Engenharia de Computação

Lógica e Programação de Computadores

Victor Machado da Silva, MSc
victor.silva@professores.ibmec.edu.br

Ibmec
PROTAGONISTAS PARA O MUNDO

Índice

Ibmec
PROTAGONISTAS PARA O MUNDO

Apresentação do curso

Apresentação do curso

- Contato: victor.silva@professores.ibmec.edu.br
- Aulas às segundas e quartas-feiras, de 9:49 às 11:50
- Grupo no Whatsapp: https://chat.whatsapp.com/Bb5EKrfEIG7AqENWjqh5Uc
- Material no GitHub: https://github.com/victor0machado/2020.2-logprog

Ibmec
PROTAGONISTAS PARA O MUNDO

Apresentação do curso

Sem.	Data	Tópico
01	10/08	Introdução à disciplina
01	12/08	Revisão de Python: tipos de dados, listas, dicionários, manipulação de arquivos, decorador @timeit
02	17/08	Algoritmos: sintaxe em pseudocódigo, recursividade
02	19/08	Algoritmos: complexidade de algoritmos, notação O, algoritmos ótimos
03	24/08	Algoritmos: exercícios
03	26/08	Listas lineares: alocação, busca, busca binária, remoção
04	31/08	Listas lineares: pilhas e filas – inserção e remoção, alocação encadeada, listas circulares
04	02/09	Listas lineares: exercícios
05	07/09	SEM AULA (7 DE SETEMBRO)
05	09/09	Árvores: definições e representações básicas, árvores binárias, percurso em árvores binárias
06	14/09	Árvores: conversão de uma floresta, árvores com costura
06	16/09	Árvores: exercícios
07	21/09	Árvores binárias de busca: conceitos, busca e inserção; árvore de partilha
07	23/09	Árvores binárias de busca: exercícios
08	28/09	Atividades em grupo / dúvidas e discussões
08	30/09	Atividades em grupo / dúvidas e discussões
09	05/10	Atividades em grupo / dúvidas e discussões
09	07/10	**P1**
10	12/10	SEM AULA (N. SRA. APARECIDA)

Ibmec
PROTAGONISTAS PARA O MUNDO

Apresentação do curso

Sem.	Data	Tópico
10	14/10	Árvores balanceadas: árvores AVL, árvores graduadas e rubro-negras, árvores B
11	19/10	Árvores balanceadas: exercícios
11	21/10	Listas de prioridades: implementação, alteração de prioridades, máximos e mínimos
12	26/10	Listas de prioridades: exercícios
12	28/10	Algoritmos de ordenação: ordenação bolha, por inserção e por intercalação
13	02/11	SEM AULA (FINADOS)
13	04/11	Algoritmos de ordenação: ordenação rápida, ordenação em heap, árvore de decisão
14	09/11	Algoritmos de ordenação: exercícios
14	11/11	Grafos: definições, representação gráfica, terminologia, caminhos e ciclos
15	16/11	Grafos: representação como estruturas de dados, métodos para percurso em grafos
15	18/11	Grafos: exercícios
16	23/11	Atividades em grupo / dúvidas e discussões
16	25/11	Atividades em grupo / dúvidas e discussões
17	30/11	Atividades em grupo / dúvidas e discussões
17	02/12	**P2**
18	07/12	SEM AULA
18	09/12	Atividade em grupo / dúvidas
19	14/12	**PS**
19	16/12	SEM AULA
20	21/12	SEM AULA
20	23/12	SEM AULA

Ibmec
PROTAGONISTAS PARA O MUNDO

Apresentação do curso

Avaliação

- Proporção:
  - Exercícios periódicos (AC): 20%
  - Projeto (AP1): 40%
  - Projeto (AP2): 40%
- Detalhes das entregas:
  - Exercícios da AC são individuais
  - Projetos de AP1 e AP2 em grupos de no mínimo 2 e no máximo 3 pessoas
  - Entrega via Integrees
- AS será uma prova com consulta, que substituirá a menor nota entre AP1 e AP2.

Ibmec
PROTAGONISTAS PARA O MUNDO

Sugestões de materiais para estudo

- Python é uma linguagem intuitiva para o aprendizado, porém é importante termos à mão livros, apostilas e outros materiais para auxiliar os estudos. Abaixo encontram-se algumas sugestões:
  - Documentação oficial em Python: https://docs.python.org/pt-br/3/index.html
  - Jayme Luiz Szwarcfiter, Lilian Markenzon – Estruturas de Dados e seus Algoritmos (LTC)
  - Dilermando Piva Junior et al – Estruturas de Dados e Técnicas de Programação (Campus)
  - Stack Overflow: https://stackoverflow.com/questions/tagged/python
  - Artigos no Medium.com: https://medium.com/search?q=python

Ibmec
PROTAGONISTAS PARA O MUNDO

Sugestões de materiais para estudo

- Canais interessantes no Youtube sobre Python e programação:
  - Programação Dinâmica: https://www.youtube.com/c/ProgramacaoDinamica/
  - Curso em Vídeo: https://www.youtube.com/c/CursoemVideo/
  - Sentdex (em inglês): https://www.youtube.com/c/sentdex
  - Filipe Deschamps: https://www.youtube.com/c/FilipeDeschamps
  - DevMedia: https://www.youtube.com/c/DevmediaBrasil

Ibmec
PROTAGONISTAS PARA O MUNDO

Configurando o ambiente Python

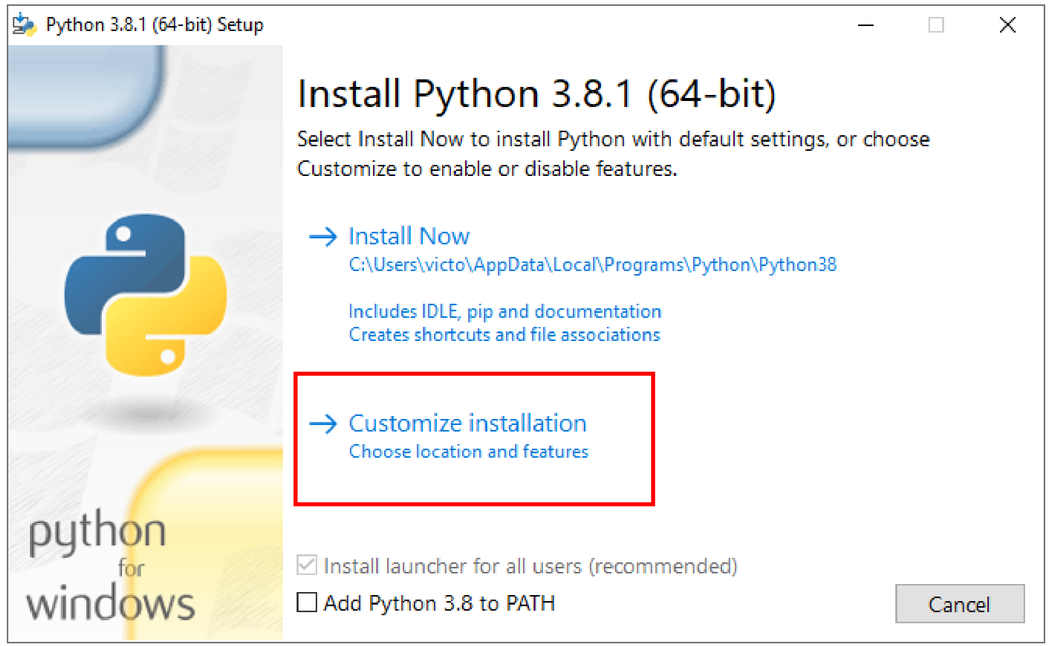
O que é Python?

- Python é uma linguagem de programação de alto nível, lançada por Guido van Rossum em 1991. Atualmente é uma das linguagens de uso mais abrangentes no mundo todo, principalmente nas áreas de Data Science e em aplicações de *back-end*, ou seja, de processamento de dados que não interagem diretamente com o usuário final.
- Diversas organizações utilizam Python atualmente:
  - Google
  - Yahoo!
  - NASA
  - AirCanada

Ibmec
PROTAGONISTAS PARA O MUNDO

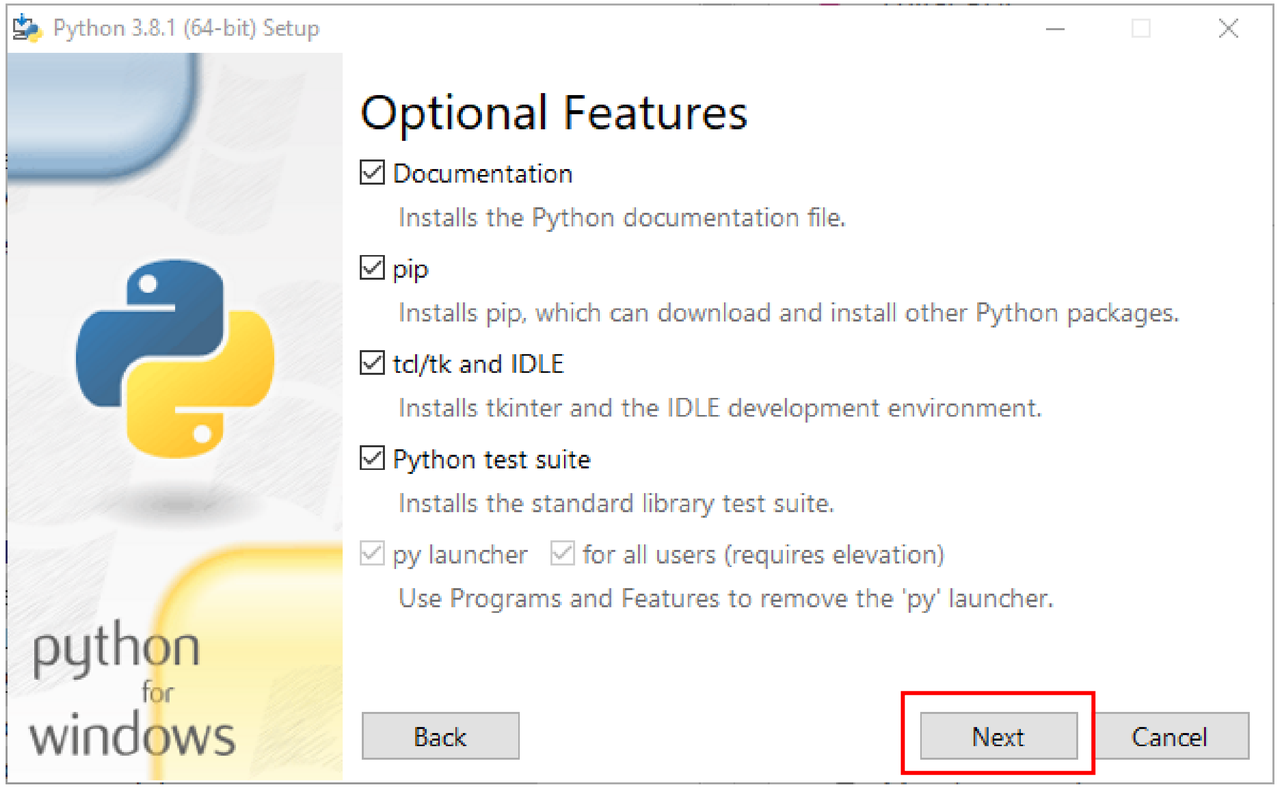
Como instalar Python?

- Neste curso podemos trabalhar com qualquer versão recente do Python, 3.7 ou superior. Para baixar o instalador para Windows, clique neste link. O download do instalador para macOS se encontra neste link.
- Este curso focará no uso do Python para Windows. Para o uso no macOS, veja no site oficial da linguagem informações particulares.
- Ao clicar no instalador, selecione a opção “Customize installation”.



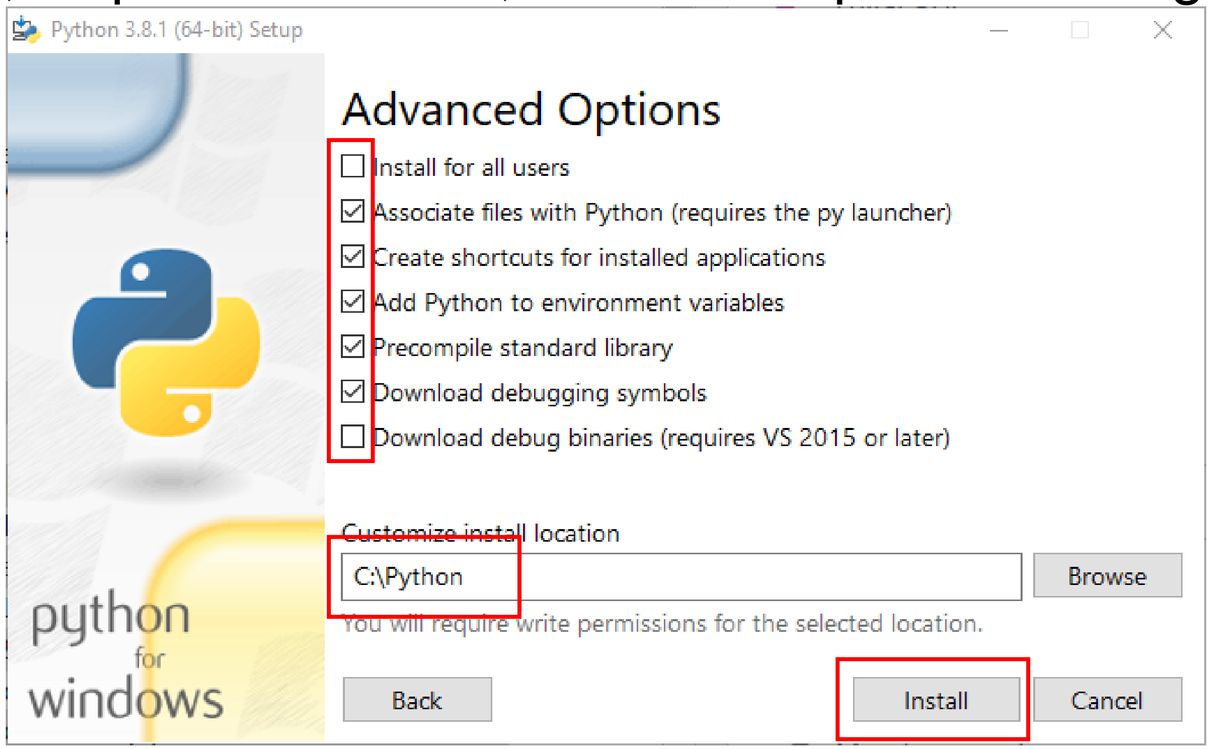
Como instalar Python?

- Na tela “Optional Features”, clique em “Next”.



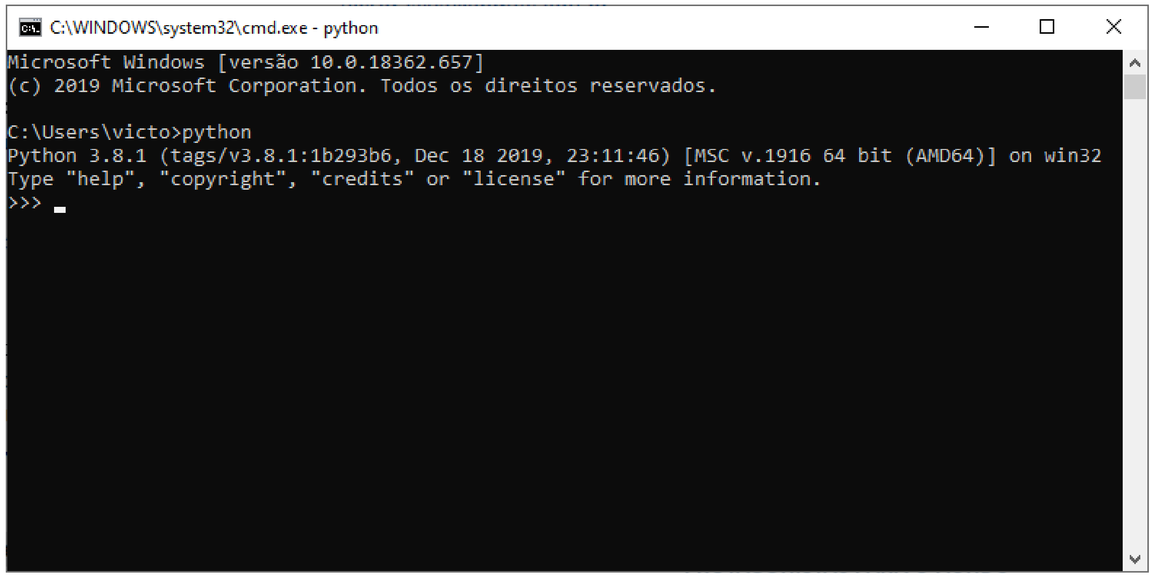
Como instalar Python?

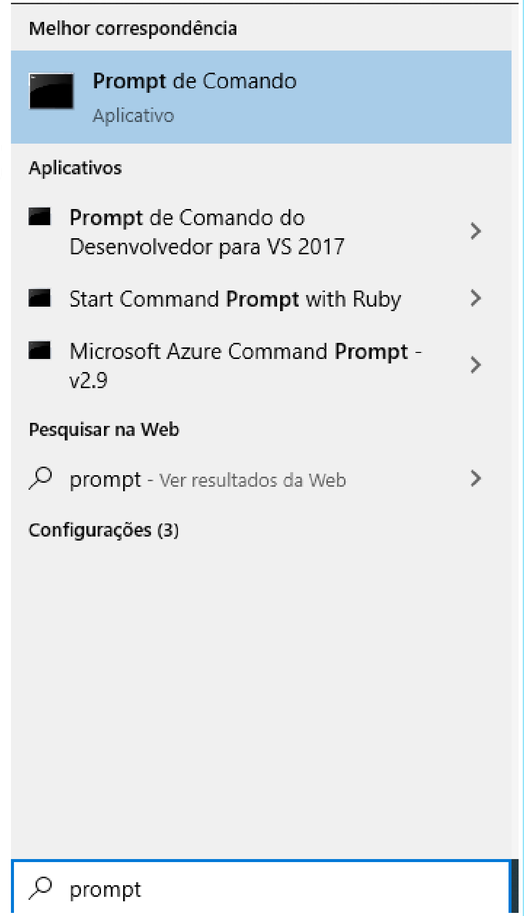
- Na tela “Advanced Options”, deixe as caixas de opções marcadas conforme a imagem abaixo.
- No campo “Customize install location”, escolha um caminho de fácil acesso. O caminho sugerido é “C:\Python”.
- Com tudo pronto, clique em “Install”, e conclua após a mensagem de sucesso.



Como instalar Python?

- Para conferir se a instalação foi bem sucedida, faça os seguintes passos:
  - Clique no botão do Windows  ;
  - Digite a caixa de pesquisa **prompt de comando**, e abra o programa;
  - Na janela que abrir, digite o comando **python** e pressione Enter;
  - O Windows deve inicializar um editor de Python na mesma janela, como mostrado abaixo.





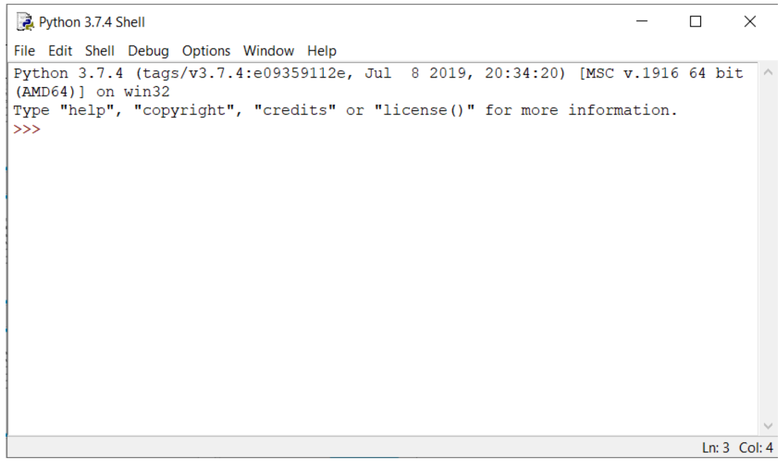
Algumas dicas iniciais após instalar o Python

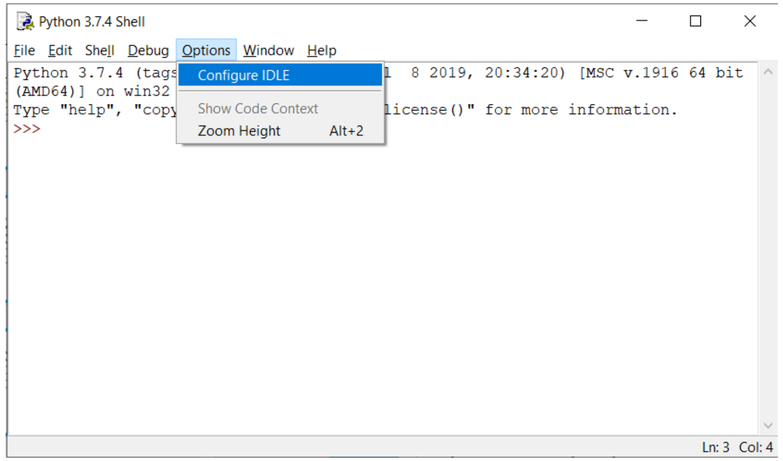
1. Antes de abrir o programa, abra o Windows Explorer, clique em **Este Computador** e, em seguida, no drive do seu computador (C: ou D:, dependendo da sua máquina);
2. Neste diretório, crie uma pasta chamada **Projetos**. Esta pasta será usada para armazenar todos os seus projetos de software;
3. Dentro da pasta de projetos, crie a pasta da disciplina (p.ex., **algoritmos**);
4. Evite utilizar caminhos muito longos (p.ex., C:\Users\12304010\Projetos\Nome-da-pessoa\Documentos\etc...) ou incluir espaços no caminhos (p.ex., C:\Victor Machado). O primeiro é muito trabalhoso para utiliza-lo recorrentemente, e o segundo pode causar alguns problemas na execução do código;
5. Sempre que criar arquivos Python, comece o nome do arquivo com uma letra (p.ex., **main.py**, **app.py**, **aula.py**). Evite usar números, espaços ou acentos nos nomes dos arquivos.

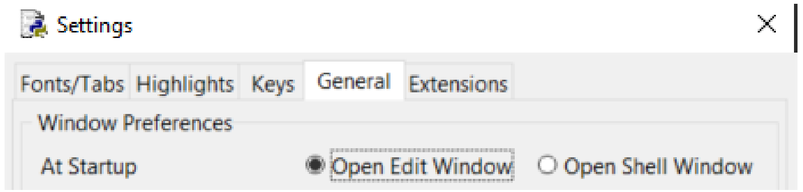
Instalando e configurando IDEs

Configurando o IDLE

- Abra IDLE utilizando o ícone do Windows e procurando pelo programa
- O programa abre no modo **Shell**, que é a janela de execução do código. Usamos essa tela apenas para checar os resultados ou quando queremos executar códigos de uma linha (testar uma função, por exemplo)
- Para abrir o editor automaticamente, vá no menu **Options > Configure IDLE**
- Na aba **General**, em **Windows Preferences**, marque a opção Open Edit Window. Clique em Ok e reinicie o IDLE



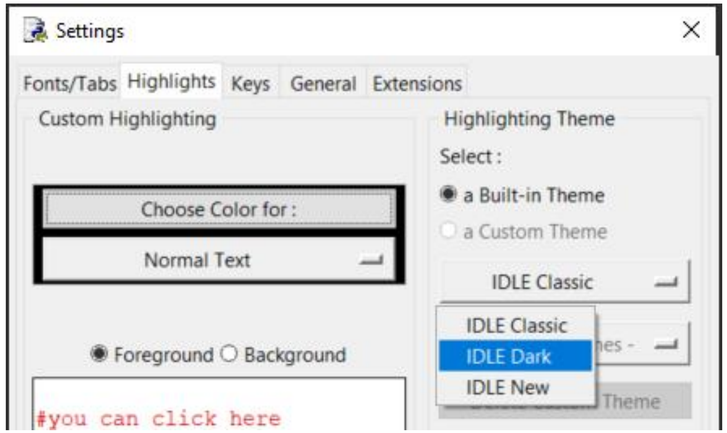




Ibmec
PROTAGONISTAS PARA O MUNDO

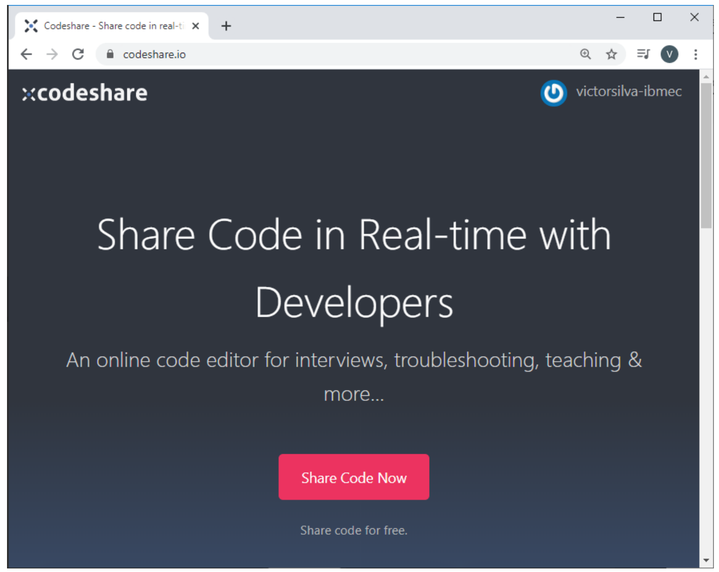
Configurando o IDLE

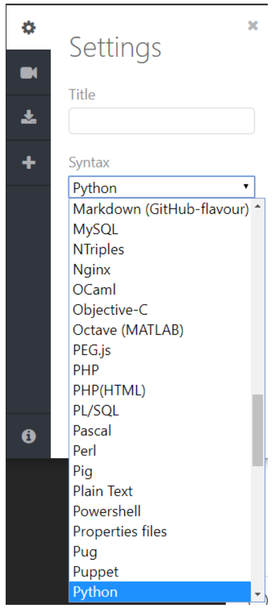
- Normalmente é usual programadores trabalharem com editores de texto que tenham um fundo escuro, o que prejudica menos a visão
- Na mesma tela de **Configurações**, vá para a aba **Highlights**, e na coluna da direita, em **Highlight Theme**, clique em **IDLE Classic** e selecione a opção **IDLE Dark**
- Clique em OK para mudar o tema para escuro

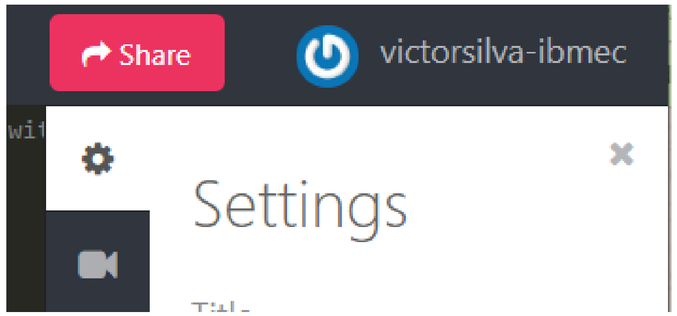


Usando o CodeShare

- Acesse http://codeshare.io e faça o cadastro
- Tendo cadastrado, na tela inicial clique em “Share Code Now”
- Um editor de texto vai aparecer. Na coluna da direita, clique na engrenagem, e em **Syntax** marque a opção **Python**
- Para compartilhar, clique em **Share**, no canto superior da janela
- O CodeShare vai liberar um link para ser usado por outras pessoas



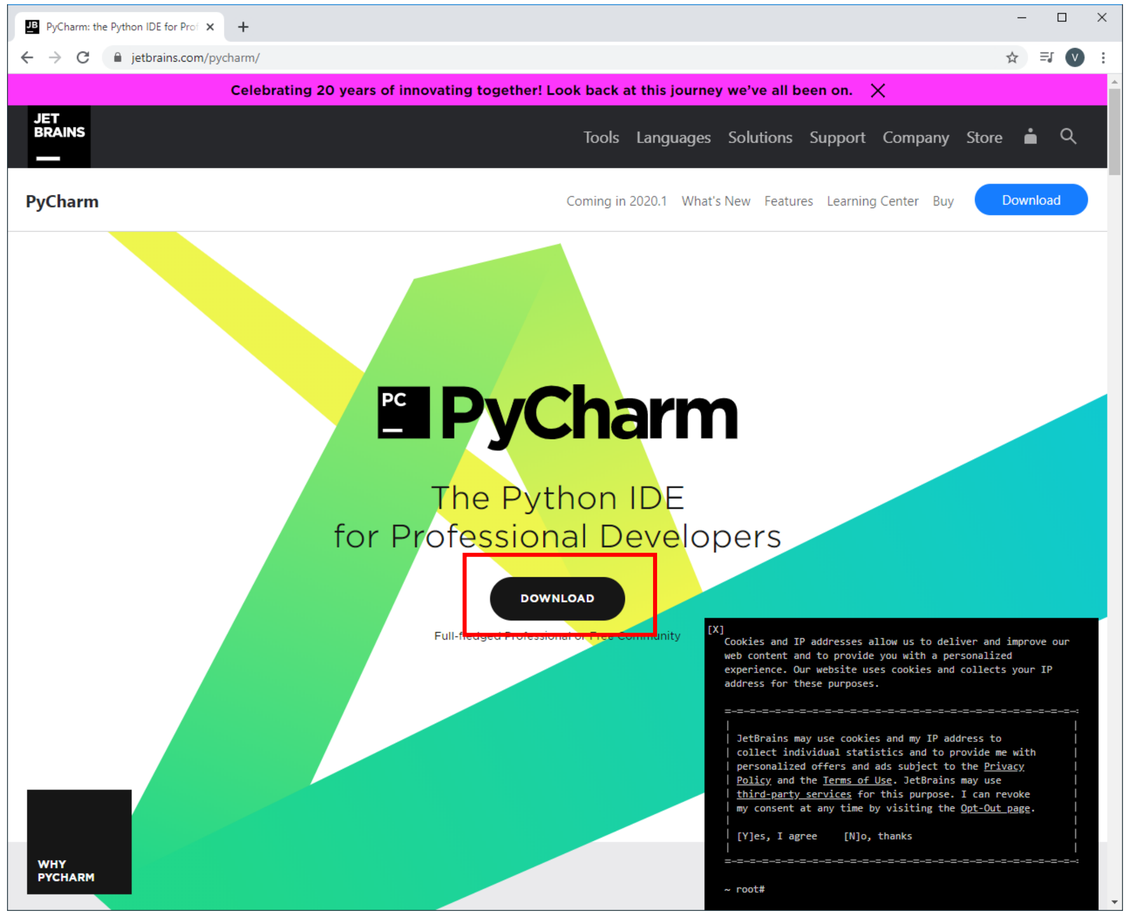




Ibmec
PROTAGONISTAS PARA O MUNDO

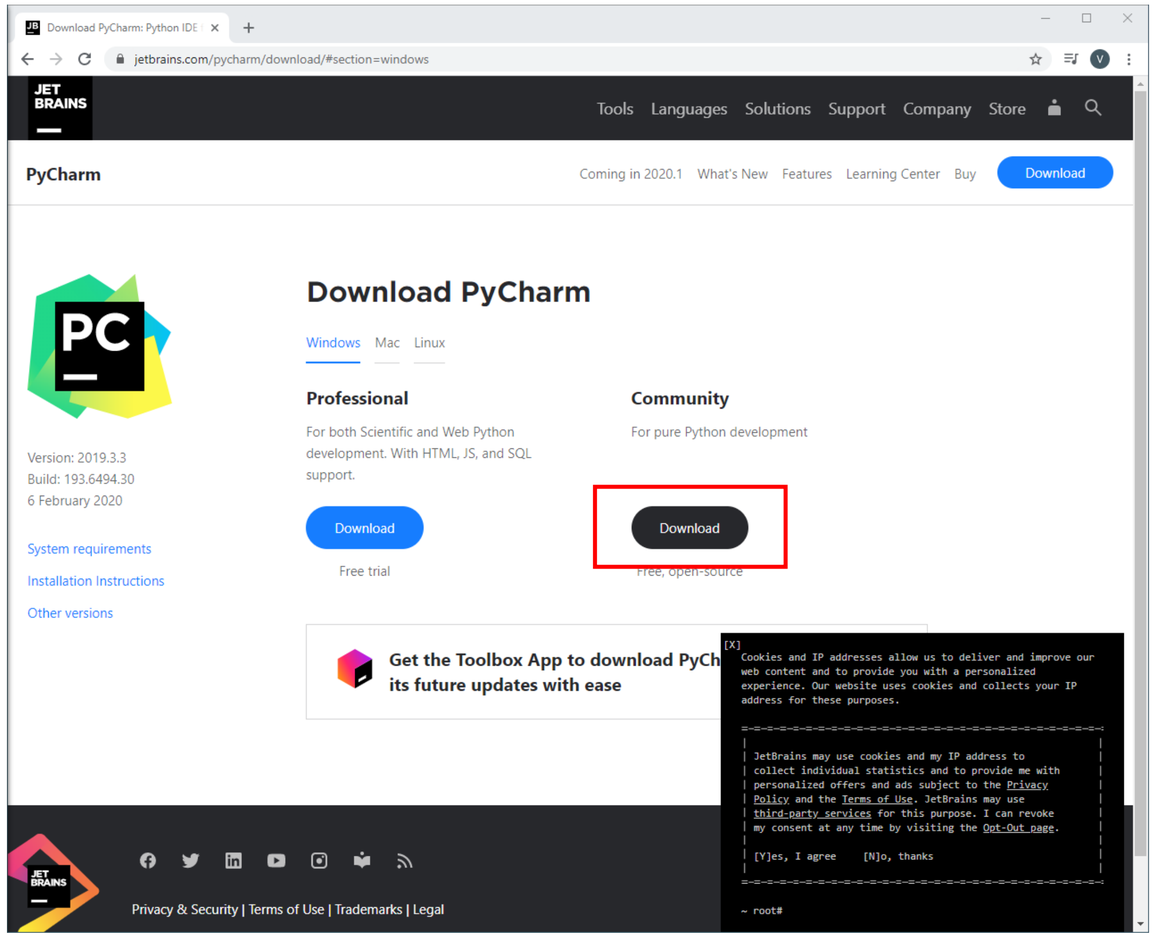
Instalando e configurando o PyCharm

- O **PyCharm** é um dos editores (ou IDEs) mais usados para a programação de aplicações em Python. Para baixar e instalar, primeiro acesse a página https://www.jetbrains.com/pycharm/ e clique em “Download”.



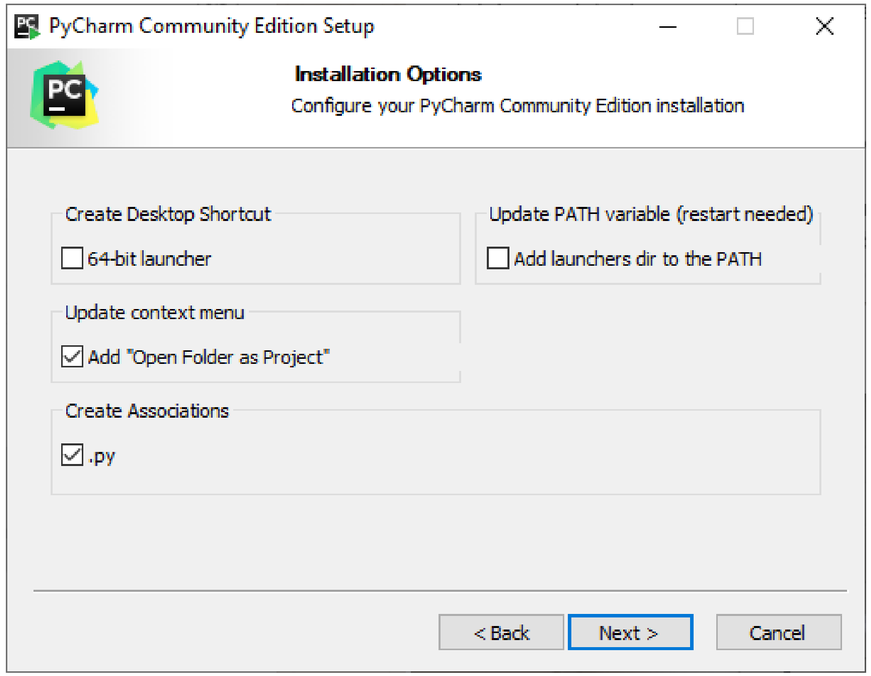
Instalando e configurando o PyCharm

- Escolha o seu sistema operacional (Windows, Mac ou Linux - vamos trabalhar com Windows) e baixe a versão “Community”. O download deve começar automaticamente.



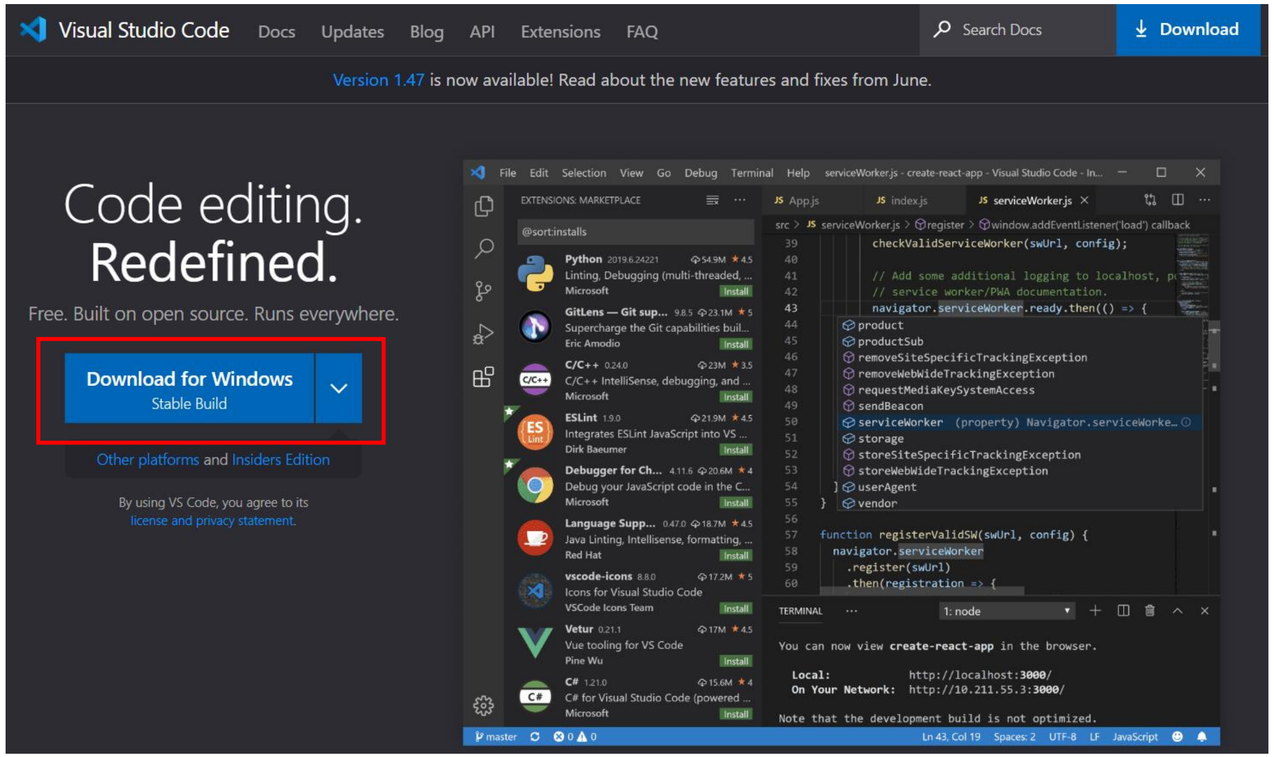
Instalando e configurando o PyCharm

- Abra o instalador do PyCharm e clique em “Next”;
- Na tela de escolha do caminho de instalação, clique em “Next”;
- Na tela seguinte, marque as opções indicadas na imagem abaixo e clique em “Next”;
- Na tela seguinte, clique em “Install” para começar a instalação.



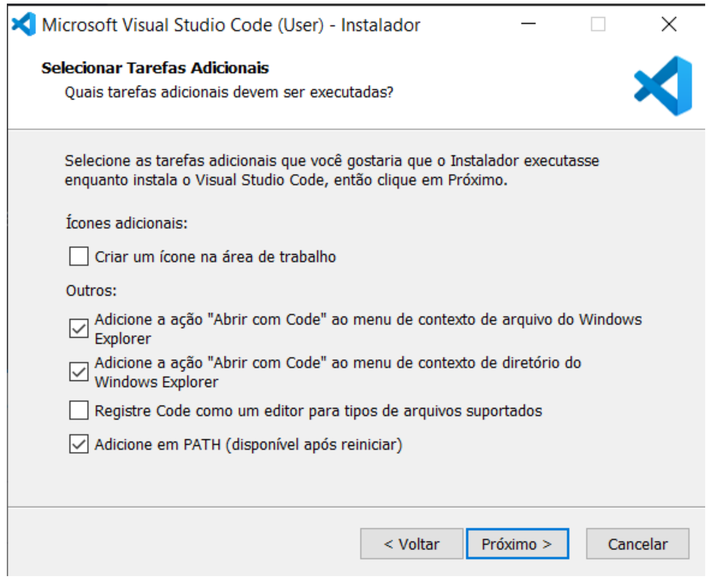
Instalando e configurando o VSCode

- O **VSCode** é um dos IDEs recentes mais famosos para basicamente qualquer linguagem de programação. Possui inúmeras extensões que facilitam e customizam o editor para a necessidade de cada programador. Para baixar e instalar, primeiro acesse a página https://code.visualstudio.com/ e clique em “Download for Windows”. Clicando na seta à direita existem opções para MAC e Linux.



Instalando e configurando o VSCode

- Abra o instalador, e após aceitar o acordo de licença e clicar em “Próximo”, clique em “Próximo” novamente até chegar na tela “Selecionar Tarefas Adicionais”
- Marque as opções abaixo e clique em “Próximo” e depois em “Instalar”

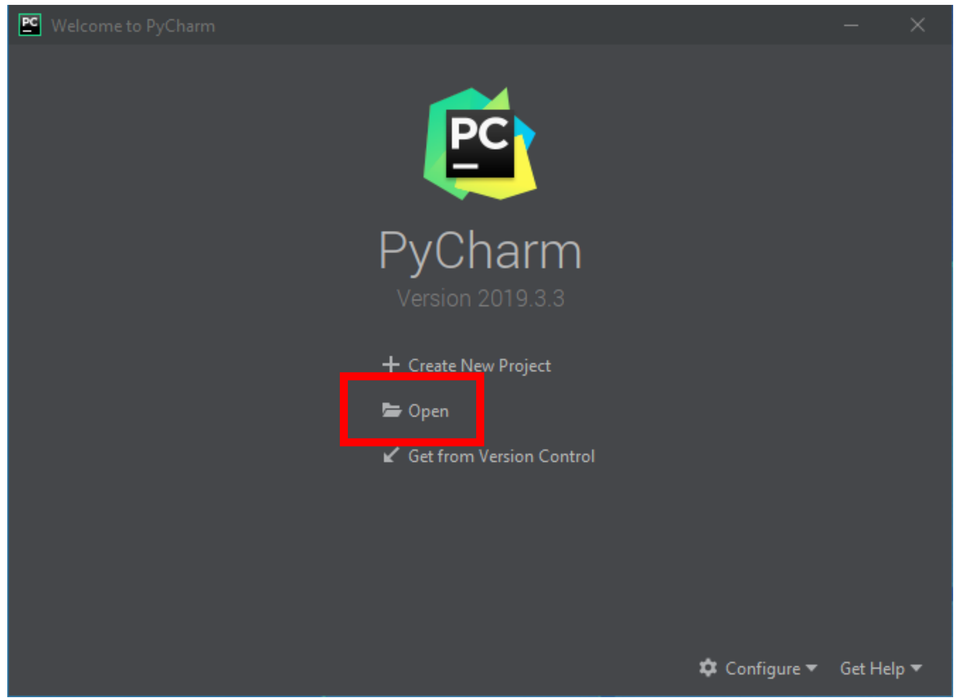


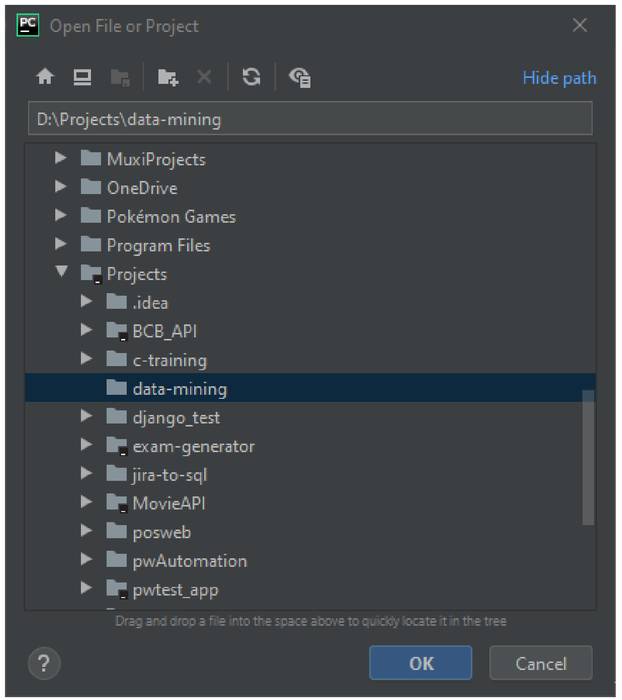
Ibmec
PROTAGONISTAS PARA O MUNDO

Utilizando o PyCharm

Iniciando o PyCharm

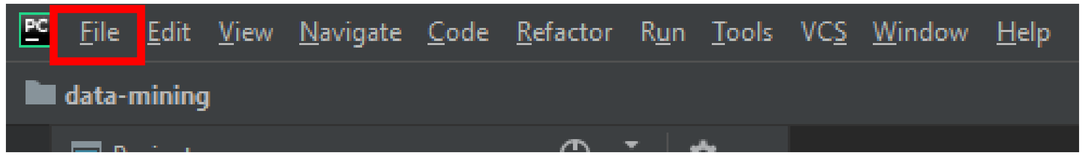
- **Se essa é a primeira vez que abre o PyCharm:**
  - Garanta que você possui uma pasta com o projeto desejado (p.ex., **D:\Projetos\data-mining**);
  - Abra o PyCharm, e na tela inicial clique em **Open**. Selecione a pasta que você criou e clique em **Ok**;

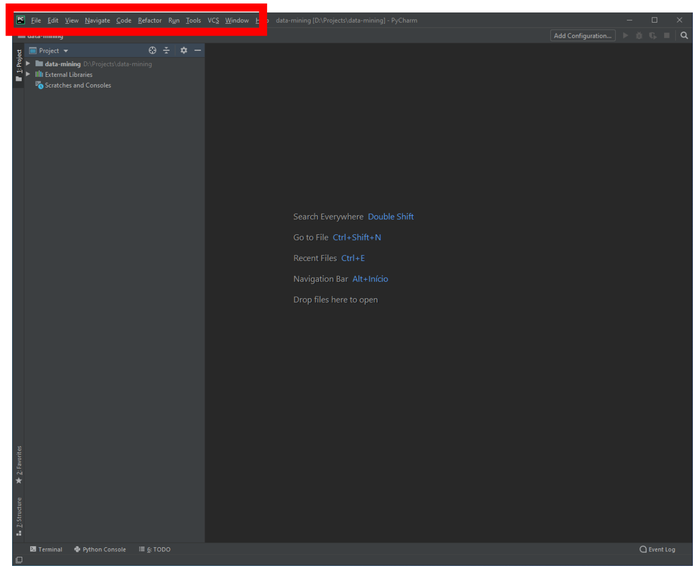


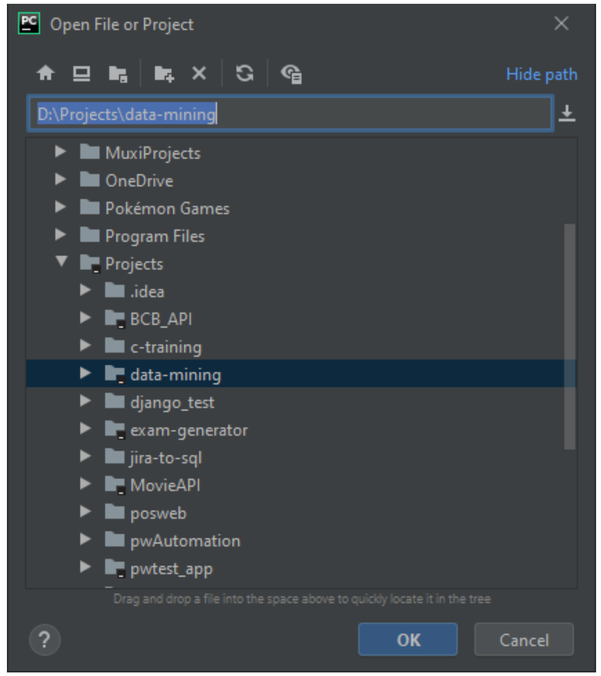


Iniciando o PyCharm

- **Se você já abriu o PyCharm antes:**
  - Garanta que você possui uma pasta com o projeto desejado (p.ex., **C:\Projetos\data-mining**);
  - Abra o PyCharm, e na tela que abrir clique em **File > Open**. Selecione a pasta que você criou e clique em **Ok**;





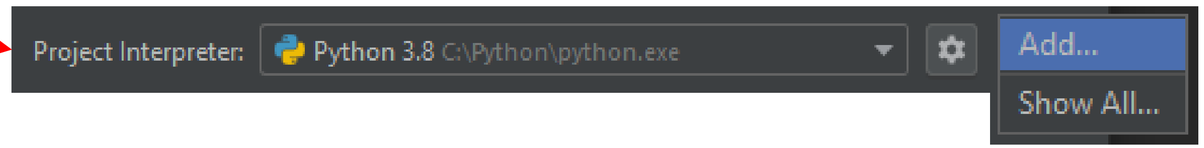


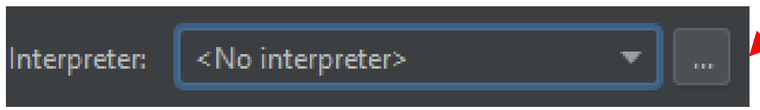
Algumas configurações do PyCharm

- **Mudar o tema para escuro:**
  - Clique em **File > Settings**;
  - Na janela que aparecer, procure por **Appearance & Behavior > Appearance**. No campo **Theme**, selecione o tema desejado (minha sugestão é o **Darcula**);
  - Clique em **Ok** para fechar a tela de configurações.

Algumas configurações do PyCharm

- **Configurar o interpretador de Python:**
  - Clique em **File > Settings**;
  - Na janela que aparecer, procure por **Project: <nome-do-projeto> > Project Interpreter**. No campo **Project Interpreter**, verifique se o Python informado é o instalado (no meu caso, **C:\Python\python.exe**). Caso não seja, clique na engrenagem e em **Add...**;
  - Na nova janela, clique em **System Interpreter**, e no campo **Interpreter** clique nos três pontos à direita para indicar o local que o seu Python está instalado. Aperte **Ok** até voltar à tela de **Settings**;
  - Novamente no campo **Project Interpreter**, altere o Python para refletir o que está instalado. Em seguida clique em **Ok**.



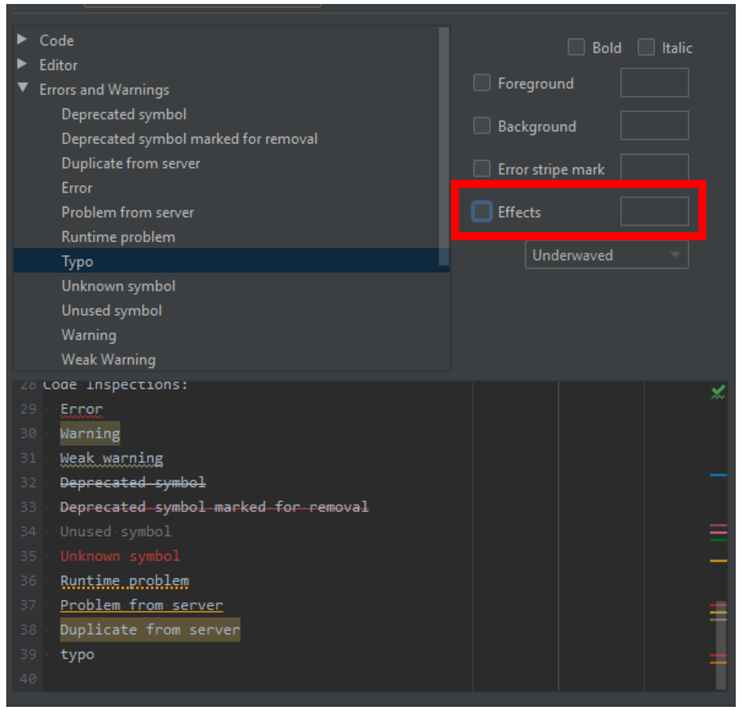


Algumas configurações do PyCharm

- **Desativar inspeção ortográfica:**
  - Clique em **File > Settings**;
  - Na janela que aparecer, procure por **Editor > Spelling**. No campo **Bundled dictionaries**, desmarque todas as opções;
  - Clique em **Ok** para sair das configurações.

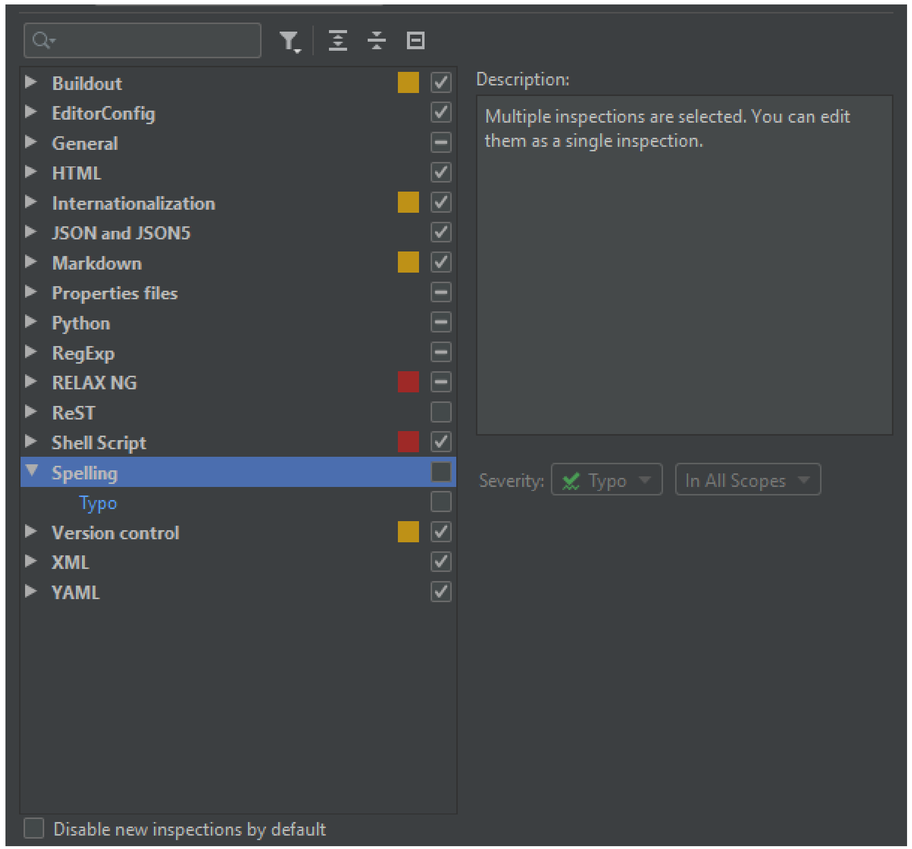
Algumas configurações do PyCharm

- **Desmarcar esquema de cores para *typos*:**
  - Clique em **File > Settings**;
  - Na janela que aparecer, procure por **Editor > Color Scheme > General**. No campo **Errors and Warnings > Typo**, desmarque a opção **Effects**;
  - Clique em **Ok** para sair das configurações.



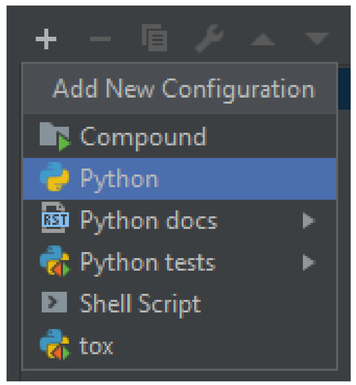
Algumas configurações do PyCharm

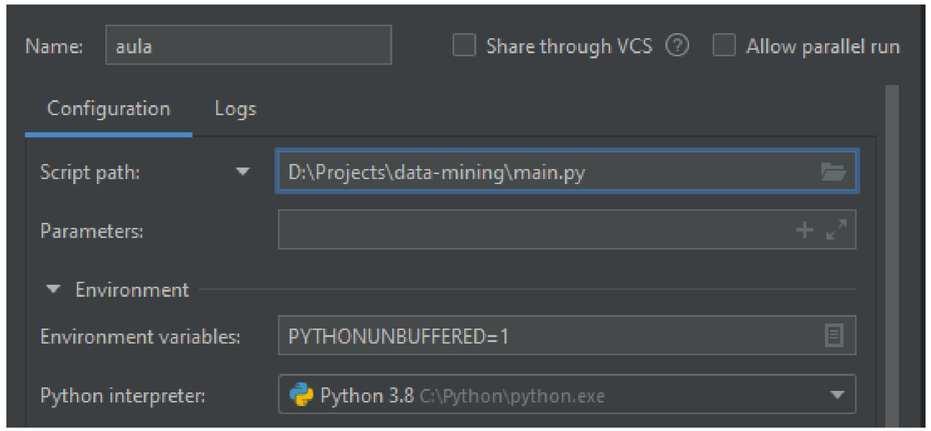
- **Ajustar os destaques de inspeções:**
  - Clique em **File > Settings**;
  - Na janela que aparecer, procure por **Editor > Inspections**. Desmarque o campo **Spelling**;
  - Na seção **Python**, recomendo *desmarcar* algumas opções para simplificar o aprendizado:
    - *Boolean variable check can be simplified;*
    - *Chained comparisons can be simplified;*
    - *Comparison with None performed with equality operators.*
  - Clique em **Ok** para sair das configurações.



Algumas configurações do PyCharm

- **Configurar uma determinada execução de código:**
  - Garanta que as configurações do slide **Configurar o interpretador de Python** foram executadas;
  - Aperte **Alt + Shift + F10** > **Edit Configurations**, ou na barra de tarefas clique em **Run > Edit Configurations**;
  - Na janela que aparecer, no canto superior esquerdo clique no botão de **+ > Python**;
  - Dê um nome para a execução (p.ex., **aula**) e em **Script path**, informe o caminho do arquivo que você deseja executar;
  - Certifique-se que o **Python interpreter** é o configurado anteriormente, clique em **Apply** e em seguida em **Close**;
  - Para rodar o arquivo, é só usar o atalho **Shift + F10**.

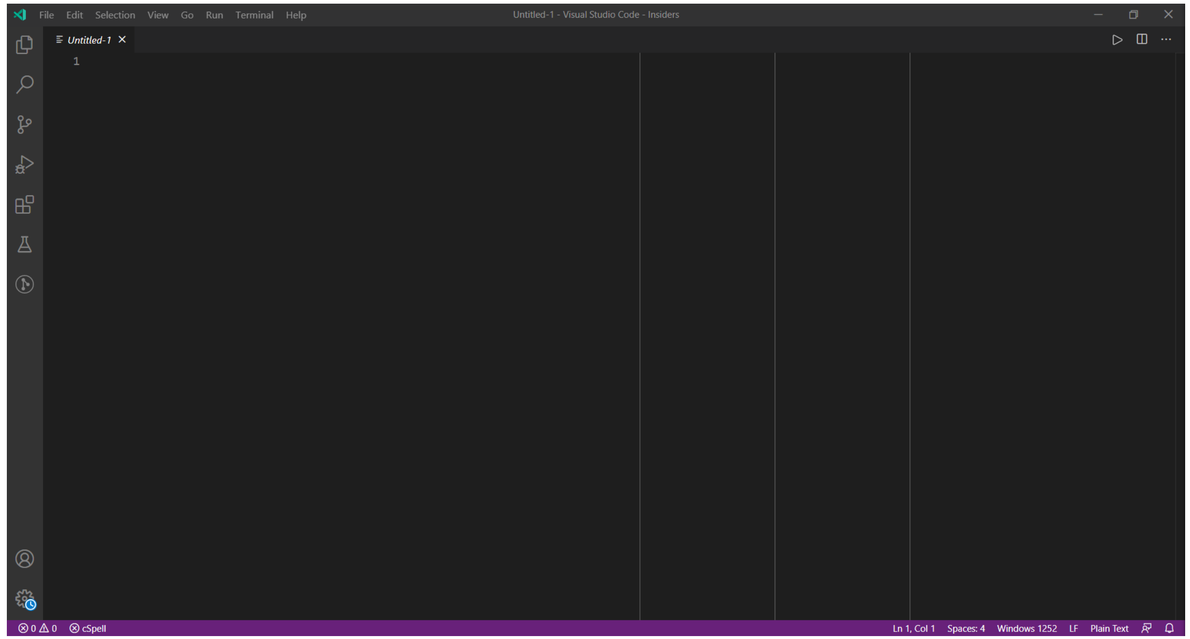




Utilizando o VSCode

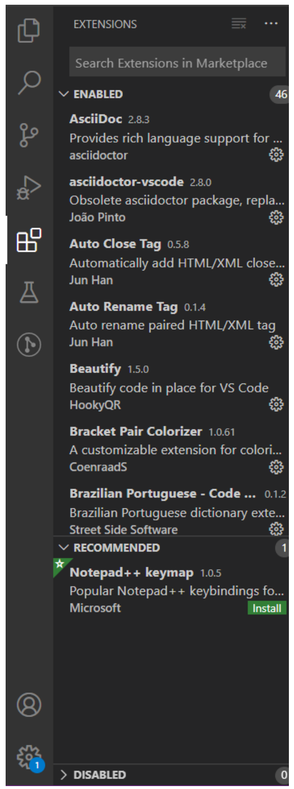
Iniciando o VSCode

- **Se essa é a primeira vez que abre o VSCode:**
  - Garanta que você possui uma pasta com o projeto desejado (p.ex., **D:\Projetos\algoritmos**);
  - Abra o VSCode, e na tela inicial clique em **File > Open Folder**. Selecione a pasta que você criou e clique em **Selecionar Pasta**.



Iniciando o VSCode

- **Instalando extensões:**
  - Boa parte do atrativo no VSCode é a possibilidade de instalar extensões e customizar a experiência de uso;
  - Para instalar extensões, vá na coluna à esquerda e clique no ícone  ;
  - Uma barra de extensões vai aparecer, e você poderá buscar as extensões desejadas e instalá-las;
  - Abaixo seguem algumas sugestões de extensões para o curso:
    - Beautify
    - Bracket Pair Colorizer
    - Brazilian Portuguese - Code Spell Checker
    - C/C++
    - Code Runner
    - GitLens - Git supercharged
    - Git History Diff
    - Git History
    - Gitconfig Syntax
    - Markdownlint
    - Partial Diff
    - Python
    - Visual Studio IntelliCode
    - vscode-python-docstring
    - AsciiDoc



Iniciando o VSCode

- **Atalhos interessantes no VSCode:**
  - Ctrl + K, Ctrl + O: Abre uma pasta
  - Ctrl + D: Quando uma palavra estiver selecionada, seleciona todas as palavras no arquivo
  - Ctrl + F: Procura por uma palavra ou sentença no arquivo
  - Ctrl + Shift + F: Procura por uma palavra ou sentença em todos os arquivos da pasta
  - Ctrl + H: Substitui uma palavra ou sentença por outra em todo o arquivo
  - Alt + ↓ ou Alt + ↑: Move a linha inteira para baixo ou para cima
  - Shift + Alt + ↓ ou Shift + Alt + ↑: Copia a linha inteira para baixo ou para cima
  - Ctrl + Alt + ↓ ou Ctrl + Alt + ↑: Incluí um ou mais cursores nas linhas abaixo ou acima

Iniciando o VSCode

- **Atalhos interessantes no VSCode:**
  - F12: Vai para a declaração de uma variável ou função
  - F12 F12: Mostra todos os usos de uma variável ou função
  - Alt + → ou Alt + ←: Retrocede ou avança na última posição do cursor
  - Ctrl + S: Salva um arquivo
  - Ctrl + P: Abre um arquivo diretamente
  - Ctrl + ,: Abre o menu de configurações
  - Ctrl + Shift + P: Abre uma dropdown com opções de configuração
  - Ctrl + ‘: Abre a janela do terminal

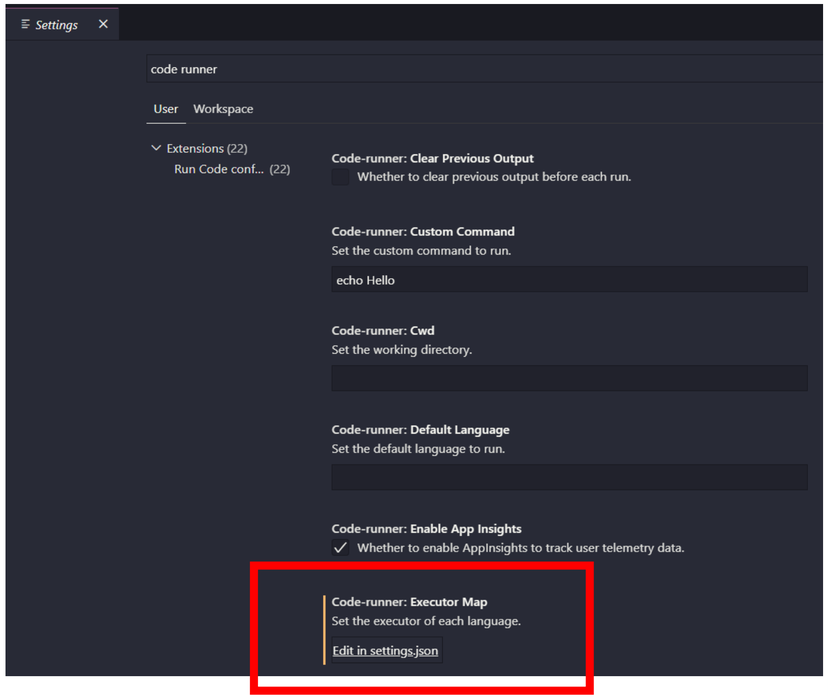
Iniciando o VSCode

- **Atalhos interessantes no VSCode:**
  - Ctrl + G: Vai para uma linha específica do arquivo
  - Ctrl +]: Divide a tela em duas
  - Ctrl + 1 (ou 2, 3, 4): Seleciona um painel específico
  - Alt + Shift + 0: Alterna entre divisão na horizontal ou na vertical
  - Ctrl + F4 ou Ctrl + W: Fecha o arquivo selecionado (se for um painel selecionado e ele estiver vazio, fecha o painel)
  - Ctrl + ;: Comenta a(s) linha(s) selecionada(s)

Iniciando o VSCode

- **Executando um código Python no VSCode com a extensão *Code Runner*:**
  - Essa extensão permite a execução do código através de um atalho
  - Com a extensão instalada, use o atalho **Ctrl + ,** ou vá em **File > Preferences > Settings**;
  - Na barra de pesquisa, digite “Code Runner”. Procure pela caixa abaixo e clique em “Edit in settings.json”;
  - Inclua as seguintes linhas (edite o endereço para o caminho em que o Python está instalado):

```
"python.pythonPath": "C:\\Python\\python.exe",
"code-runner.executorMap": {
    "python": "\"C:\\Python\\python.exe\""
},
"code-runner.runInTerminal": true,
```



  - Para rodar, na janela do código use o atalho **Ctrl + Alt + N**.

Configurando o pylint

Configurando o pylint

- O pylint é um pacote do Python que auxilia na adequação do código aos padrões de programação da comunidade Python
- Para instalar o pylint...
  - ...para MacOS veja aqui
  - ...para Windows veja aqui
- Para executar o pylint basta entrar com o comando **pylint <caminho>**, com o caminho do arquivo que deseja rodar o linter.

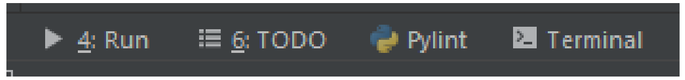
```
C:\Users\vmachado>pylint D:\Victor\Pessoal\IBMEC\2020.2\ALG\Aulas\teste.py
************* Module teste
D:\Victor\Pessoal\IBMEC\2020.2\ALG\Aulas\teste.py:1:0: C0114: Missing module docstring (missing-module-docstring)

-----------------------------------
Your code has been rated at 0.00/10
```

Ibmec
PROTAGONISTAS PARA O MUNDO

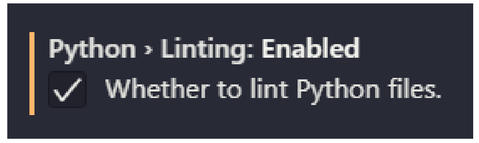
Configurando o pylint

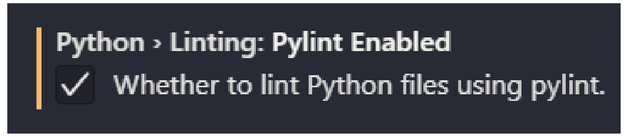
- O pylint ainda pode ser rodado direto no terminal do VSCode ou do PyCharm, basta abrir o terminar e seguir os mesmos passos mencionados anteriormente. No entanto, os ambos os IDEs fornecem meios de utilizar o pylint diretamente durante a implementação do código.
- **Usando o pylint no PyCharm:**
  - O PyCharm já possui a inspeção automática do editor, após configurar o interpretador. Para identificar as inspeções vá em **File > Settings** e na janela clique em **Editor** e depois em **Inspections**;
  - O PyCharm também possui um plugin. Em **File > Settings**, clique em **Plugins** e busque por **Pylint**. Instale e reinicie o IDE. Sempre que abrir um arquivo Python o IDE vai incluir a opção “Pylint” no canto inferior esquerdo. Para executar basta clicar na opção e depois em “Run”.



Configurando o pylint

- **Usando o pylint no VSCode:**
  - Com a extensão “Python” instalada no VSCode e o pacote pylint instalado, abra o menu de configurações (use o atalho **Ctrl + ,** ou vá em **File > Preferences > Settings**) e procure por python linting;
  - Marque as seguintes opções:
    - Python > Linting: Enabled
    - Python > Linting: Lint On Save
    - Python > Linting: Pylint Enabled
  - Para rodar o pylint sem ser pelo terminal, use o atalho **Ctrl + Shift + P** e em seguida digite “Run Linting” e aperte Enter. O IDE vai marcar no código os problemas.





Ibmec
PROTAGONISTAS PARA O MUNDO

Instalando pacotes pelo PyPI

Instalando pacotes pelo PyPI

- Uma das grandes vantagens ao se programar em Python é ter à disposição uma gama de pacotes e bibliotecas disponíveis pela própria comunidade, que desenvolve novas funcionalidades e distribui online, na maioria das vezes de forma gratuita.
- Um dos locais mais confiáveis e mais simples de se obter um novo pacote é através do PyPI, um repositório oficial de pacotes da linguagem, mantido pela própria organização que mantém o Python.
- O PyPI é acessado utilizando uma ferramenta chamada **pip**, que é instalada automaticamente ao se instalar o interpretador de Python. Portanto, a instalação de novos pacotes é muito fácil de se realizar, com apenas uma linha de comando.

Instalando pacotes pelo PyPI

- Instalando pacotes via pip no Windows:
  - Clique no botão do Windows  ;
  - Digite a caixa de pesquisa **prompt de comando**, e abra o programa;
  - No terminal entre com o seguinte comando:

```
C:\Users\vmachado>C:\Python\python.exe -m pip install pylint
```

  - Lembre-se de substituir o caminho do Python para o caminho instalado no seu computador, e altere **pylint** para o pacote escolhido.

Ibmec
PROTAGONISTAS PARA O MUNDO

Instalando pacotes pelo PyPI

- A instalação do Python no MacOS não costuma vir com o pip. Caso não tenha vindo, tente fazer os passos abaixo primeiro:
  - Abra o terminal (pasta **Applications > Utilities > Terminal**);
  - Entre com os seguintes comandos:

```
curl https://bootstrap.pypa.io/get-pip.py > get-pip.py
sudo python get-pip.py
```

- Para instalar um pacote via pip no MacOS:
  - Usando o mesmo terminal usado para instalar o pip, entre com o seguinte comando:

```
sudo pip install <nome_do_pacote>
```

- Veja o vídeo abaixo para mais detalhes:
  - https://www.youtube.com/watch?v=yBdZZGPpYxg

Ibmec
PROTAGONISTAS PARA O MUNDO

Algoritmos

Introdução

- Um algoritmo é um processo sistemático para a resolução de um problema. O desenvolvimento de algoritmos é particularmente importante para problemas a serem solucionados em um computador, pela própria natureza do instrumento utilizado.
- Um algoritmo computa uma saída, o resultado do problema, a partir de uma entrada, as informações inicialmente conhecidas e que permitem encontrar a solução do problema. Durante o processo de computação o algoritmo manipula dados, gerados a partir da sua entrada.
- O estudo de estruturas de dados não pode ser desvinculado de seus aspectos algorítmicos. A escolha correta da estrutura adequada a cada caso depende diretamente do conhecimento de algoritmos para manipular a estrutura de maneira eficiente.

Apresentação dos algoritmos

- As convenções seguintes serão utilizadas com respeito à linguagem:
  - O início e o final de cada bloco são determinados por endentação, isto é, pela posição da margem esquerda. Se uma certa linha do algoritmo inicia um bloco, ele se estende até a última linha seguinte, cuja margem esquerda se localiza mais à direita do que a primeira do bloco;
  - A declaração de atribuição é indicada pelo símbolo :=;
  - As declarações seguintes são empregadas com significado semelhante ao usual:

```
se... então
se... então... senão
enquanto... faça
para... faça
pare
```

  - Variáveis simples, vetores, matrizes e registro são considerados como tradicionalmente em linguagens de programação. Os elementos de vetores e matrizes são identificados por índices entre colchetes.

```
para i := 1, ..., |__n/2__|
    temp := S[i]
    S[i] := S[n - i + 1]
    S[n - i + 1] := temp
```

Ibmec
PROTAGONISTAS PARA O MUNDO

Recursividade

- Um tipo especial de procedimento será utilizado, algumas vezes, ao longo do curso. É aquele que contém, em sua descrição, uma ou mais chamadas a si mesmo. Um procedimento dessa natureza é denominado **recursivo**.
- Naturalmente, todo procedimento, recursivo ou não, deve possuir pelo menos uma chamada proveniente de um local exterior a ele. Essa chamada é denominada **externa**.
- Um procedimento não recursivo é, pois, aquele em que todas as chamadas são externas.

Recursividade

- O exemplo clássico mais simples de recursividade é o cálculo do fatorial de um inteiro $n \geq 0$:

```
função fat(i)
    fat(i) := se i <= 1 então 1 senão i * fat(i - 1)
```

```
fat[0] := 1
para j := 1, ..., n faça
    fat[j] := j * fat[j - 1]
```

- Um exemplo conhecido, onde a solução recursiva é natural e intuitiva, é o do **Problema da Torre de Hanói**.

Ibmec
PROTAGONISTAS PARA O MUNDO

Recursividade

- A solução do problema é descrita a seguir. Naturalmente, para $n > 1$, o pino-trabalho deve ser utilizado como área de armazenamento temporário. O raciocínio utilizado para resolver o problema é semelhante ao de uma prova matemática por indução. Suponha que se saiba como resolver o problema até $n - 1$ discos, $n > 1$, de forma recursiva. A extensão para n discos pode ser obtida pela realização dos seguintes passos:
  - Resolver o problema da Torre de Hanói para os $n - 1$ discos do topo do pino-origem A, supondo que o pino-destino seja C e o trabalho seja B;
  - Mover o n-ésimo pino (maior de todos) de A para B;
  - Resolver o problema da Torre de Hanói para os $n - 1$ discos localizados no pino C, suposto origem, considerando os pinos A e B como trabalho e destino, respectivamente.

```
procedimento hanoi(n, A, B, C)
se n > 0 então
    hanoi(n - 1, A, C, B)
    mover o disco do topo de A para B
    hanoi(n - 1, C, B, A)
```

Complexidade de algoritmos

- Conforme já mencionado, uma característica muito importante de qualquer algoritmo é o seu tempo de execução. Naturalmente, é possível determiná-lo através de métodos empíricos, isto é, obter o tempo de execução através da execução propriamente dita do algoritmo, considerando-se entradas diversas.
- Ao contrário do método empírico, o método analítico visa aferir o tempo de execução de forma independente do computador utilizado, da linguagem e dos compiladores empregados e das condições locais de processamento.

Complexidade de algoritmos

- As seguintes simplificações serão introduzidas para o modelo proposto:
  - Suponha que a quantidade de dados a serem manipulados pelo algoritmo seja suficientemente grande. Somente o comportamento assintótico será avaliado.
  - Não serão consideradas constantes aditivas ou multiplicativas na expressão matemática obtida. Isto é, a expressão matemática obtida será válida, a menos de tais constantes.
- O processo de execução de um algoritmo pode ser dividido em etapas elementares, denominadas *passos*. Cada passo consiste na execução de um número fixo de operações básicas cujos tempos de execução são considerados constantes.

```
para i := 1, ..., |__n/2__|
    temp := S[i]
    S[i] := S[n - i + 1]
    S[n - i + 1] := temp
```

Ibmec
PROTAGONISTAS PARA O MUNDO

Complexidade de algoritmos

- Como exemplos adicionais, considere os problemas de determinar as matrizes soma C e produto D de duas matrizes dadas.

```
para i := 1, ..., n faça
    para j := 1, ..., n faça
        C[i][j] := A[i][j] + B[i][j]
```

```
para i:= 1, ..., n faça
    para j := 1, ..., n faça
        C[i][j] := 0
        para k := 1, ..., n faça
            C[i][j] := C[i][j] + A[i][k] * B[k][j]
```

Ibmec
PROTAGONISTAS PARA O MUNDO

Complexidade de algoritmos

- Noção de complexidade:
  - Seja A um algoritmo, $\{E_1, \dots, E_m\}$, o conjunto de todas as entradas possíveis de A. Denote por t_i o número de passos efetuados por A, quando a entrada for E_i. Definem-se, com p_i sendo a probabilidade de ocorrência da entrada E_i:
    - Complexidade do pior caso: $\max_{E_{i \in E}}\{t_i\}$;
    - Complexidade do melhor caso: $min_{E_{i \in E}}\{t_i\}$;
    - Complexidade do caso médio: $\sum_{1 \leq i \leq m}(p_i \times t_i)$.
  - As complexidades têm por objetivo avaliar a eficiência de tempo ou espaço. A complexidade de tempo de pior caso corresponde ao número de passos que o algoritmo efetua no seu pior caso de execução, isto é, para a entrada mais desfavorável. De certa forma, a complexidade de pior caso é a mais importante das três mencionadas.

Ibmec
PROTAGONISTAS PARA O MUNDO

A Notação O

- Quando se considera o número de passos efetuados por um algoritmo, podem-se desprezar constantes aditivas ou multiplicativas.
- Por exemplo, um valor de número de passos igual a 3n será aproximado para n.
- Além disso, como o interesse é restrito a valores assintóticos, termos de menor grau também podem ser desprezados. Assim, um valor de número de passos igual a $n^2 + n$ será aproximado para n^2. O valor $6n^3 + 4n - 9$ será transformado em n^3.
- Torna-se útil, portanto, descrever operadores matemáticos que sejam capazes de representar situações como essas. A notação O será utilizada com essa finalidade.

Ibmec
PROTAGONISTAS PARA O MUNDO

A Notação O

- Sejam f, h funções reais positivas de variável inteira n. Diz-se que f é $O(h)$, escrevendo-se $f = O(h)$, quando existir uma constante $c > 0$ e um valor inteiro n_o, tal que:

$$n > n_o \Rightarrow f(n) \leq c \times h(n)$$

- Ou seja, a função h atua como um limite superior para valores assintóticos da função f. Em seguida são apresentados alguns exemplos da notação O.

$$f = n^2 - 1 \Rightarrow f = O(n^2)$$

$$f = n^3 - 1 \Rightarrow f = O(n^3)$$

$$f = 403 \Rightarrow f = O(1)$$

$$f = 5 + 2\log n + 3\log^2 n \Rightarrow f = O(\log^2 n)$$

Ibmec
PROTAGONISTAS PARA O MUNDO

OBRIGADO!

www.ibmec.br

/ibmec

ibmec

@ibmec_oficial

ibmec

Ibmec

PROTAGONISTAS PARA O MUNDO