo as fases metodológicas no desenvolvimento de software.
- Cumprindo regras no uso dos elementos e sintaxe da programação.
- Executando a programação e corrigindo erros.

Programa (organização dos conteúdos em 8 módulos)

1. Pensamento computacional e algoritmos
2. Python e ambiente de desenvolvimento (Anaconda, VS Code, Spyder, Jupyter)
3. Fundamentos da linguagem: tipos de dados, coleções e controlo do programa
4. Funções, ficheiros, iteradores e geradores

5. Programação orientada a objetos
6. Tratamento de dados: texto, CSV, Excel e SQL (Pandas e NumPy)
7. Análise gráfica e bibliotecas: Matplotlib, Turtle/Pygame e widgets
8. Testes, depuração, desempenho e documentação

Conteúdos do referencial (conhecimentos)

- Pensamento computacional — princípios.
- Algoritmos — princípios.
- Metodologias de desenvolvimento de software.
- Linguagem de programação Python — características; vantagens; diferenças entre versões; utilizações de Python.
- Anaconda e ambientes de desenvolvimento integrado — IDE (Spyder e VS Code).
- Jupyter Notebook.
- Tratamento de dados de várias fontes: TEXTO, CSV, SQL, XLS.
- Programação em Python — variáveis; tipos de dados; tuplos; listas; dicionários; controlo do programa (if, match, for e while).
- Manipulação de ficheiros e iteradores.
- Benchmark.
- Profilers de memória e CPU.
- Widgets.
- Geradores.
- Programação orientada a objetos em Python — classes e objetos; métodos e atributos; construtores; herança; decoradores.
- Bibliotecas — Pandas, Turtle e NumPy ou Pygame; análise gráfica com Matplotlib; importação de SQL, CSV.
- Testes unitários.
- Funcionalidades de um editor de texto.
- Normas e regulamentos aplicáveis.

Aptidões

- Interpretar manuais, guiões e tutoriais técnicos.
- Interpretar os princípios do pensamento computacional.
- Criar algoritmos.
- Reconhecer as características, elementos e sintaxe da linguagem de programação.
- Utilizar orientações metodológicas para planear as etapas de criação de um programa.
- Instalar o ambiente de desenvolvimento.
- Utilizar as funcionalidades do ambiente de desenvolvimento para programar em Python.
- Utilizar os elementos e a sintaxe da linguagem Python para programar scripts.
- Utilizar bibliotecas em programas criados em linguagem Python.
- Corrigir os erros identificados durante os testes.
- Criar um guião técnico, relatório ou manual de utilizador do programa.
- Aplicar normas e regulamentos.

Atitudes

Responsabilidade pelas suas ações · Autonomia no âmbito das suas funções · Empenho e persistência na resolução de problemas · Ética · Iniciativa · Rigor · Sentido crítico · Sentido de organização · Respeito pelas regras e normas definidas.

Metodologia

Formação síncrona a distância (Microsoft Teams), com metodologia ativa e demonstrativa: exposição breve dos conceitos com demonstração ao vivo no VS Code/Jupyter, seguida de exercícios práticos individuais acompanhados (partilha de ecrã para apoio). Cada módulo articula teoria e aplicação, com dados e casos próximos da realidade profissional dos formandos. O curso culmina num mini-projeto de tratamento de dados e num projeto final individual.

Avaliação

Avaliação contínua (exercícios entregues por módulo — 30%), projeto final (30%) e teste de avaliação final (40%), na escala de 0 a 20 valores. Considera-se «com aproveitamento» a classificação final igual ou superior a 10 valores. A assiduidade rege-se pelo Regulamento do Formando do IEFP.

Recursos

- Computador com Windows, macOS ou Linux, câmara, microfone e ligação à Internet
- Microsoft Teams (sessões síncronas) e conta do Portal Office facultada pelo IEFP
- Python 3.14.7 (python.org) e Visual Studio Code com as extensões Python e Jupyter
- Bibliotecas: pandas, numpy, matplotlib, openpyxl, pygame, pytest
- Anaconda/Spyder (opcional, abordado na sessão 3)
- Manual do formando, caderno de exercícios e ficheiros de exercícios (ZIP)
- Página de apoio do curso (teste diagnóstico e materiais)