



TÉCNICO LISBOA

# Fundamentos da Programação LEIC/LETI

## Aula 01

Apresentação

Ambiente de desenvolvimento (PYTHON3)

# Objetivos

- Fornecer conhecimentos sobre conceitos fundamentais relativos à atividade de **programação**:
  - Algoritmo
  - Abstração procedimental e abstração de dados
  - Programação como construção de abstrações
  - paradigmas de programação, etc.
- Os alunos deverão:
  - Dominar os conceitos apresentados
  - Ser capazes de desenvolver programas numa linguagem de programação de alto nível, o **Python**.

# Corpo docente

## Aulas Teóricas (Responsável)



**Alberto Abad** Responsável  
alberto.abad@tecnico.ulisboa.pt

## Aulas Práticas



**Ana Almeida Matos**  
ana.matos@tecnico.ulisboa.pt



Carlos Manuel Ferreira Carvalho  
carlos.f.carvalho@tecnico.ulisboa.pt



João Miguel Ferreira Oliveira  
joao.ferreira.oliveira@tecnico.ulisboa.pt



**Pedro Tiago Gonçalves Monteiro**  
pedro.tiago.monteiro@tecnico.ulisboa.pt



**Rúben Miguel de Almeida Inácio**  
ruben.inacio@tecnico.ulisboa.pt

# Organização UC

## Aulas teóricas

- 6 semanas x 5 aulas x 1 hora (4 Out – 12 Nov)
- Feriados sem aulas: 5 Out e 1 Nov



## Aulas práticas

- 11 aulas x 2 horas (7 Out – 12 Nov)
- 6 MINI-TESTES: L02, L04, L06, L08, L10, L11 + Repesc. (T28)
- Regras: Turno próprio e frequência



## Projetos

- **Proj1** Enunciado: 15 Out; Entrega: 5 Nov @ 17h00
- **Proj2** Enunciado: 29 Out; Entrega: 19 Nov @ 17h00



## Exame repescagem

- 24 Novembro @ 13h00



# Método de avaliação

## Época normal: Avaliação contínua 100%

- Nota testes aulas práticas (L)  
 $L = 10\% L1 + 15\% L2 + 15\% L3 + 20\% L4 + 20\% L5 + 20\% L6$   
(possibilidade substituir um teste L1-L5 no último dia de aulas teóricas )
- Nota projeto (P)  
 $P = 40\% P1 + 60\% P2$ ; P1 e P2 notas de 1o e 2o projeto;

$$NF = 50\% P + 50\% L$$

(se  $P \geq 9.5$  e  $L \geq 9.5$ )

## Época recuso: Avaliação contínua (projeto) + Exame

- Nota de exame (E)

$$NF = 50\% P + 50\% \max\{E, L\}$$

(se  $P \geq 9.5$  e  $\max\{E, L\} \geq 9.5$ )

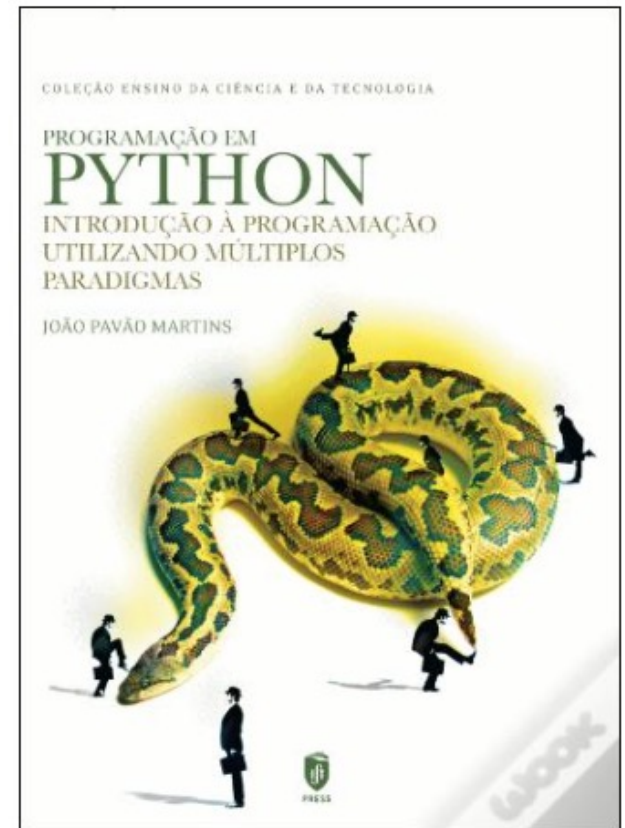
**Outros:** Trabalhadores estudantes; época especial

# Programa/Planeamento aulas

Semana	Dia	Teóricas			Lab		Avaliação	
		Aula	Tópico	Livro	Aula	Tópico		
1	04-Oct	1	Apresentação da cadeira. Ambiente de desenvolvimento.			Sem aula		
	05-Oct	Feriado República						
	06-Oct	2	Computadores, algoritmos e programas. Sintaxe e semântica.	1.1 - 1.4				
	07-Oct	3	Expressões. Tipos elementares de dados. Nomes e atribuição. Predicados e condições. Comunicação com o exterior.	2.1 - 2.5	1			Sintaxe.
	08-Oct	4	Sequenciação. Seleção. Repetição.	2.6 - 2.9				
2	11-Oct	5	Ambientes. Abstração procedimental. Ambientes. Abstração procedimental.	3.1 - 3.4	2	Elementos básicos de programação.	Ficha 1: Sintaxe/BNF (10%)	
	12-Oct	6	Erros. Módulos. Visualização e execução de programas. Depuração.	3.5 - 14				
	13-Oct	7	Tuplos.	4.1				
	14-Oct	8	Ciclos contados.	4.2	3	Funções.	Publicação 1º Proj	
3	15-Oct	9	Cadeias de caracteres revisitadas. Formatação.	4.3 - 4.4				
	18-Oct	10	Listas. Método de passagem de parâmetros.	5.1 - 5.2	4	Tuplos e ciclos contados.	Ficha 2: Elementos básicos + Funções (15%)	
	19-Oct	11	Listas. O crivo de Eratóstenes.	5.3				
	20-Oct	12	Algoritmos de procura e de ordenação.	5.4 - 5.5				
4	21-Oct	13	O tipo dicionário. Frequência de letras num texto. Dicionários de dicionários.	8.1 - 8.3	5	Listas.		
	22-Oct	14	Discussão enunciado projeto. Boas práticas. Dúvidas. Exercícios e exemplos da matéria anterior.					
	25-Oct	15	Abstração em programação. Números complexos. Essência da abstração de dados. Complexos como dicionários.	9.1 - 9.2	6	Dicionários.	Ficha 3: Tuplos, listas e ciclos contados (15%)	
	26-Oct	16	Tipos abstratos de dados.	9.3 - 9.4				
	27-Oct	17	Ficheiros: open, leitura.	10.1 - 10.2				
5	28-Oct	18	Ficheiros: escrita, with statement, modulos para leitura.	10.3	7	Abstração de dados.	Publicação 2º Proj	
	29-Oct	19	Discussão de exercícios e exemplos. Dúvidas do projeto.					
	01-Nov	Feriado Todos os Santos						
	02-Nov	20	Funções revisitadas. Programação funcional. Estruturação de funções. Funções internas. Scope.	6.1 - 6.3	8	Ficheiros.	Ficha 4: Dicionários + Abstração (20%)	
6	03-Nov	21	Recursão e iteração. Recursão de operações adiadas.	7.1 - 7.2				
	04-Nov	22	Recursão de cauda.	7.3	9	Recursão.	Entrega 1º Proj 17:00	
	05-Nov	23	Recursão em árvore e recursão múltipla. Considerações sobre eficiência.	7.4 - 7.5				
	08-Nov	24	Funções de ordem superior. Funções como parâmetros. Funções Lambda.	6.4	10	Processos iterativos e recursivos.	Ficha 5: Ficheiros + Recursão (20%)	
	09-Nov	25	Funções de ordem superior. Funcionais sobre listas I.	6.4				
7	10-Nov	26	Funções de ordem superior. Funcionais sobre listas II.	6.4				
	11-Nov	27	Funções de ordem superior. Funções como valor.	6.5	11	Funcionais sobre listas.	Ficha 6: Funcionais + Ex. Sumário (20%)	
	12-Nov	28	Ficha de recuperação. Dúvidas projeto. Epílogo					
	15-Nov							
8	16-Nov							
	18-Nov							
	19-Nov							
							Entrega 2º Proj 17:00	

# Bibliografia (I)

1. COMPUTADORES, ALGORITMOS E PROGRAMAS
2. ELEMENTOS BÁSICOS DE PROGRAMAÇÃO
3. FUNÇÕES
4. TUPLOS E CICLOS CONTADOS
5. LISTAS
6. FUNÇÕES REVISITADAS
7. RECURSÃO E ITERAÇÃO
8. DICIONÁRIOS
9. ABSTRAÇÃO DE DADOS
10. FICHEIROS
- ~~11. PROGRAMAÇÃO COM OBJETOS~~
- ~~12. ESTRUTURAS LINEARES~~
- ~~13. ÁRVORES~~
14. O DESENVOLVIMENTO DE PROGRAMAS



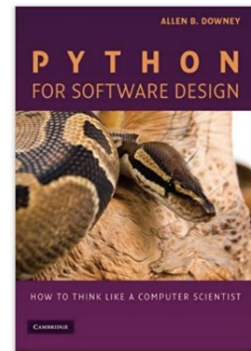
# Bibliografia (II)

## Principal

- *Programação em Python: Introdução à programação com múltiplos paradigmas:*

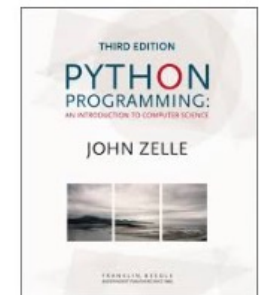
João P. Martins 2013 IST Press

- *Python for Software Design:* Downey A.B 2009 Cambridge University Press

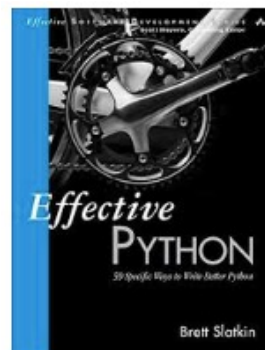
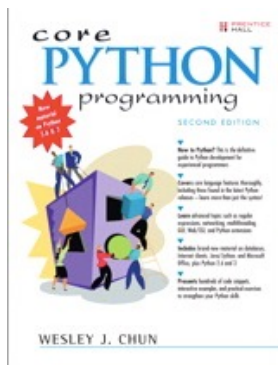


## Secundária

- *Python Programming: An Introduction to Computer Science:* Zelle J.M. 2010 Franklin, Beedle & Associates



## Outros





# Material de Apoio

Página Inicial · Fundamentos d... x +

fenix.tecnico.ulisboa.pt/disciplinas/FProg/2021-2022/1-semester

**Ufi** TÉCNICO LISBOA

## Fundamentos da Programação + info

PT / EN

### Últimos anúncios

**Início das aulas de Fundamentos da Programação**  
29 Setembro 2021, 18:18 · Alberto Abad

Bem-vindos a Fundamentos da Programação da LEIC-T/LET!!!!


Este ano as **aulas teóricas** terão início na segunda-feira dia 4 de outubro de 2021.


As **aulas práticas** só terão início no dia 7 de outubro de 2021 (IMPORTANTE: os alunos só podem assistir ao turno de laboratório em que estejam inscritos).


Mais informações sobre o planeamento de aulas e a organização da disciplina podem ser encontradas na página da disciplina e também serão facilitadas nos primeiros dias de aulas.


Qualquer dúvida ou questão deve ser enviada para o endereço oficial da disciplina: [fp-tagus@tecnico.ulisboa.pt](mailto:fp-tagus@tecnico.ulisboa.pt).

### Corpo Docente

 **Alberto Abad** Responsável  
[alberto.abad@tecnico.ulisboa.pt](mailto:alberto.abad@tecnico.ulisboa.pt)

 **Ana Almeida Matos**  
[ana.matos@tecnico.ulisboa.pt](mailto:ana.matos@tecnico.ulisboa.pt)

 **Carlos Manuel Ferreira Carvalho**  
[carlos.f.carvalho@tecnico.ulisboa.pt](mailto:carlos.f.carvalho@tecnico.ulisboa.pt)

 **João Miguel Ferreira Oliveira**

- Página Inicial
- Grupos
- Avaliação
- Bibliografia
- Horário
- Métodos de Avaliação
- Objectivos
- Planeamento
- Programa
- Túrnos
- Anúncios
- Sumários
- Notas
- Planeamento das aulas
- Pauta
- Horário de dúvidas
- Contatos
- Material de Apoio**

# Horários de atendimento

## Alberto Abad (responsável)

- Terças das 10h00 às 10h30: Auditório A2 (presencial)
- Quartas das 11h30 às 13h00: Sala 2.N11.21 (presencial)
- Quintas das 11h00 às 11h30: Auditório A2 (presencial)
- Quintas das 15h00 às 17h00: Zoom (<https://videoconf-colibri.zoom.us/j/83957266218>)



## Pedro Monteiro

- Segundas das 14h00 às 16h00: Sala 2N1.25 (presencial)
- Sextas das 13h00 às 14h00: Sala 2N1.25 (presencial)



## Ana Matos TBD

- Segunda das 13h30 às 16h30: Zoom (online)



## Rúben Inácio

- Quintas das 13h00 às 14h30: Zoom (<https://zoom.us/j/9088669634?pwd=R01kRlVRVXExamICdnFvMnFIYTlZz09>)
- Sextas das 15h20 às 16h50: Sala 2.N11.15 (presencial)



## João Oliveira

- Quartas das 13h00 às 14h30: Sala 2N11.15 (presencial)
- Sextas das 11h30 às 12h30: Sala 2N11.15 (presencial)



## Carlos Carvalho

- Terça das 14h30 às 16h00: Sala 2N11.15 (presencial)



# Comunicação e informações

## Página da UC

<https://fenix.tecnico.ulisboa.pt/disciplinas/FProg/2021-2022/1-semester>

## Mail oficial da UC

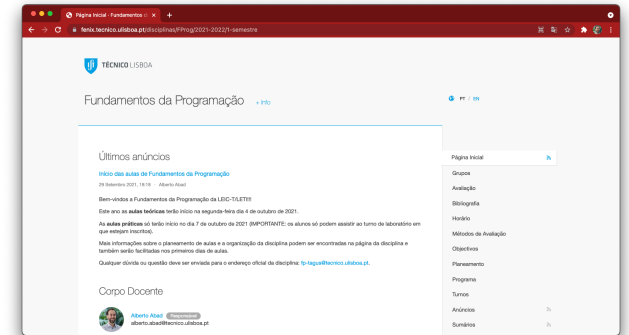
[fp-tagus@tecnico.ulisboa.pt](mailto:fp-tagus@tecnico.ulisboa.pt)

## Slack da UC

<http://fp-tagus.slack.com>

#annoucements, #fp2021-duvidas,  
#random

(registrar com @tecnico.ulisboa.pt)



# Fraude



# Ambiente de desenvolvimento PYTHON ( $\geq 3.6$ )

# Porque o Python?

- Alto nível
- Oferece múltiplos paradigmas:
  - *functional, object-oriented, imperative, ...*
- Escalável/reutilizável/extensível
- Portável
- Eficiente e rápido (prototipagem)
- Gestão de memória e tipos dinâmicos
- Fácil de aprender e ler
- Muito utilizado

# Porque o Python?

## TIOBE Index for September 2018

### September Headline: Python enters the TIOBE index top 3 for the first time

Python has entered the TIOBE index top 3 for the first time in its history. This really took a long time. At the beginning of the 1990s it entered the chart. Then it took another 10 years before it reached the TIOBE index top 10 for the first time. After that it slowly but surely approached the top 5 and eventually the top 3. Python is becoming increasingly ubiquitous. It is already the first choice at universities (for all kinds of subjects for which programming is demanded) and is now also conquering the industrial world. Python's selling points are easy to learn, easy to install and easy to deploy.

Sep 2018	Sep 2017	Change	Programming Language	Ratings	Change
1	1		Java	17.436%	+4.75%
2	2		C	15.447%	+8.06%
3	5	▲	Python	7.653%	+4.67%
4	3	▼	C++	7.394%	+1.83%
5	8	▲	Visual Basic .NET	5.308%	+3.33%
6	4	▼	C#	3.295%	-1.48%
7	6	▼	PHP	2.775%	+0.57%
8	7	▼	JavaScript	2.131%	+0.11%
9	-	▲	SQL	2.062%	+2.06%
10	18	▲	Objective-C	1.509%	+0.00%
11	12	▲	Delphi/Object Pascal	1.292%	-0.49%
12	10	▼	Ruby	1.291%	-0.64%
13	16	▲	MATLAB	1.276%	-0.35%
14	15	▲	Assembly language	1.232%	-0.41%

# Porque o Python?

## TIOBE Index for October 2020

October Headline: Python is getting closer to the second position













Oct 2020	Oct 2019	Change	Programming Language	Ratings	Change
1	2	▲	C	16.95%	+0.77%
2	1	▼	Java	12.56%	-4.32%
3	3		Python	11.28%	+2.19%
4	4		C++	6.94%	+0.71%
5	5		C#	4.16%	+0.30%
6	6		Visual Basic	3.97%	+0.23%
7	7		JavaScript	2.14%	+0.06%
8	9	▲	PHP	2.09%	+0.18%
9	15	▲▲	R	1.99%	+0.73%
10	8	▼	SQL	1.57%	-0.37%
11	19	▲▲	Perl	1.43%	+0.40%
12	11	▼	Groovy	1.23%	-0.16%
13	13		Ruby	1.16%	-0.16%
14	17	▲	Go	1.16%	+0.06%



# Porque o Python?

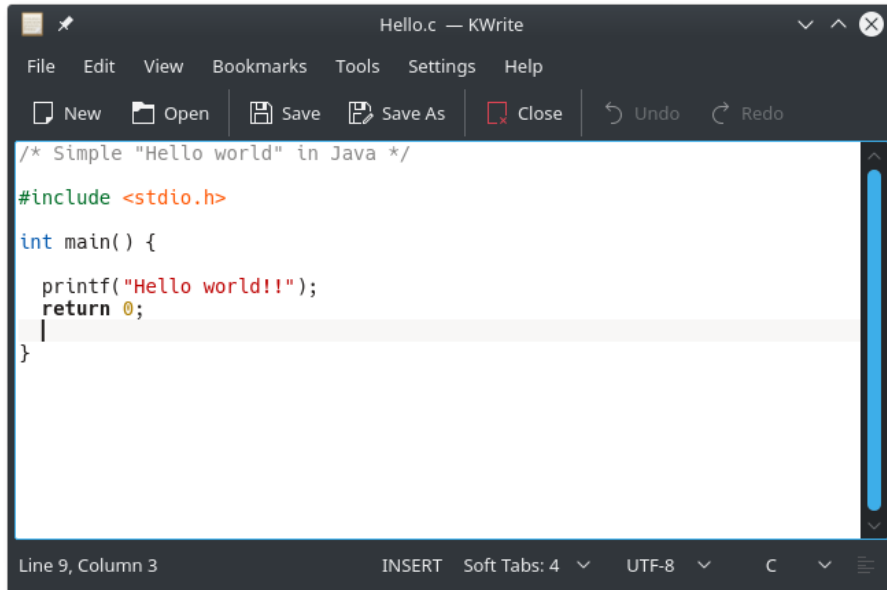
## TIOBE Index for September 2021

September Headline: Python is about to take over top position

Sep 2021	Sep 2020	Change	Programming Language	Ratings	Change
1	1		 C	11.83%	-4.12%
2	3	▲	 Python	11.67%	+1.20%
3	2	▼	 Java	11.12%	-2.37%
4	4		 C++	7.13%	+0.01%
5	5		 C#	5.78%	+1.20%
6	6		 Visual Basic	4.62%	+0.50%
7	7		 JavaScript	2.55%	+0.01%
8	14	▲▲	 Assembly language	2.42%	+1.12%
9	8	▼	 PHP	1.85%	-0.64%
10	10		 SQL	1.80%	+0.04%
11	22	▲▲	 Classic Visual Basic	1.52%	+0.77%
12	17	▲▲	 Groovy	1.46%	+0.48%

# Porque o Python?

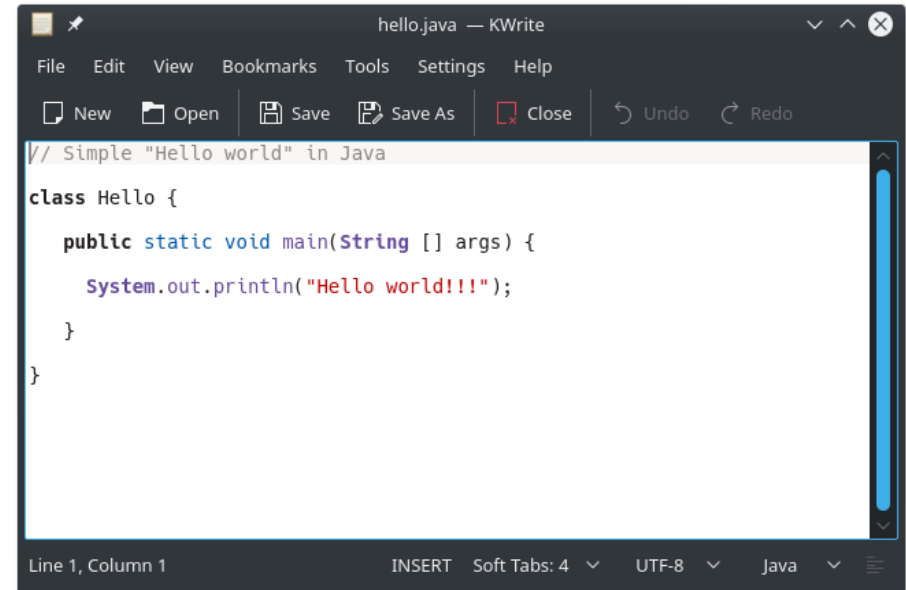
- Comparado com:
  - C é mais rápido, mas muito mais difícil de dominar
  - Java é mais ou menos igual de rápido, mas mais difícil de utilizar, e verboso.



The screenshot shows a code editor window titled "Hello.c — KWrite". The menu bar includes File, Edit, View, Bookmarks, Tools, Settings, and Help. The toolbar contains icons for New, Open, Save, Save As, Close, Undo, and Redo. The code content is as follows:

```
/* Simple "Hello world" in Java */  
  
#include <stdio.h>  
  
int main() {  
    printf("Hello world!!");  
    return 0;  
}
```

The status bar at the bottom indicates "Line 9, Column 3", "INSERT", "Soft Tabs: 4", "UTF-8", and "C".

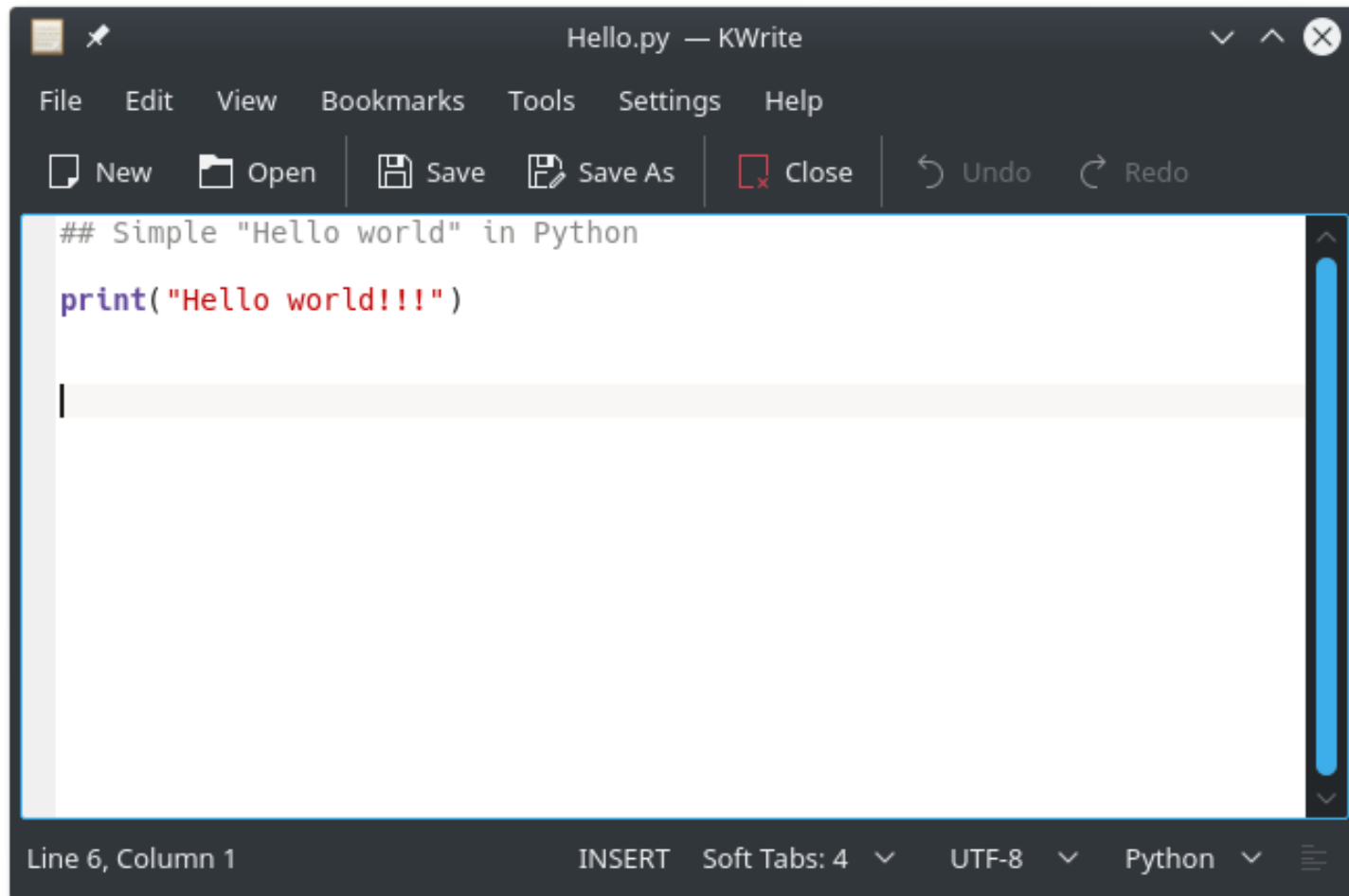


The screenshot shows a code editor window titled "hello.java — KWrite". The menu bar includes File, Edit, View, Bookmarks, Tools, Settings, and Help. The toolbar contains icons for New, Open, Save, Save As, Close, Undo, and Redo. The code content is as follows:

```
// Simple "Hello world" in Java  
  
class Hello {  
    public static void main(String [] args) {  
        System.out.println("Hello world!!!");  
    }  
}
```

The status bar at the bottom indicates "Line 1, Column 1", "INSERT", "Soft Tabs: 4", "UTF-8", and "Java".

# Porque o Python?



The image shows a screenshot of a KWrite text editor window. The window title is "Hello.py — KWrite". The menu bar includes "File", "Edit", "View", "Bookmarks", "Tools", "Settings", and "Help". The toolbar contains icons for "New", "Open", "Save", "Save As", "Close", "Undo", and "Redo". The main text area contains the following Python code:

```
## Simple "Hello world" in Python  
print("Hello world!!!")
```

The cursor is positioned at the end of the second line. The status bar at the bottom shows "Line 6, Column 1", "INSERT", "Soft Tabs: 4", "UTF-8", and "Python".

# The Python Programming language

<https://www.python.org>

  Socialize

- About
- Downloads
- Documentation
- Community
- Success Stories
- News
- Events

```
# Python 3: Simple output (with Unicode)
>>> print("Hello, I'm Python!")
Hello, I'm Python!

# Input, assignment
>>> name = input('What is your name?\n')
>>> print('Hi, %s.' % name)
What is your name?
Python
Hi, Python.
```



## Quick & Easy to Learn

Experienced programmers in any other language can pick up Python very quickly, and beginners find the clean syntax and indentation structure easy to learn. [Whet your appetite](#) with our Python 3 overview.

- 1
- 2
- 3
- 4
- 5

Python is a programming language that lets you work quickly and integrate systems more effectively. [>>> Learn More](#)

[https://en.wikipedia.org/wiki/Python \(programming language\)](https://en.wikipedia.org/wiki/Python_(programming_language))

<https://docs.python.org/3/>

# Python: Interpreter

```
2. ist90700@sigma02.ist.utl.pt: /afs/ist.utl.pt/users/0/0/ist90700 (ssh)
alberto@caprica ~ $ ssh sigma.tecnico.ulisboa.pt -l ist90700
Password:
Linux sigma02.ist.utl.pt 4.9.0-8-amd64 #1 SMP Debian 4.9.110-3+deb9u4 (2018-08-21) x86_64
#####

 Bem vindo ao cluster sigma.

Informacoes : https://dsi.tecnico.ulisboa.pt/
Suporte     : dsi@tecnico.ulisboa.pt

##### v6.4n
ist90700@sigma02:~$ python
Python 2.7.13 (default, Nov 24 2017, 17:33:09)
[GCC 6.3.0 20170516] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
ist90700@sigma02:~$ python3
Python 3.5.3 (default, Jan 19 2017, 14:11:04)
[GCC 6.3.0 20170118] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> |
```

- Localmente:
  - Utilizando o terminal em sistemas Unix-like
- Remotamente por **ssh**:
  - Ativar serviços de **afs** e **shell** no [self service da DSI](#).
  - Necessário para acesso aos computadores do laboratório
- Atenção à versão de Python ( $\geq 3.6$ )
- Interpretador: *interativo vs script*

# PyCharm: Python IDE ++

<https://www.jetbrains.com/pycharm/>

**JET BRAINS**

Tools Languages Solutions Support Store

PyCharm Coming in 2018.3 What's New Features Docs & Demos Buy [Download](#)

**PC** **PyCharm**

Python IDE  
for Professional Developers

[DOWNLOAD NOW](#)

Full-fledged Professional or Free Community

# WING: Python IDE

<https://wingware.com>

# WING PYTHON IDE

The INTELLIGENT DEVELOPMENT ENVIRONMENT for PYTHON PROGRAMMERS

## Wing 101 - Version 6.1.0 - Released 2018-07-30

The best Python IDE. And I have tried them all! -- *Ahmed Ali*

Wing 101 is a very simple free Python IDE designed for teaching beginning programmers. It omits most features found in Wing Pro. [Compare Products](#)

If you are new to programming, check out the book [Python Programming Fundamentals](#) and accompanying screen casts, which use Wing IDE 101 to teach programming with Python.

Wing 101 is free to use for any purpose and does not require a license to run.

- Tutorial
- Quick Start Guide
- What's New

**Also Available:** [Patches](#)

**Other OSes:** [Windows](#) [Linux 64-bit](#)

**Other Versions:** [7.0.0.0-alpha](#) [6.0.12](#) [5.1.12](#) [5.0.9](#) [4.1.14](#) [3.2.13](#) [all versions](#)

**Other Products:** [Wing Pro](#) [Wing Personal](#) — [Compare Product Features](#)

## Download Wing 101:

OS X Installer  
64-bit Intel

SHA1: [afacfecb9ac07d2202692ad8e613640638d1ff36](#)

[Supported OSes](#)

[Supported Python Versions](#)

[Change Log](#)

<b>Manual</b>	<a href="#">HTML</a>	<a href="#">US Letter</a>	<a href="#">A4</a>
<b>How-Tos</b>	<a href="#">HTML</a>	<a href="#">US Letter</a>	<a href="#">A4</a>
<b>Tutorial</b>	<a href="#">HTML</a>	<a href="#">US Letter</a>	<a href="#">A4</a>

# iPython & Jupyter: Advanced interpreter and interactive notebooks

<http://jupyter.org>



```
pip3 install jupyter
```

**IP**[y]: IPython  
Interactive Computing



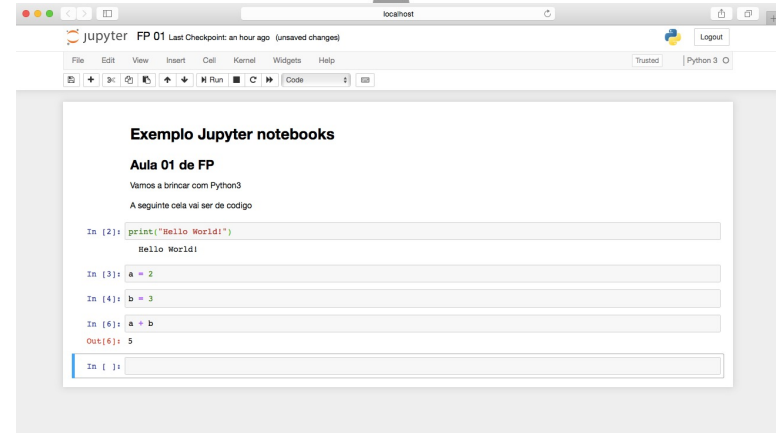
```
1. IPython: Users/alberto (Python)
alberto@DemosAir ~ $ ipython
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 26 2018, 23:26:24)
Type 'copyright', 'credits' or 'license' for more information
IPython 6.5.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]:

In [1]:

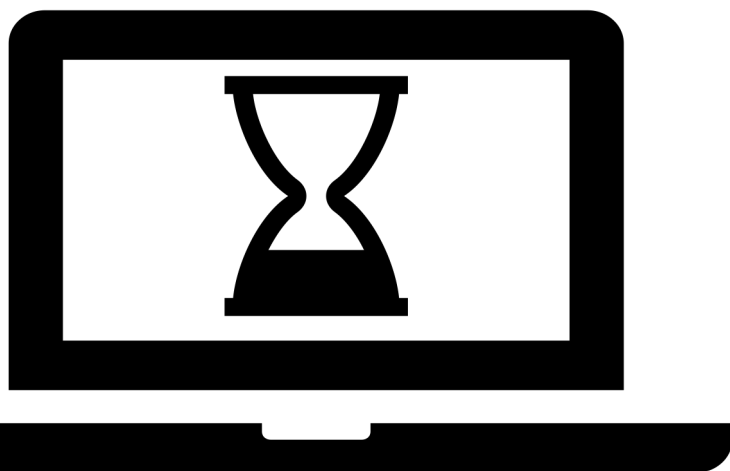
In [1]: print("Hello World!")
Hello World!

In [2]:
```





# Tempo de prática: Tarefas



- Explorar site da cadeira
- Registrar no Slack e deixar uma mensagem (#random)
- Instalar Python3
- Abrir um interpretador
- “Olá mundo”
- Instalar um IDE (PyCharm ou WingIDE recomendado)
- Instalar Jupyter notebooks
- ...

# Tarefas antes da próxima aula

- Completar as tarefas do tempo de prática de hoje
- Ler o Capítulo 1 do livro da disciplina

