

ORGANIZAÇÃO DE PROGRAMAS EM PYTHON

PROGRAMAÇÃO APLICADA A MATEMÁTICA

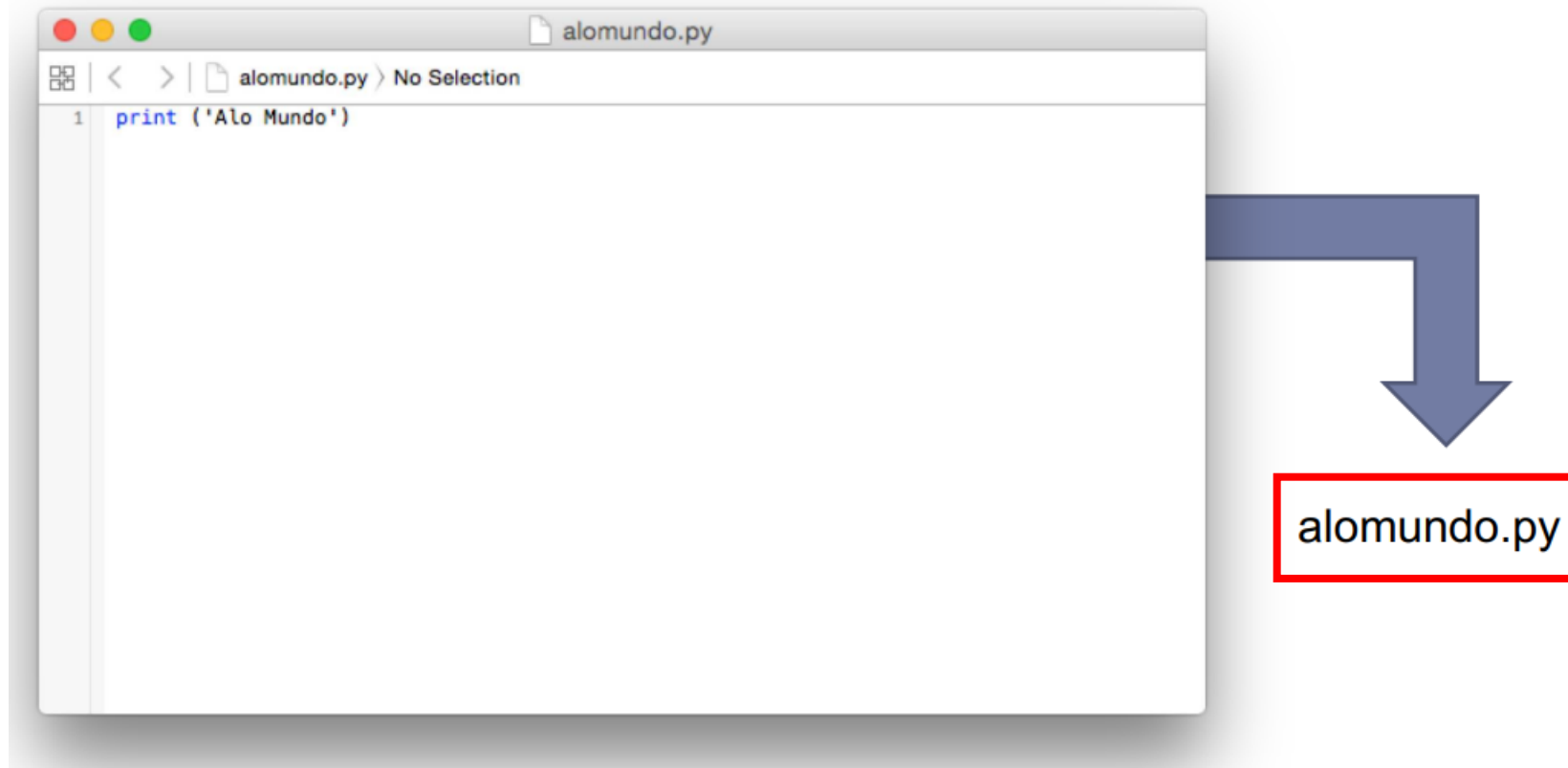
Meirylene Avelino

meirylenerea@id.uff.br

VAMOS PROGRAMAR EM PYTHON! MAS...

- Como um programa é organizado?
- Quais são os tipos de dados disponíveis?
- Como variáveis podem ser declaradas?
- Como atribuir valores às variáveis?
- Como entrada e saída básica de dados podem ser feitas?

PRIMEIRO PASSO: ESCREVER O PROGRAMA!

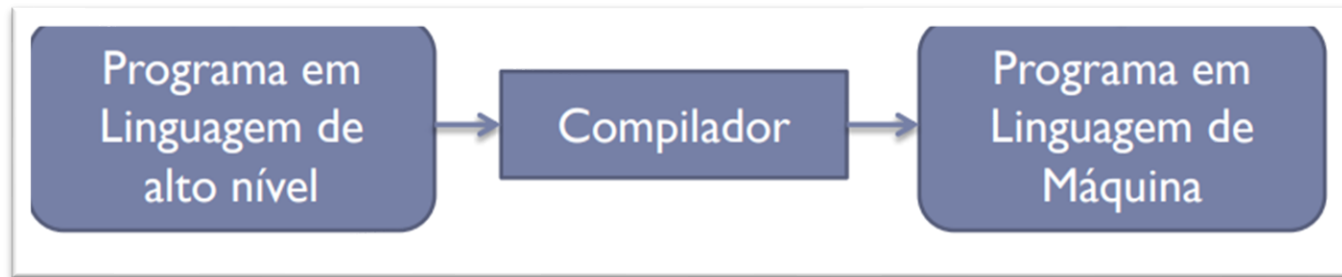


MAS O COMPUTADOR NÃO CONHECE PYTHON!!!

- O computador só entende binário
 - Linguagem de zeros e uns
 - 010010011101010101001010101
- Precisamos traduzir o programa Python para binário

COMPILAÇÃO

- Na maioria das linguagens, antes de executar um programa, é necessário compilar o programa
- O compilador gera um arquivo “executável”
 - Esse novo arquivo é o que será de fato executado



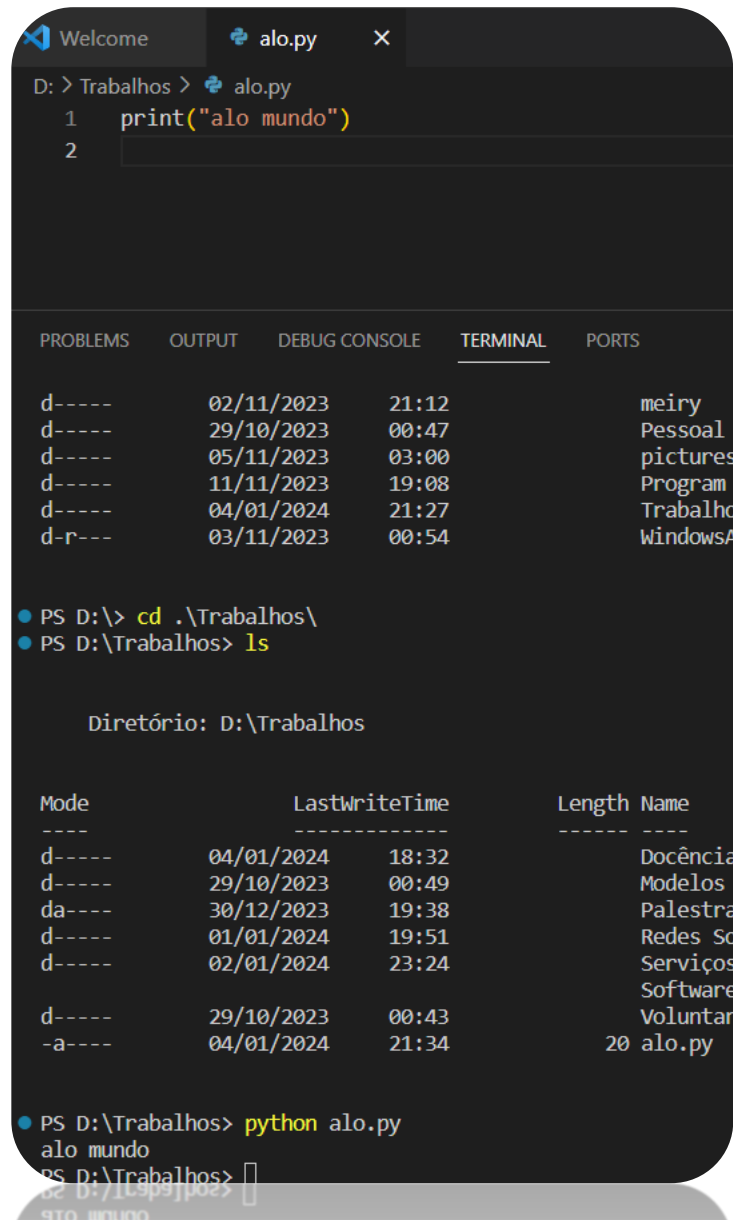
INSTALAÇÃO DO INTERPRETADOR PYTHON

- Download do Python 3.12
- IMPORTANTE HABILITAR A OPÇÃO DE ADD PATH

<https://www.python.org/downloads/>



EXECUÇÃO



```
Welcome | alo.py x
D: > Trabalhos > alo.py
1 print("alo mundo")
2

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

d----- 02/11/2023 21:12 meiry
d----- 29/10/2023 00:47 Pessoal
d----- 05/11/2023 03:00 pictures
d----- 11/11/2023 19:08 Program
d----- 04/01/2024 21:27 Trabalho
d-r--- 03/11/2023 00:54 WindowsA

PS D:\> cd .\Trabalhos\
PS D:\Trabalhos> ls

Diretório: D:\Trabalhos

Mode                LastWriteTime         Length Name
----                -
d-----          04/01/2024         18:32     Docência
d-----          29/10/2023          00:49     Modelos
da----          30/12/2023          19:38     Palestra
d-----          01/01/2024          19:51     Redes So
d-----          02/01/2024          23:24     Serviços
          Software
d-----          29/10/2023          00:43     Voluntar
-a----          04/01/2024          21:34      20 alo.py

PS D:\Trabalhos> python alo.py
alo mundo
PS D:\Trabalhos>
```

VAMOS FAZER
JUNTOS?

NOTEPAD X IDE

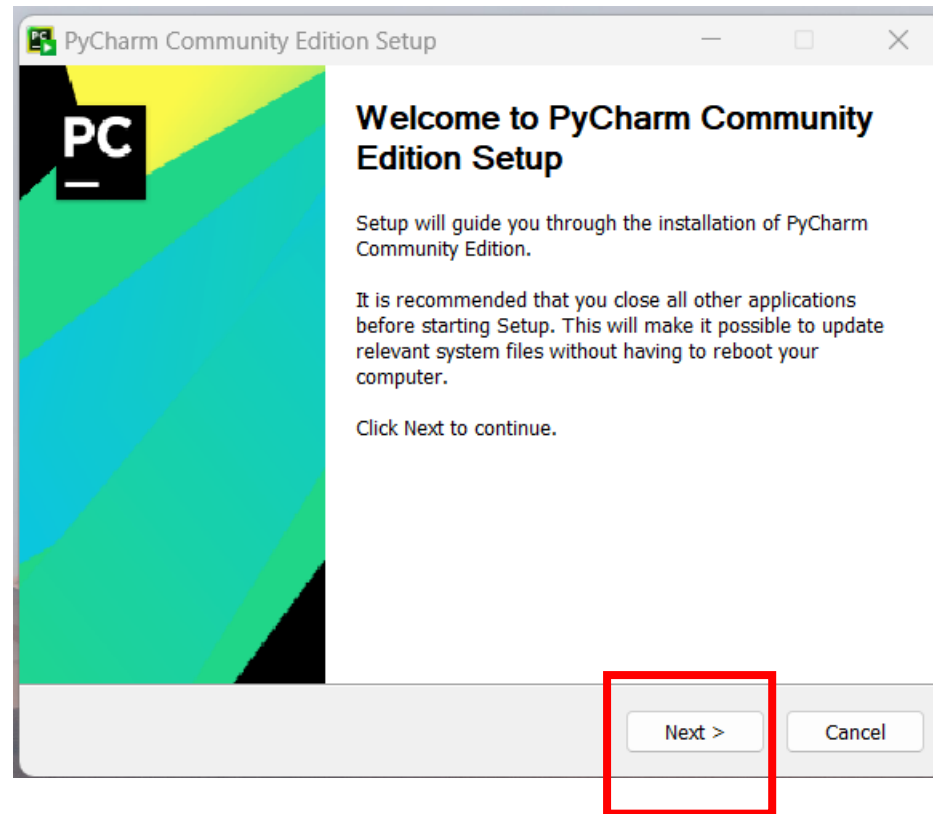
- Dificuldade do Notepad
 - Editor básico, sem ajuda para programar
 - Execução externa
- Integrated Development Environment (IDE)

INSTALAÇÃO DO PYCHARM

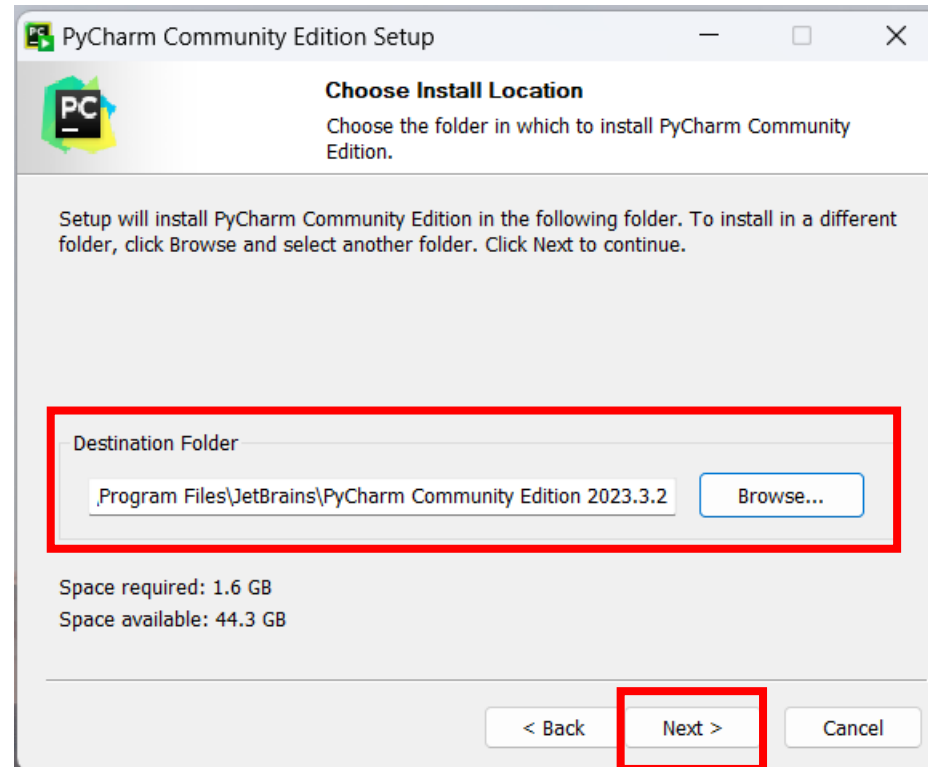
- Os alunos podem optar por qualquer outra IDE ou editor. Nas aulas, iremos usar um pouco dos dois tipos.
- Download do PyCharm



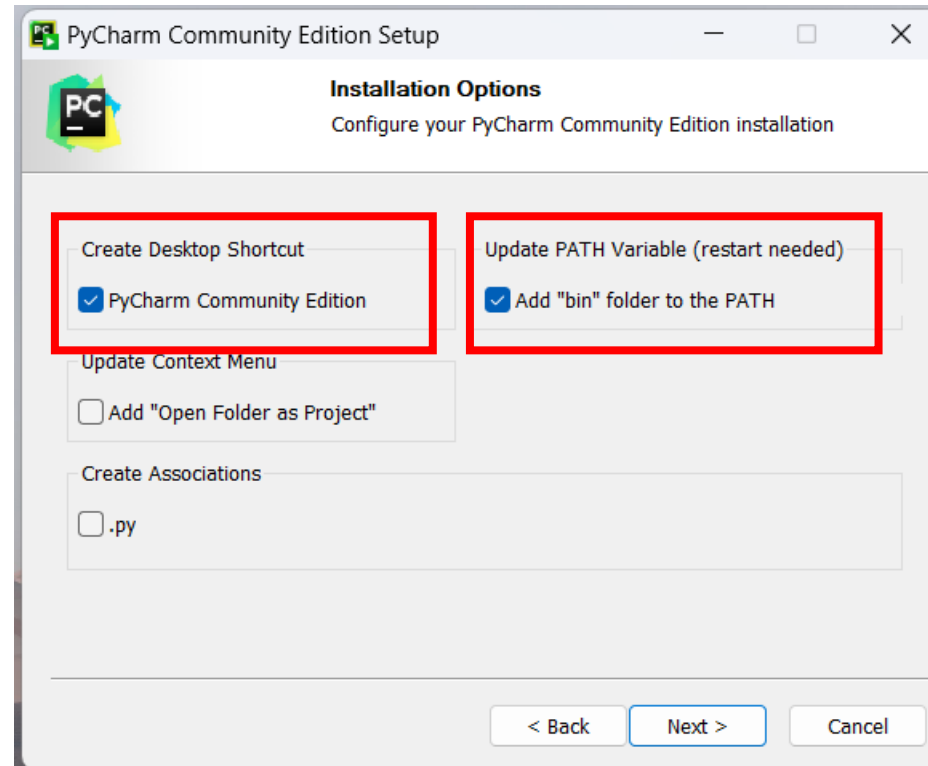
INSTALAÇÃO DO PYCHARM



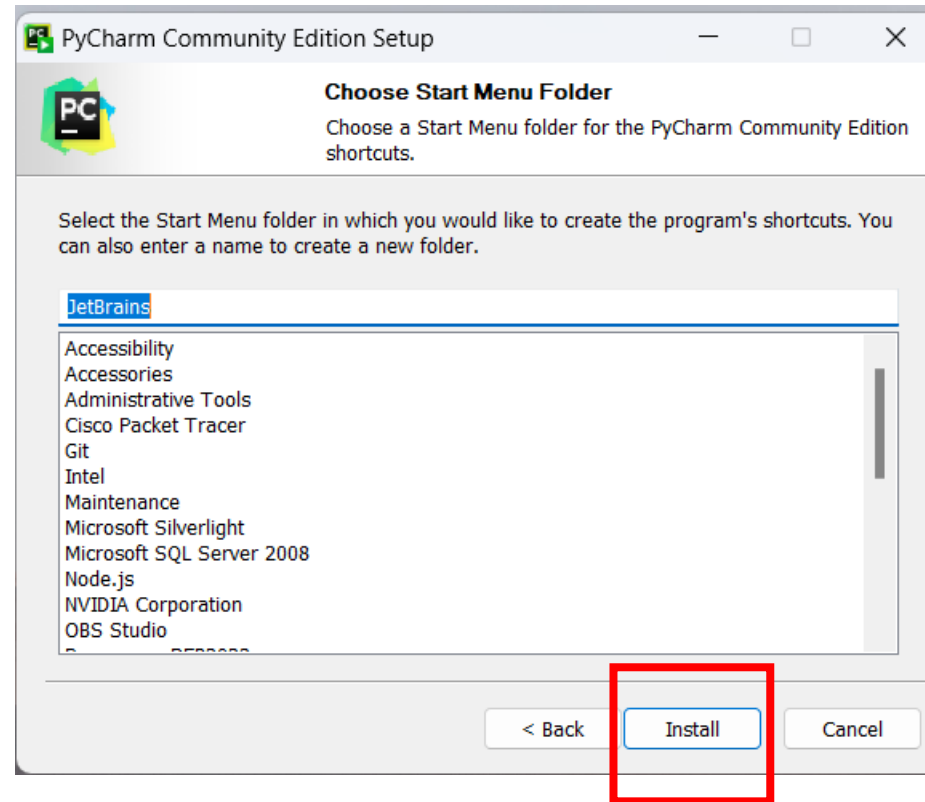
INSTALAÇÃO DO PYCHARM



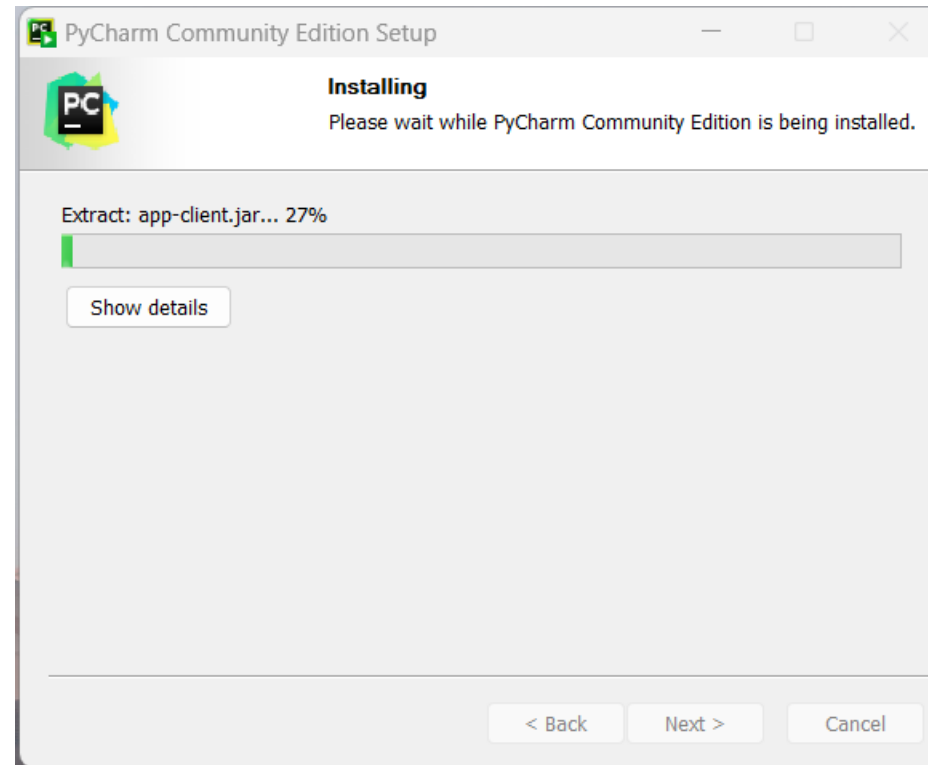
INSTALAÇÃO DO PYCHARM



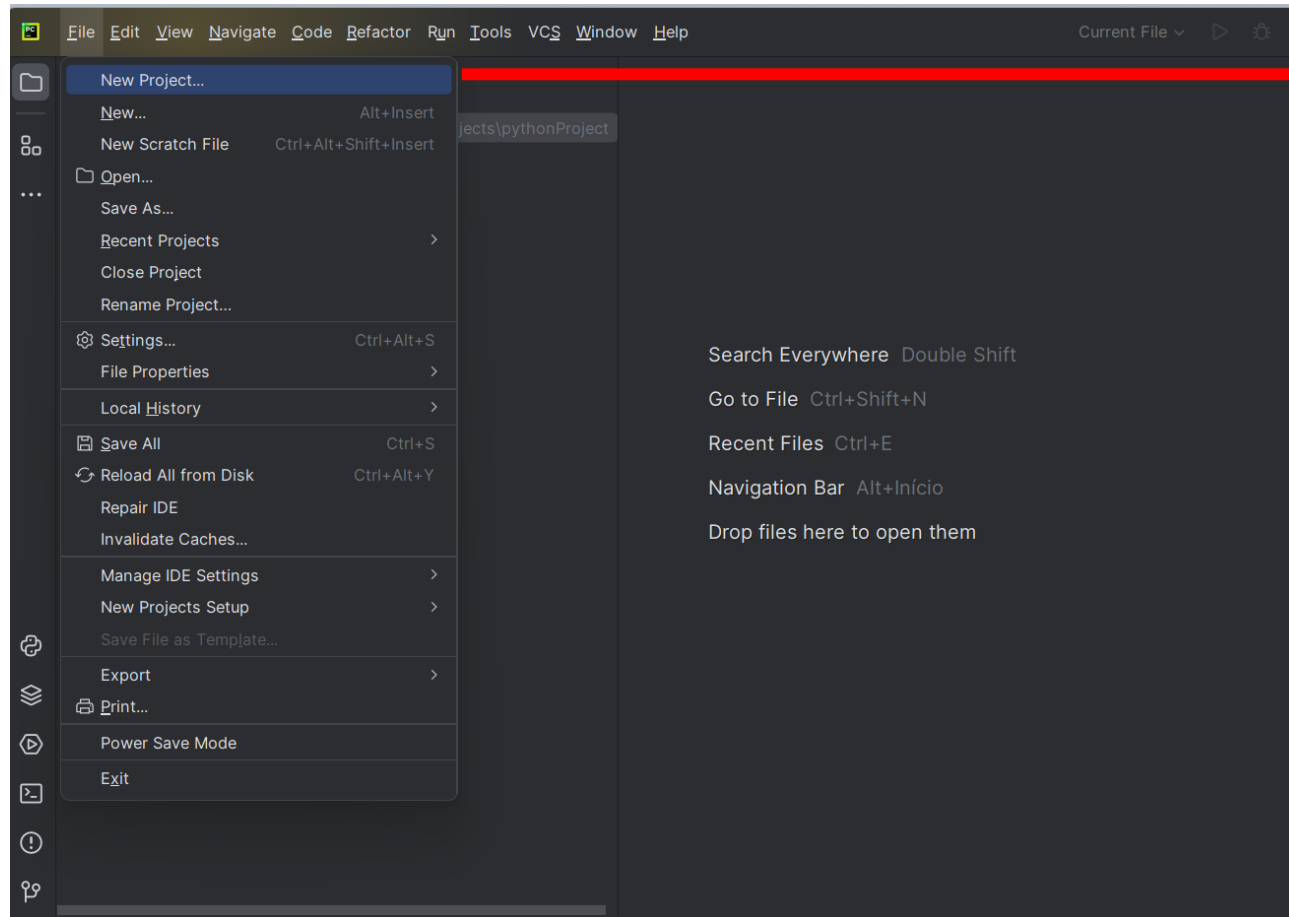
PRIMEIRO PASSO: ESCREVER O PROGRAMA!



PRIMEIRO PASSO: ESCREVER O PROGRAMA!

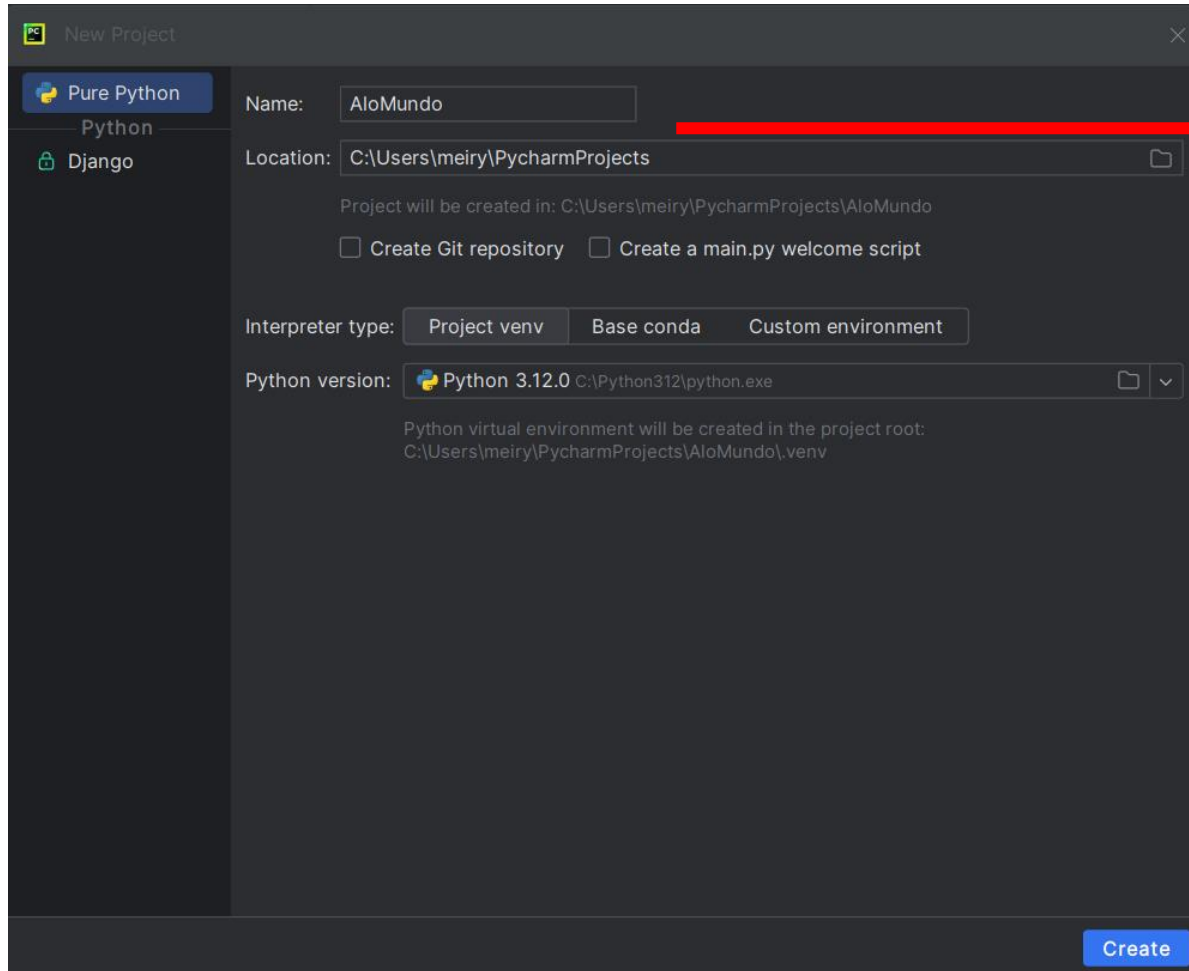


CRIANDO UM PROJETO PYCHARM



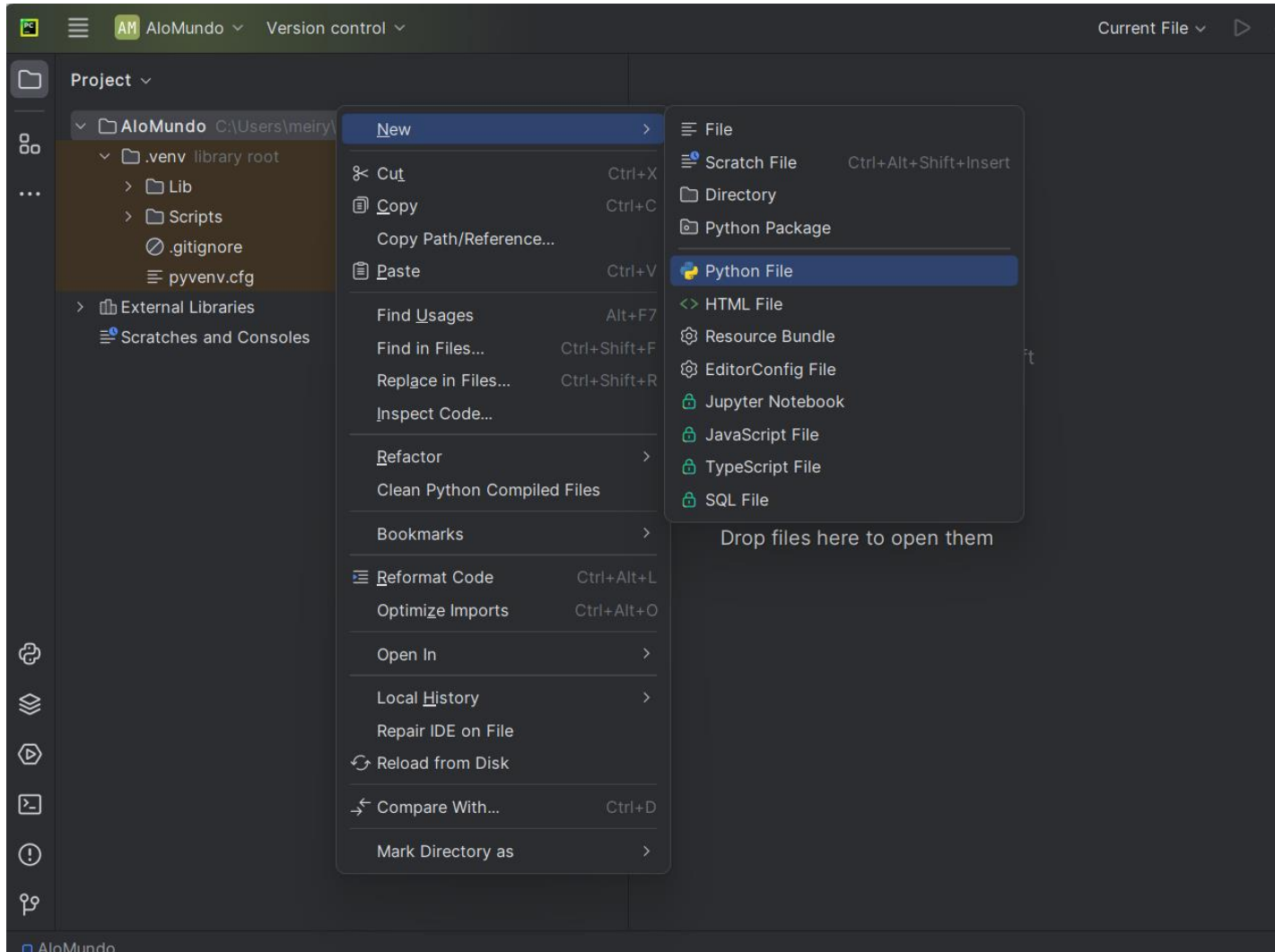
**Selecionar
New Project**

CRIANDO UM PROJETO PYCHARM



Definir o nome e local do projeto e clicar em *Create* ao final

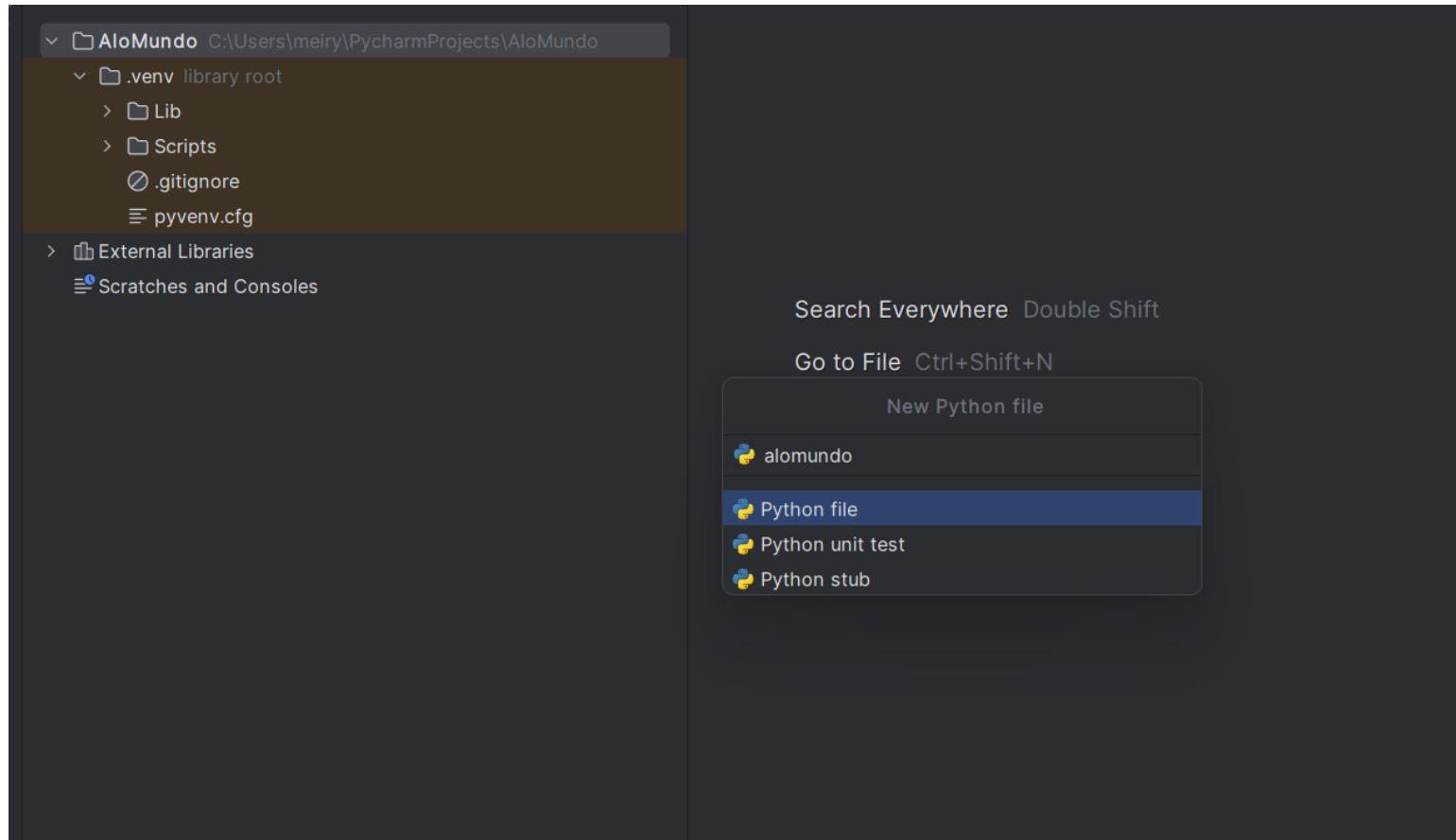
CRIANDO UM ARQUIVO PYCHARM



**Clicar com o
botão da
direita sobre
o nome do
projeto**

**Selecionar
New/Python
File**

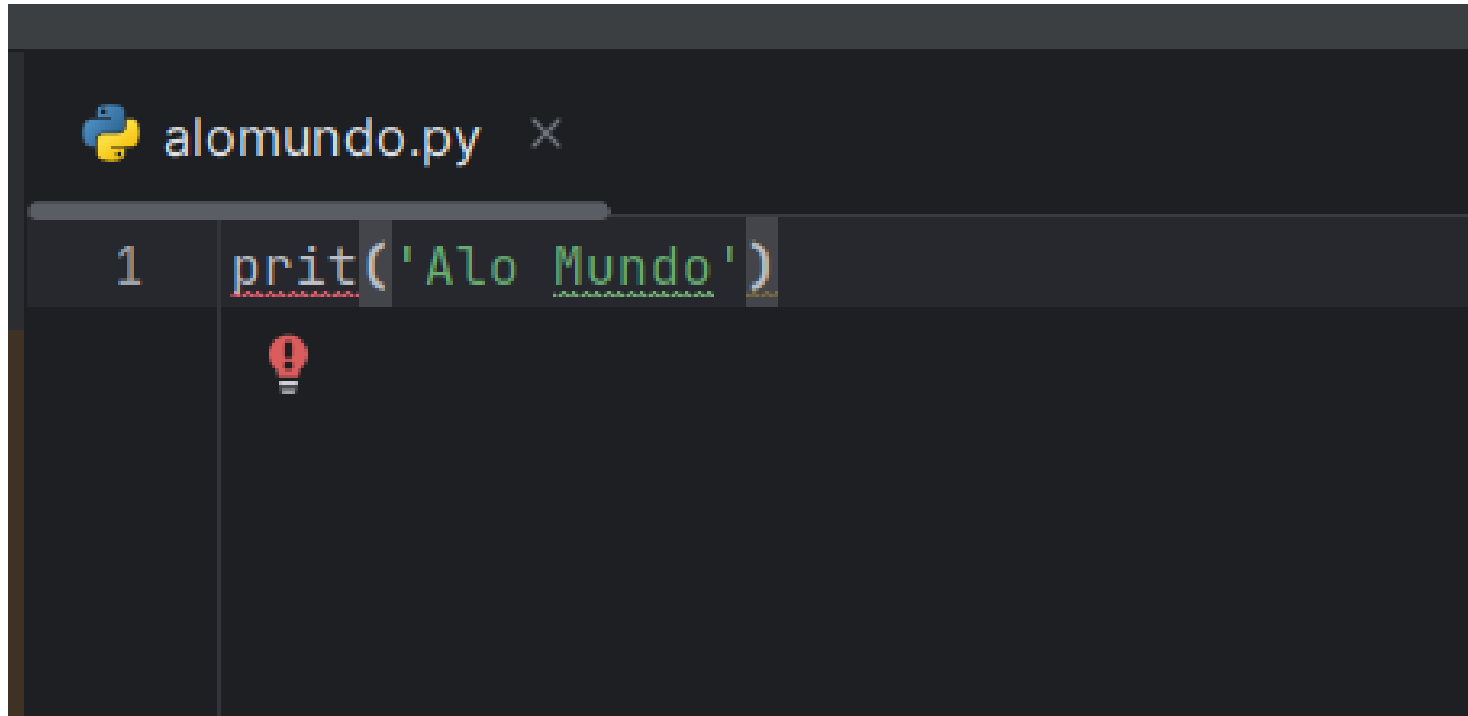
CRIANDO UM ARQUIVO PYCHARM



**Clicar com o
botão da
direita sobre
o nome do
projeto**

**Selecionar
New/Python
File**

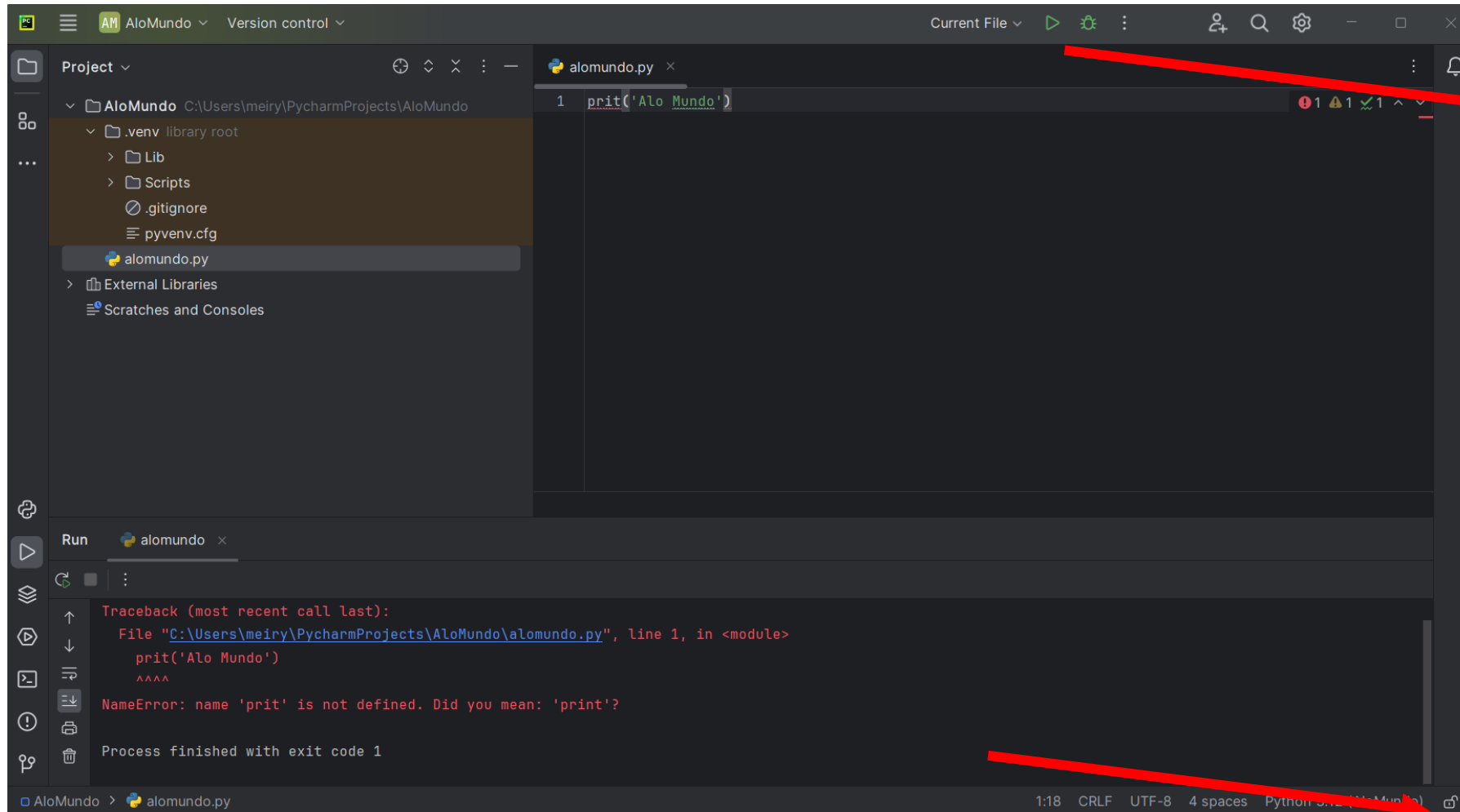
ESCREVENDO O PROGRAMA NO PYCHARM



```
1 print('Alo Mundo')
```

**Avisos sobre
erros
durante a
edição do
código**

EXECUTANDO O PROGRAMA NO PYCHARM



Clicar neste ícone para executar o programa

No painel inferior ocorrerá a entrada e saída de dados

INSTALAÇÃO DO VISUAL STUDIO CODE

“ Notepad boladão ”



<https://code.visualstudio.com/>

ROTEIRO DE INSTALAÇÃO

**Python
IDE ou “editor de texto”**

REGRAS BÁSICAS

Sequência dos comandos é importante

Blocos devem ser criados usando endentação (com espaços ou tab)

COMENTÁRIOS

- Comentários são trechos do programa voltados para a leitura por humanos, e ignorados pelo interpretador
- Começam com o símbolo `#`
 - Tudo na linha após `#` é ignorado pelo interpretador
- Use comentários para documentar seu código e fazer com que ele seja fácil de entender por outras pessoas

ATRIBUIÇÃO DE VALORES

- Em Python, o operador de igualdade (=) é usado para atribuir valores às **variáveis** (são as caixas utilizadas como exemplos anteriores!)
- É equivalente ao símbolo de atribuição ← que usávamos no pseudocódigo
- Sempre na forma: **variável = valor ou expressão**
 - A expressão do lado direito é processada
 - O valor gerado é atribuído à variável

EXEMPLO DE PROGRAMA EM PYTHON

```
# Este programa calcula a area de um triangulo retangulo
altura = 15
base = 3
area = (altura * base) / 2
print(area)
```

QUAIS SÃO OS TIPOS DE DADOS DISPONÍVEIS?

Em Python, toda variável tem um tipo

Com isso, o computador pode saber quais operações são permitidas

Os tipos podem ser divididos em três grupos

- Tipos numéricos (inteiro, float, ...)
- Tipos textuais (caractere e string)
- Tipo lógico (booleano)

Os tipos são definidos dinamicamente, pelo próprio Python

- Não é preciso dizer de que tipo é cada variável

EXEMPLO DE VARIÁVEIS LÓGICAS(BOOLEAN)

x = True

y = False

EXEMPLO DE VARIÁVEIS TEXTUAIS (STRING)

nome = 'Maria'

sobrenome = "Silva"

letra = 'A'

texto = 'Alo Mundo'

EXEMPLO DE VARIÁVEIS NUMÉRICAS

$a = -5$

$b = 10$

$c = 200$

$d = -12312312$

$e = 345092834$

$f = 2.5$

$g = 0.6023$

$e24$

$h = 0.4e-3$

TIPAGEM DINÂMICA

```
a = -5      ➔ inteiro
b = 10      ➔ inteiro
c = 200     ➔ inteiro
d = -12312312 ➔ inteiro
e = 345092834 ➔ inteiro
f = 2.5      ➔ float
g = 0.6023e24 ➔ float
h = 0.4e-3   ➔ float
```

Tipo é
determinado
automaticamente
pelo Python no
momento de
criação da
variável

TIPAGEM FORTE

Uma vez que uma variável tenha um valor de um tipo, ele não pode ser usado como se fosse de outro tipo.

Exemplo:

```
a = 10  
b = '20'  
c = a + b
```

```
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
TypeError: unsupported operand type(s) for  
+: 'int' and 'str'
```


REGRAS PARA NOMES DE VARIÁVEIS

Os nomes de variáveis devem respeitar algumas regras

- São sensíveis a caixa
- Podem ter tamanho ilimitado (mas evite abusos)
- Devem começar com letra ou underline (_)
- Outros caracteres podem ser letras, números ou underline
- Não podem ter espaço nem acentos
- Não podem ser uma palavra reservada da linguagem

ENTRADA DE DADOS

Para entrada de dados, usamos **input**

É possível informar um texto que aparecerá impresso na tela para que o usuário saiba que o programa está esperando a entrada de um valor

```
nome = input('Digite o nome do aluno: ')\nprint(nome)
```

INPUT LÊ DADOS COMO STRING

Você pode usar o comando `type` para saber o tipo que o Python atribuiu a uma variável

```
altura = input('Digite a altura do triangulo: ')\nprint(type(altura))\nbase = input('Digite a base do triangulo: ')\nprint(type(base))\n...
```

MUDANÇA DE TIPO

Usar `int()`, `float()` ou `eval()` para fazer o Python ler variáveis de tipo numérico

```
altura = int(input('Digite a altura do triangulo: '))
print(type(altura))
base = int(input('Digite a base do triangulo: '))
print(type(base))
area = (base * altura)/2
print('A area do triangulo eh: ', area)
```

SAÍDA DE DADOS

Para saída de dados, usamos `print`

EXEMPLO DE ENTRADA E SAÍDA DE DADOS

```
print('Niterói é muito legal')
```

```
print(123)
```

```
altura = 10
```

```
print(altura)
```

```
print('Vamos pular uma linha \n')
```

```
print('O nome do aluno eh', nome)
```

VOLTANDO AO EXEMPLO DE PROGRAMA EM PYTHON

```
altura = int(input('Digite a altura do triangulo: '))  
base = int(input('Digite a base do triangulo: '))  
area = (base * altura)/2  
print('A area do triangulo eh:', area)
```

FORMATAÇÃO DE NÚMEROS

- É possível especificar uma máscara no comando print para imprimir números com um determinado formato
- Pode-se, por exemplo, fazer com que um float seja impresso com apenas duas casas decimais
- `print("%.2f" % variável)`
 - f é usado para números do tipo float
 - d é usado para números inteiros
 - s é usado para strings

VOLTANDO AO EXEMPLO DE PROGRAMA EM PYTHON

```
altura = int(input('Digite a altura do triangulo: '))
base = int(input('Digite a base do triangulo: '))
area = (base_* altura)/2
print('Altura = %4d' % altura)
print('Base = %4d' % base)
print('A area do triangulo eh %.2f' % area)
```

IMPRIMINDO VÁRIAS VARIÁVEIS AO MESMO TEMPO

```
altura = int(input('Digite a altura do triangulo: '))
base = int(input('Digite a base do triangulo: '))
area = (base * altura)/2
tipo = "retangulo"
print('A area do triangulo %s de altura %.0f e base %.0f
      eh: %.2f' % (tipo, altura, base, area))
```

ORGANIZAÇÃO DE PROGRAMAS EM PYTHON

PROGRAMAÇÃO APLICADA A MATEMÁTICA

Meirylene Avelino

meirylenerea@id.uff.br