# Engenharia de Software para Inteligência Artificial – Notas de Aula

Sumário

[Engenharia de Software para Inteligência Artificial – Notas de Aula 1](#_Toc58163801)

[Unidade 01 - Engenharia de Software 2](#_Toc58163802)

[Ambiente de Engenharia de Software 2](#_Toc58163803)

[Modelagem de Software 2](#_Toc58163804)

[Ciclo de Vida de Software 3](#_Toc58163805)

[Técnicas de Inteligência Artificial Aplicada à Engenharia de Software 4](#_Toc58163806)

[Exemplos de Inteligência Artificial Aplicada à Engenharia de Software 5](#_Toc58163807)

[Unidade 02 - Começando a programar 5](#_Toc58163808)

[Python para Inteligência Artificial 5](#_Toc58163809)

[Conhecendo os Tipos de Dados 6](#_Toc58163810)

[Estruturas Condicionais e de Repetição 6](#_Toc58163811)

[Funções 6](#_Toc58163812)

[Unidade 03 - Manipulação de dados 6](#_Toc58163813)

[Numpy 6](#_Toc58163814)

[Pandas 10](#_Toc58163815)

[Unidade 04 - Visualização de dados 11](#_Toc58163816)

[A biblioteca Matplotlib 12](#_Toc58163817)

[A biblioteca Gráfica BOKEH 13](#_Toc58163818)

# Unidade 01 - Engenharia de Software

Essa unidade tem como objetivo:

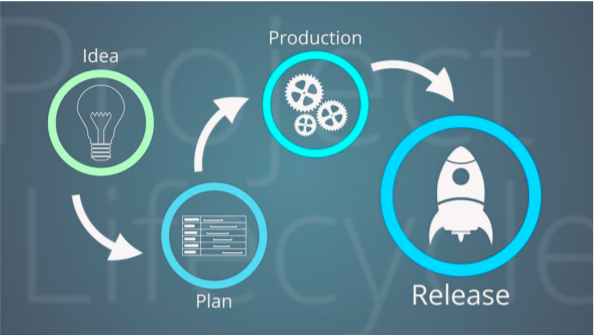
* Apresentar a parte introdutória;
* Explicar os princípios de Engenharia de software;
* Demonstrar Gerenciamento de Projetos;
* Apresentar a Engenharia de software aplicada a IA;
* Expor soluções de mercado que usam a IA aplicada.

## Ambiente de Engenharia de Software

* O desenvolvimento de software para alcançar o sucesso desejado precisa ser desenhado, planejado, gerenciado, executado dentro de prazos por equipe treinada, testado, e ter **manutenção contínua** dentro do seu **ciclo de vida**.

## Modelagem de Software

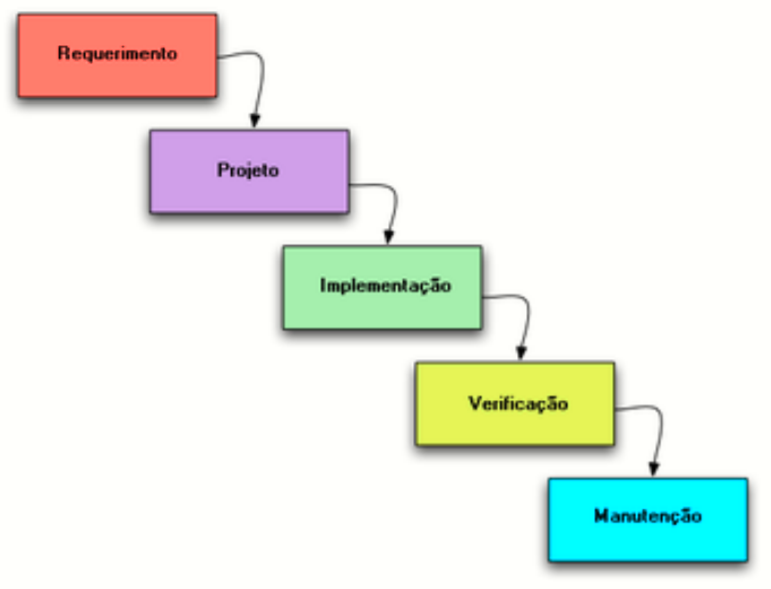
* O projeto de desenvolvimento deve se basear num tripé de processos bem definidos e maduros
  + **Métodos:** estudo da metodologia que melhor se aplica ao objetivo que se busca para o software que se deseja desenvolver, isto é, os objetivos que devem ser alcançados;
  + **Ferramentas:** quais ferramentas de mercado melhor auxiliam ao desenvolvimento, isto é, que irão entregar mais recursos para maximizar e otimizar a etapa de desenvolvimento;
  + **Procedimentos:** quais tarefas deverão ser executadas, e como, para o sucesso do projeto.
* **Requisitos de software**
  + **Funcionais:** O que o software fará, as funções da aplicação;
  + **Não-Funcionais:** Como o software irá desempenhar as suas funções, aspectos relacionados a qualidade do software.



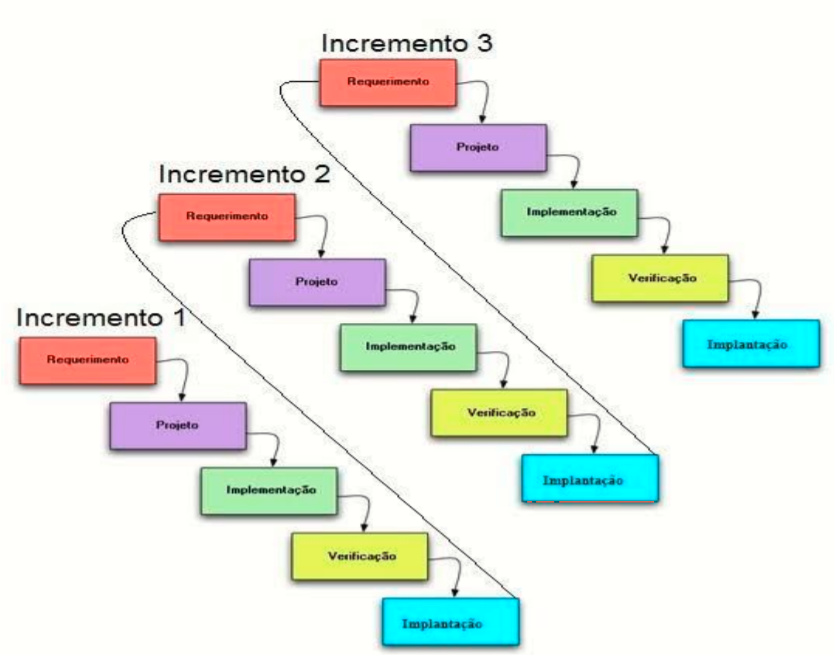
* **Implementação de requisitos:** O que fazer?, Como fazer?, e O que mudar?
  + **Definição:** deverão ser tratados aspectos como: análise de requisitos, análise do sistema de software, e planejamento do projeto de software;
  + **Desenvolvimento:** serão tratados os assuntos relacionados com o projeto em si, a codificação, e a realização dos testes;
  + **Manutenção:** onde serão levantadas e executadas as correções e adaptações necessárias, assim como as melhorias funcionais do software em questão.

## Ciclo de Vida de Software

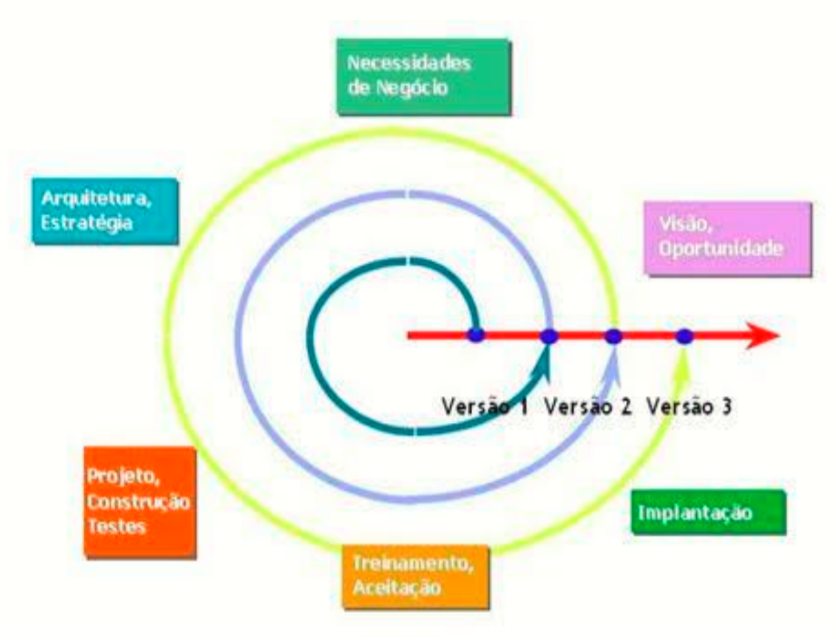
* Fases do ciclo de vida do software
  + Projeto
  + Implementação
  + Avaliação e melhora contínua do software
* Os modelos mais difundidos de clico de vida de software são
  + **Cascata:** definido como uma sequência unidirecional de tarefas que transforma requisitos em funcionalidades de um sistema. É modelado tendo o fim de uma fase e o início da próxima, resultando na aparência de uma “cascata” com fases dependentes e interligadas gerando resultados que, ao final do processo, serão juntados para gerar o resultado final.
    - Modelo engessado, desatualizado.



* + **Iterativo:** tem como resultado uma sequência de versões executáveis do software, gerindo assim uma integração contínua da arquitetura do sistema para a produção de novas versões.
    - Modelo repete etapas
    - Faz tantas repetições quantas forem necessárias para amadurecer o software



* + **Incremental e Espiral (Mais atual):** também chamado de modelo de Boehm, é um modelo cíclico e experimental, que minimiza o grau de risco do projeto e maximiza os de definição e implementação.
    - Etapas curtas com foco em entregar um produto acabado e funcional ao cliente o mais rapidamente possível.
    - Cada ciclo agrega novas funcionalidades.



## Técnicas de Inteligência Artificial Aplicada à Engenharia de Software

* **Machine Learning:** É uma tecnologia pertencente ao campo da inteligência artificial dedicada a medir a capacidade que o software tem de aprender sozinho, bem como tornar isso possível. Hoje, o aprendizado das máquinas é a principal tecnologia impulsionadora da inteligência artificial.
  + Esses algoritmos usam análises estatísticas para prever respostas mais precisamente e entregam o melhor resultado preditivo com menos chance de erro.
* Tipos de aprendizado para Machine Learning
  + **Supervisionado:** são aqueles em que o ser humano precisa interagir controlando a saída e entrada de dados e interfere no treinamento da máquina fazendo comentários sobre a precisão das previsões. Por fim, a máquina aplica o que foi aprendido no seu algoritmo para a próxima análise.
  + **Não supervisionado:** os algoritmos utilizam o deep learning (aprendizagem profunda) para processar tarefas complexas sem o treinamento humano.

## Exemplos de Inteligência Artificial Aplicada à Engenharia de Software

* **Google:** realiza o preenchimento automático das buscas e prevê com alta precisão o que o usuário deseja pesquisar. O Google também já usa a IA para testar carros 100% autônomos, evitando colisões e engarrafamentos sozinho.
* **Amazon:** uma das maiores empresas de e-commerce do mundo faz recomendações personalizadas de produtos e serviços aos usuários usando algoritmos de Machine Learning.
* **Waze:** o aplicativo de navegação reúne dados do usuário, na internet, e da região que está inserido, via satélite, para indicar as melhores rotas em apenas um clique.
* **Facebook:** a rede social mais famosa do mundo conta com o reconhecimento facial das imagens postadas para recomendar marcações em fotos.
* **Siri:** o aplicativo de assistente pessoal usa o processamento de voz para se comunicar com o usuário.

# Unidade 02 - Começando a programar

Essa unidade tem como objetivo:

* Explicar as linguagens de programação;
* Apresentar os tipos de dados mais comuns;
* Demonstrar as estruturas de dados (condicionais e repetição);
* Apresentar as funções em Python.

## Python para Inteligência Artificial

* Python
  + Simplicidade e versatilidade de Sintaxe
  + Menos complexo que C++ e Java
  + Multiplataforma
  + Alto nível
  + Estilos de programação orientados a objetos, procedimentais e funcionais
  + Rica e extensa variedade de bibliotecas e ferramentas
* Ferramentas adicionais
  + Jupyter Notebook
  + Google Colab

## Conhecendo os Tipos de Dados

* String
* Números: int, float
* Coleções: Listas, Tuplas, Conjuntos, Dicionários

## Estruturas Condicionais e de Repetição

* If/Elif/Else
* For
* While

## Funções

* Criação de função com **def**

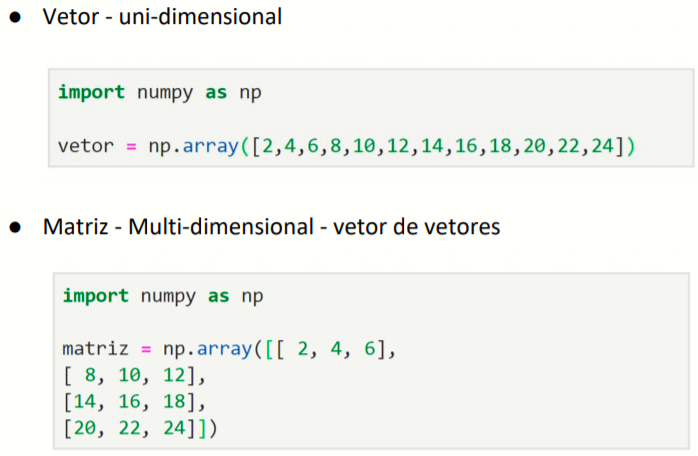
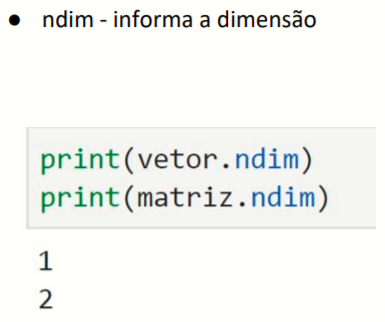
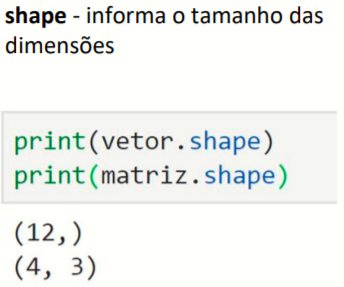
# Unidade 03 - Manipulação de dados

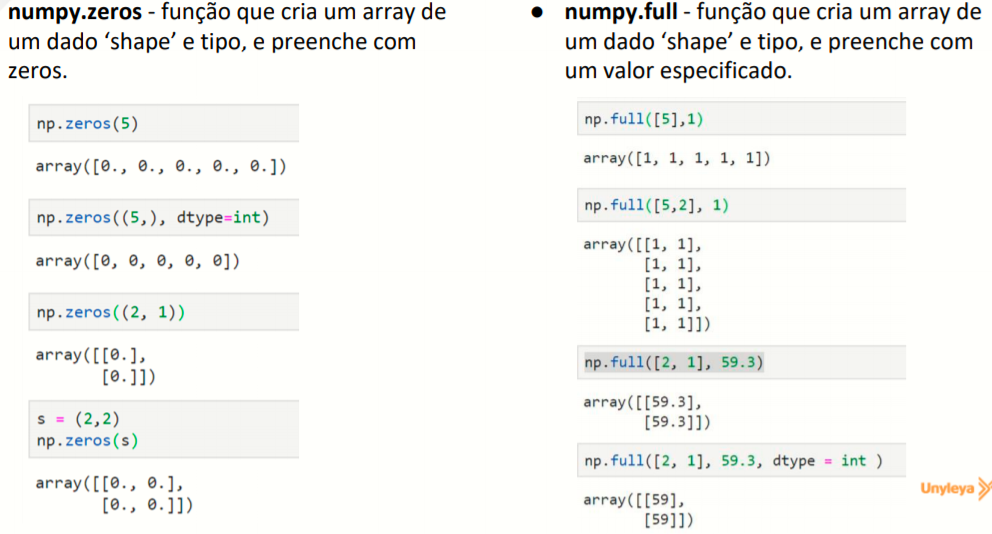
Essa unidade tem como objetivo:

* Apresentar as bibliotecas Python;
* Tratar da Manipulação de dados com Numpy (vetores e matrizes);
* Explicar a Manipulação de dataframes (leitura de dados) com a biblioteca Pandas.

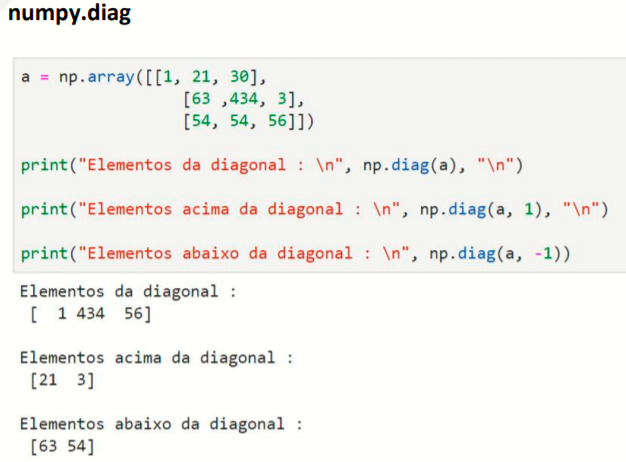
## Numpy

* Tensores

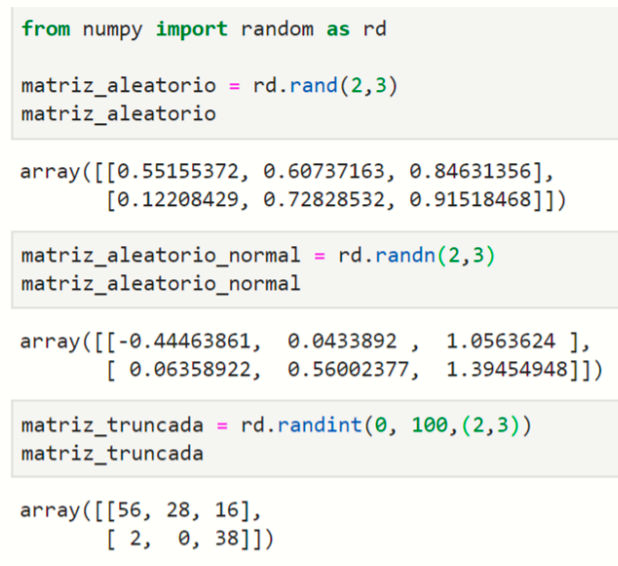
  



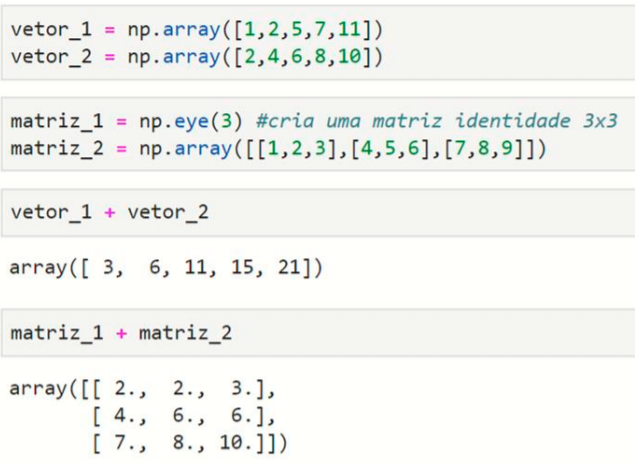
* **Diag:** Exibe elementos da diagonal principal, ou diagonais superiores e inferiores.



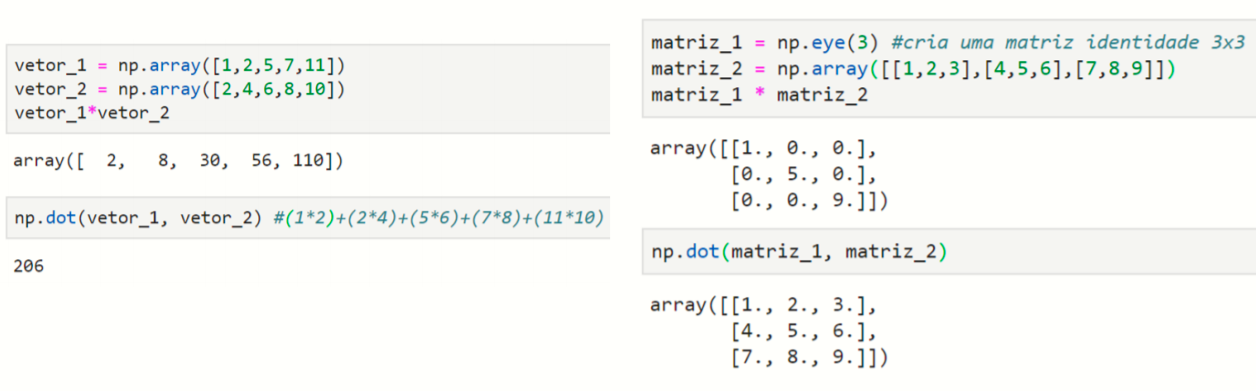
* **Rand:** cria uma matriz com entradas criadas aleatoriamente a partir de uma distribuição uniforme.
* **Randn**: cria uma matriz com entradas criadas aleatoriamente a partir de uma distribuição normal.
* **Randint:** cria uma matriz com inteiros em uma faixa de valor determinada.



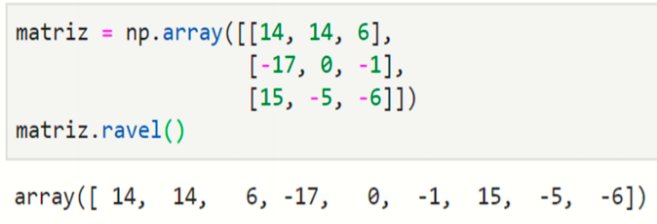
* Operações Matemáticas – Soma de Vetores e Matrizes



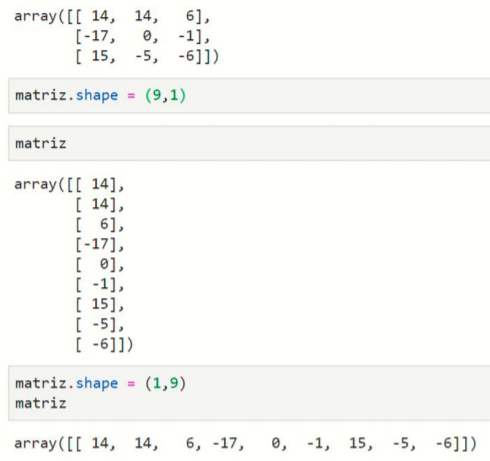
* Multiplicação



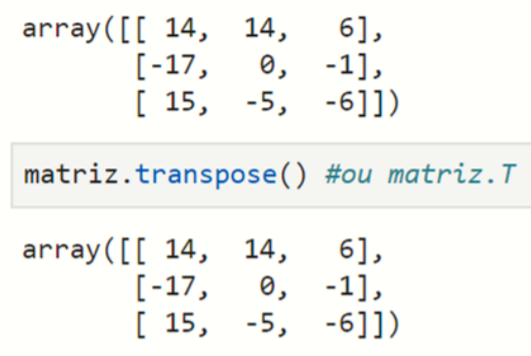
* Manipulação de Matrizes – Função **Ravel:** Faz com que os elementos sejam distribuídos em uma única linha, de forma ordenada da esquerda para a direita, linha por linha.



* **Shape:** Muda a quantidade de linhas e colunas em uma matriz. A única coisa que deve ser respeitada é a quantidade de elementos na matriz. O novo shape deve respeitar a propriedade size.

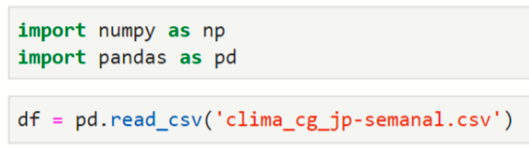


* **Transpose:** cria a matriz transposta, trocando linhas por colunas.



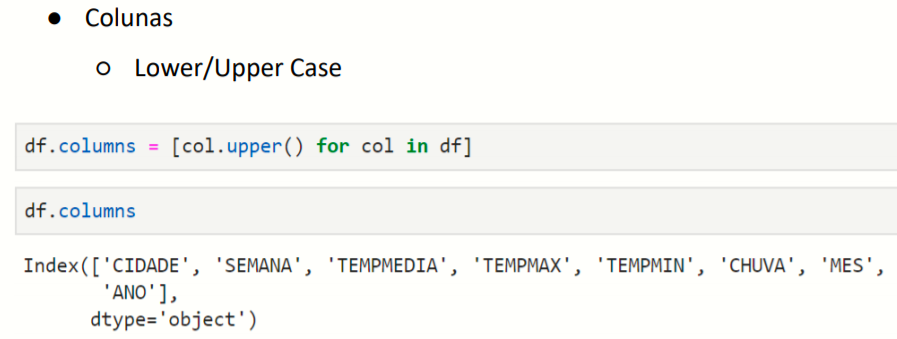
## Pandas

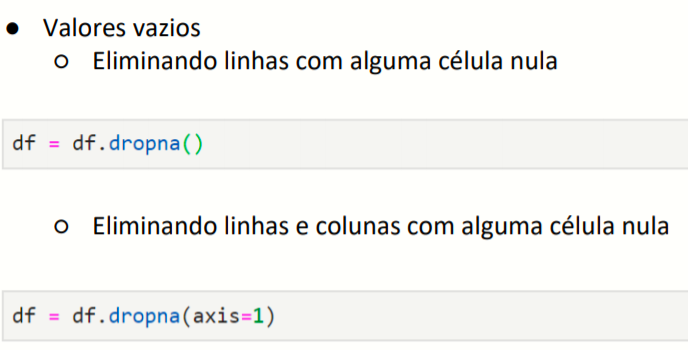
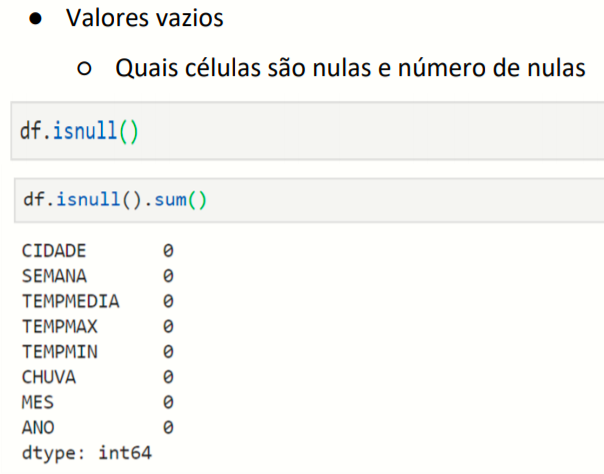
* Biblioteca para Python
  + Fácil gestão de dados
  + Carregar dados em arquivos de diferentes formatos
  + Aplicar transformações
    - Junções com outros conjuntos de dados
    - Remover dados incorretos ou desnecessários
  + Processar bad data
    - Dados que não estão organizados em forma útil para o trabalho em questão
* Importação de dados
  + CSV: read\_csv
  + JSON: read\_json
  + HTML: read\_html
  + Microsoft Excel: read\_excel
* **DataFrame:** equivalente a uma tabela.

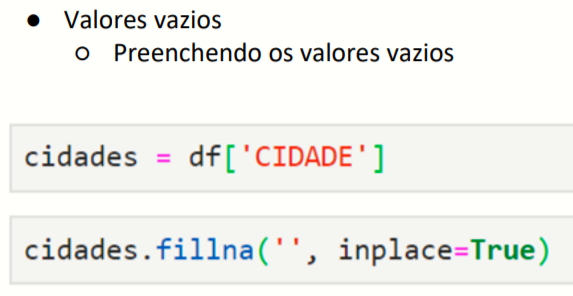


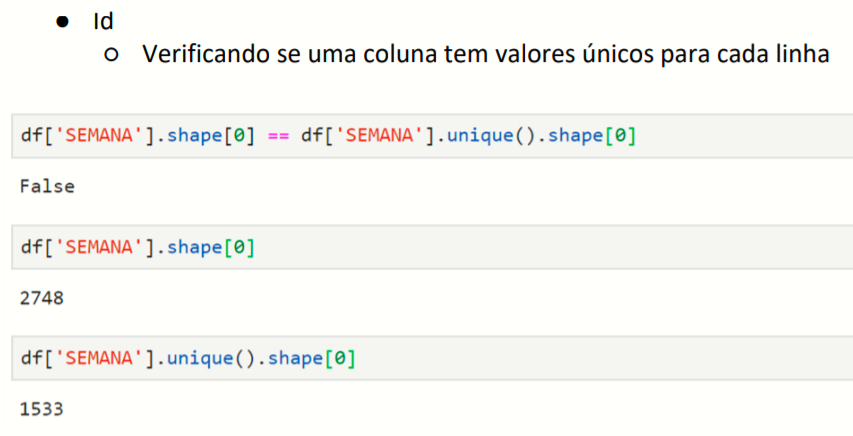
* Alguns comandos
  + df.head()
  + df.tail()
  + df.info()
* Eliminando Duplicadas: 











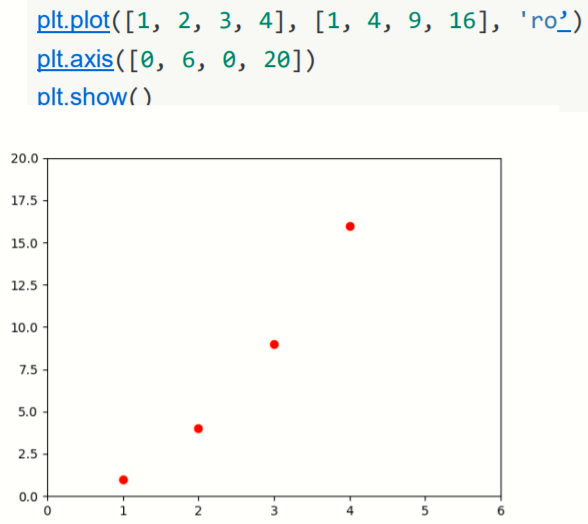
# Unidade 04 - Visualização de dados

Essa unidade tem como objetivo:

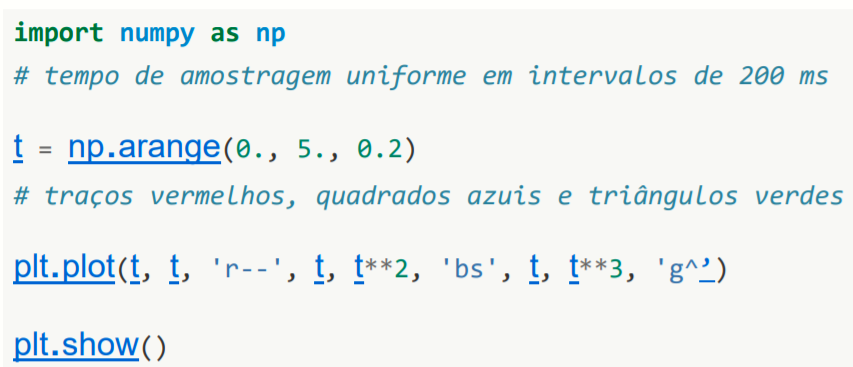
* Apresentar os recursos de visualização de dados da biblioteca MATPLOTLIB;
* Demonstrar os recursos de visualização de dados da biblioteca BOKEH.

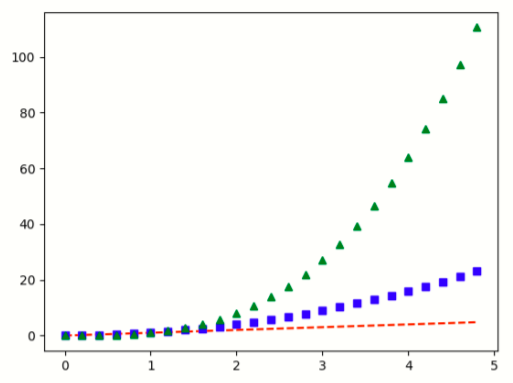
## A biblioteca Matplotlib

* É uma das bibliotecas mais usadas para gráficos na comunidade Python, sendo conhecida por seu formato simples e prático.
* import matplotlib.pyplot as plt
* Exemplo



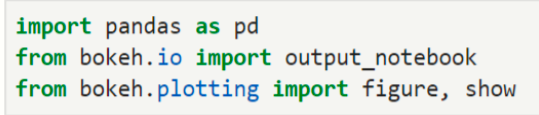
* Trabalhando com matrizes



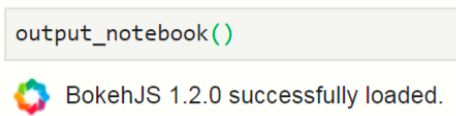


## A biblioteca Gráfica BOKEH

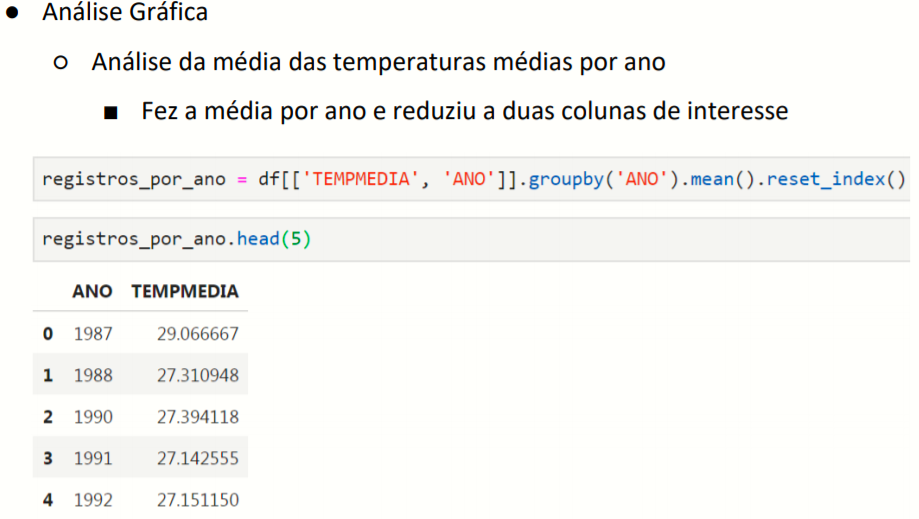
* Biblioteca de visualização com capacidade de criar gráficos interativos prontos para a web, que podem ser facilmente produzidos como objetos, Documentos HTML ou aplicativos da Web interativos.
* **Instalação** 
  + Uma opção é usar a distribuição Anaconda Python e inserir este comando em um prompt de comando do Bash ou do Windows: **conda install bokeh**
    - Isso instala todas as dependências de que o Bokeh precisa. O Anaconda minimiza o esforço de instalação em todas as plataformas e configurações, incluindo Windows, e também instala os exemplos no examples/subdiretório de seu diretório de instalação do Anaconda ou Miniconda.
  + Outra opção, se você tiver instaladas todas as dependências necessárias, como NumPy, você pode usar pip na linha de comando: **pip install bokeh**
* **Importando a biblioteca**

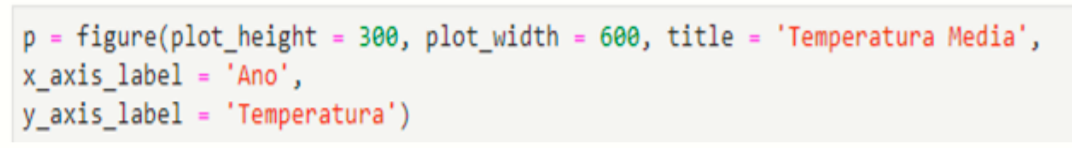


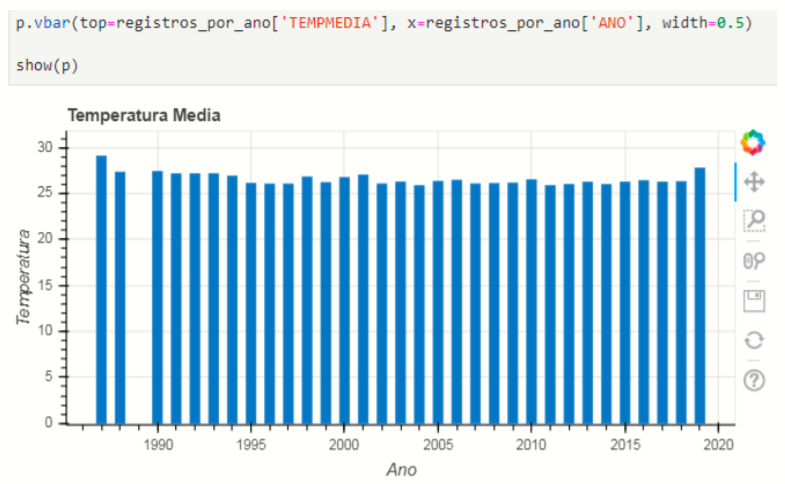
* **Visualização no computador**



* **Exemplo**







* Alterando o tipo do gráfico



* Formatando o gráfico

