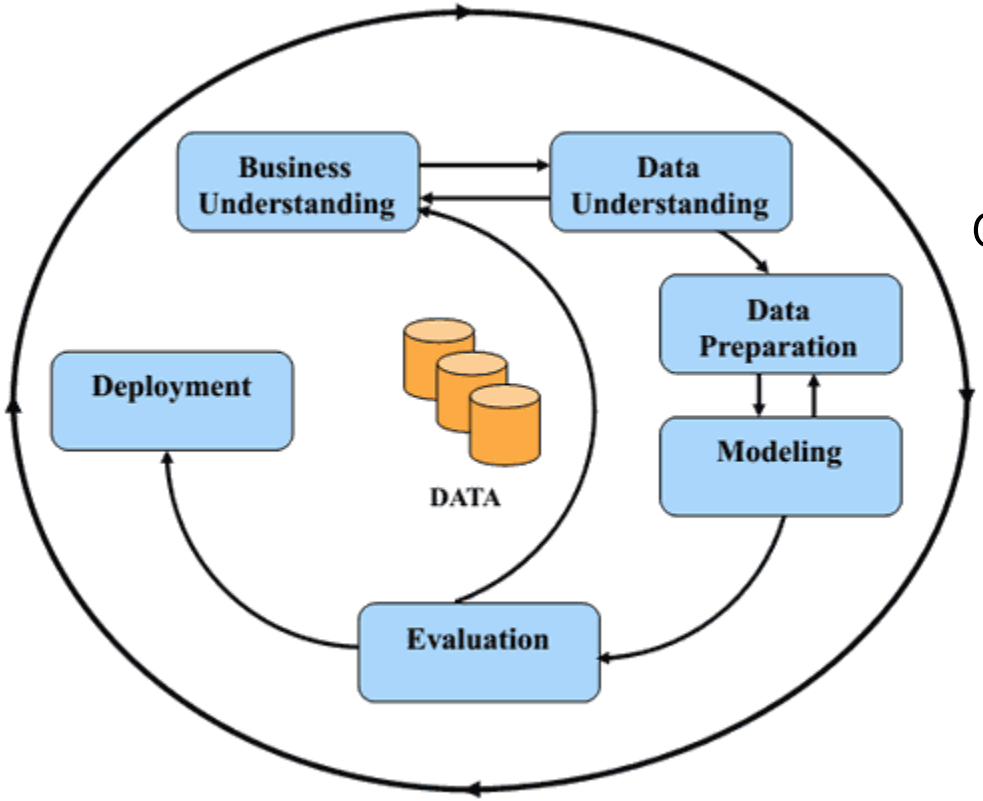
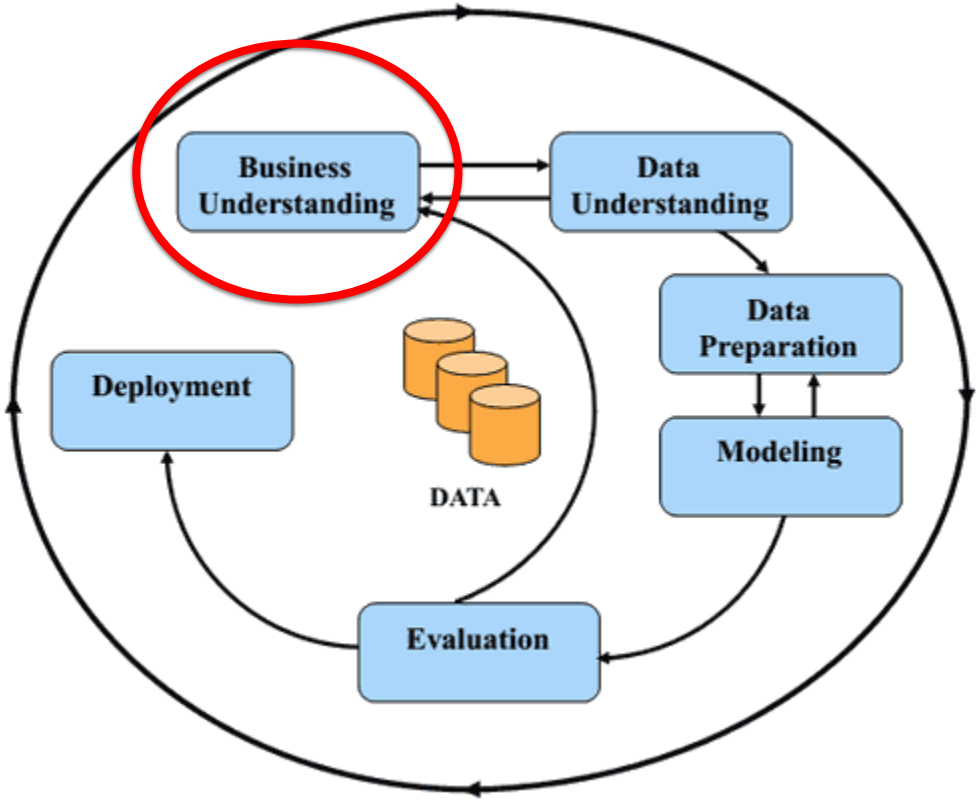
Aula 05



CRISP-DM é uma abordagem estruturada de mineração de dados proposta por um grupo de estudos patrocinado pela união europeia – Cross Industry Standard Process for Data Mining. O modelo abstrai seis passos comuns em projetos de mineração, como na imagem abaixo.

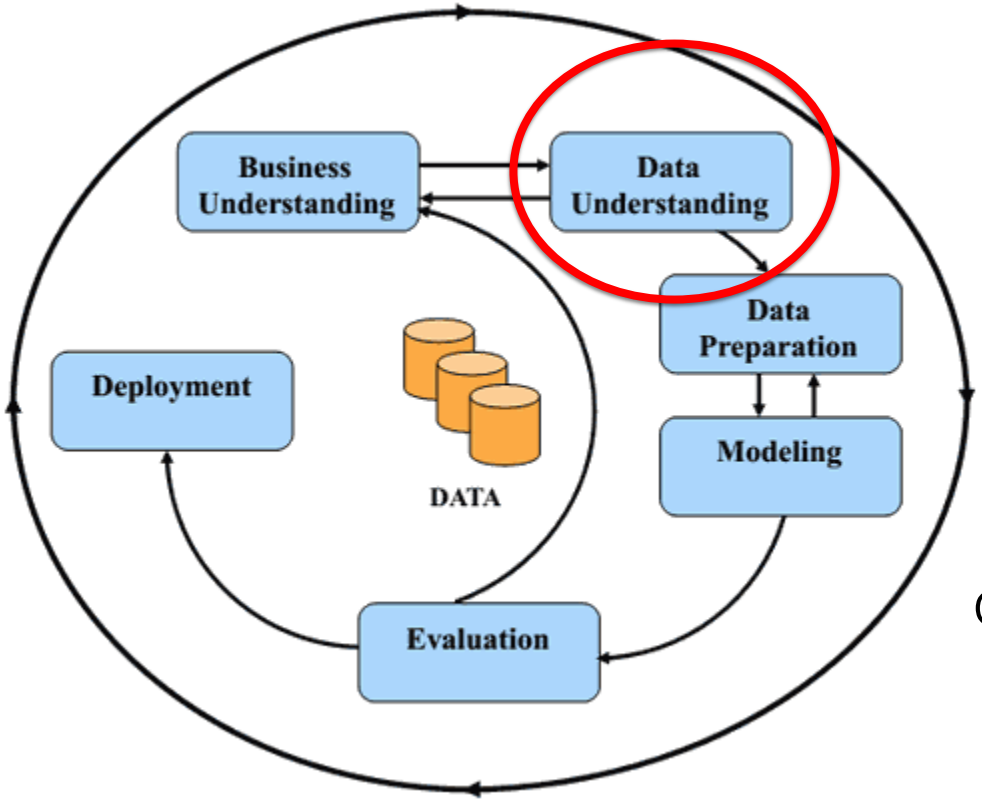


Business Understanding

Principais Objetivos

Definir os critérios de sucesso
Formas de produção?

Como integrar o output com as tecnologias existentes?



Data Understanding

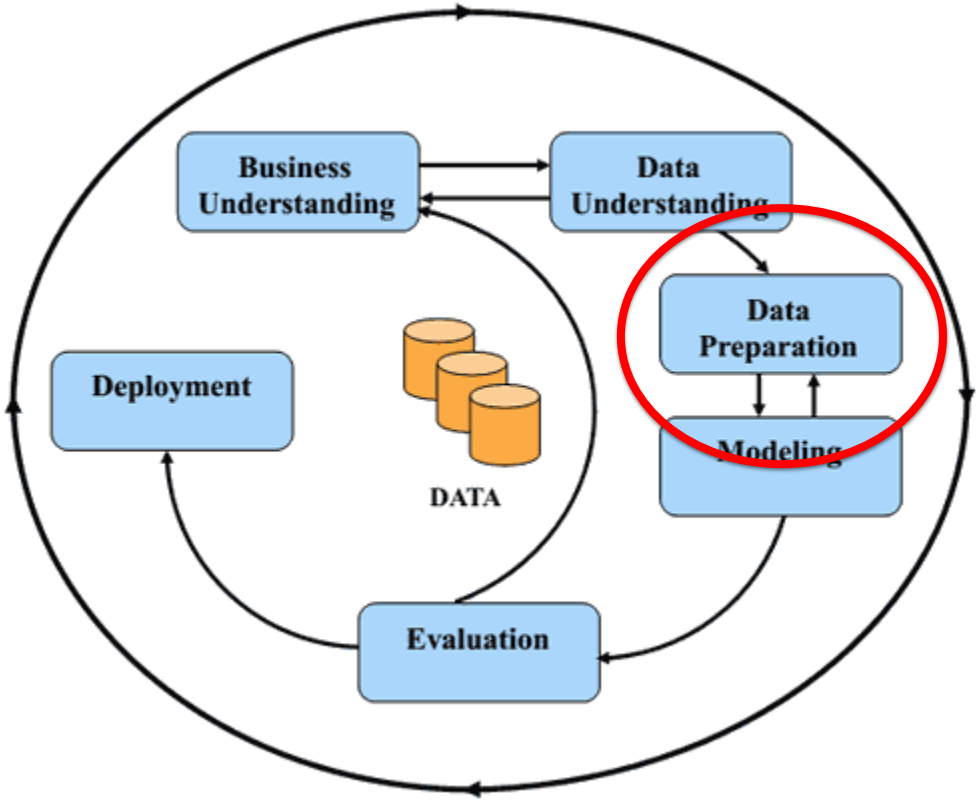
Principais Objetivos

Recolha de dados

Quais são as fontes de dados?

Análise exploratória de dados
construção de gráficos de dados simples (histogramas, etc)
para ajudar a compreender a distribuição de dados

Data Preparation



Principais Objetivos

Preparação de dados

Limpeza de Dados - Tratamento de Ruído, Deteção de Anomalias,

Transformação de dados

Normalização de dados, discretização de dados

Redução de dados

•**Completude**: Refere-se à ausência de valores ausentes. Dados completos garantem que todas as informações necessárias estão disponíveis.

•**Consistência**: Os dados devem ser uniformes em toda a base, sem contradições ou duplicações que possam comprometer a análise.

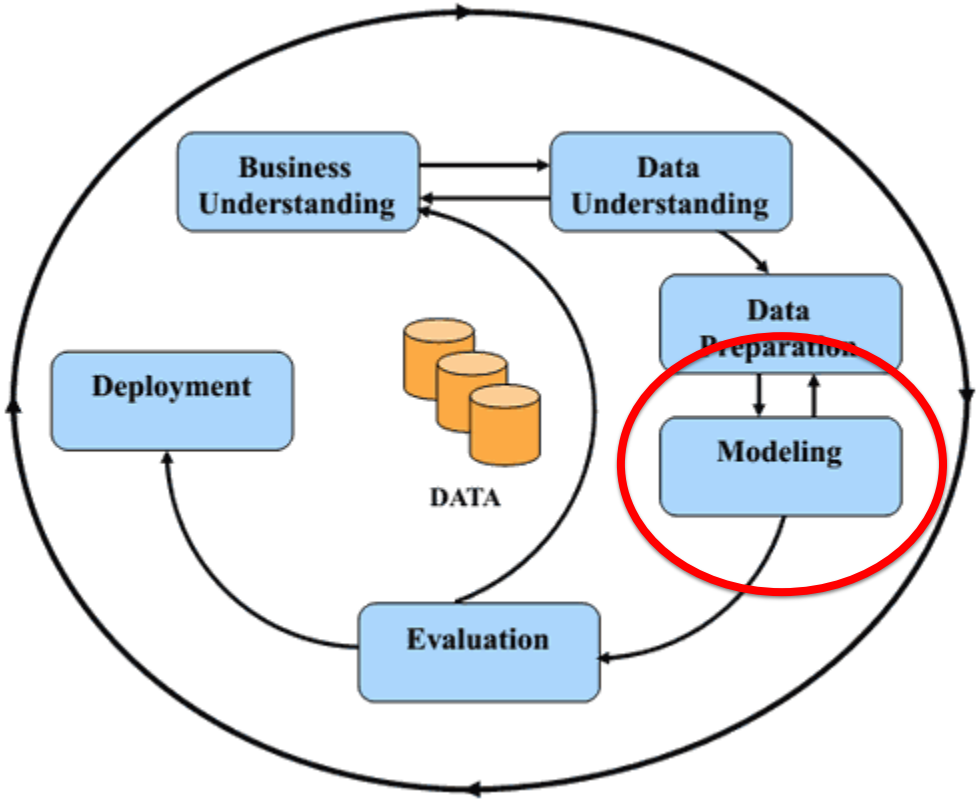
•**Precisão**: A exatidão dos dados garante que eles refletem corretamente a realidade ou a fonte original.

•**Atualidade**: Dados devem estar atualizados, refletindo informações relevantes no momento da análise.

•**Validade**: Os dados devem cumprir restrições e regras definidas (e.g., formatos, intervalos de valores).

•**Integridade**: Os relacionamentos entre diferentes partes dos dados devem ser mantidos, garantindo que não haja lacunas em sistemas conectados.

Modeling



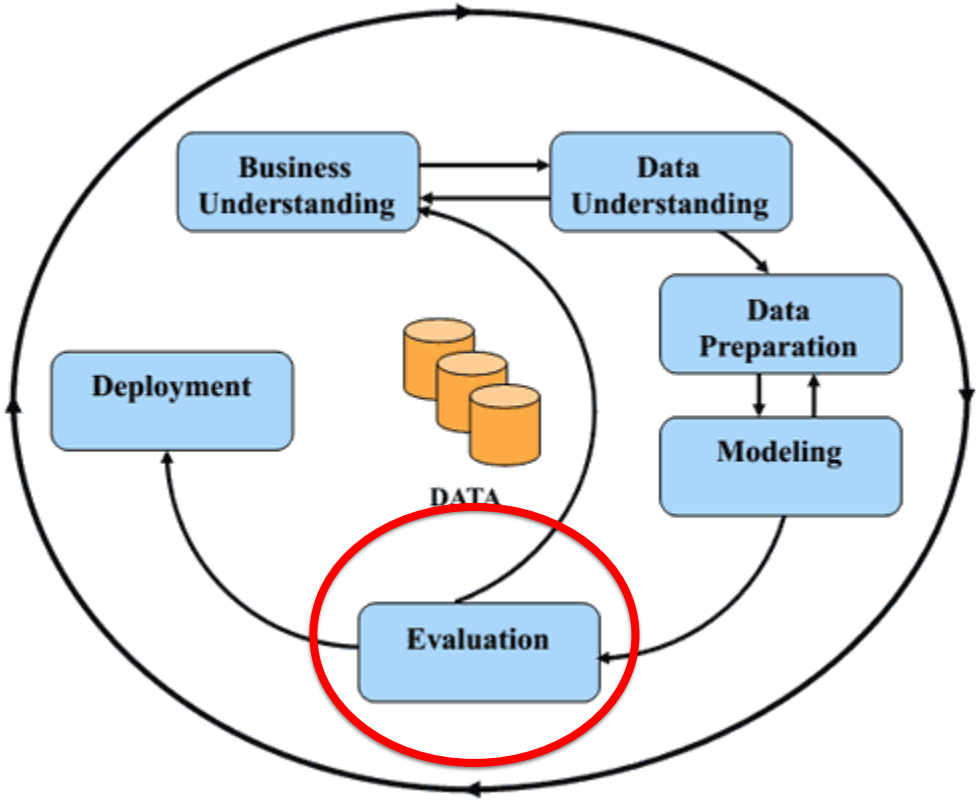
Principais Objetivos

Preditiva vs Prescriptiva

Preditiva: Modelos de previsão (regressão), classificação

Prescriptiva: Achar padrões, entender os clusters, criar insights

Evaluation



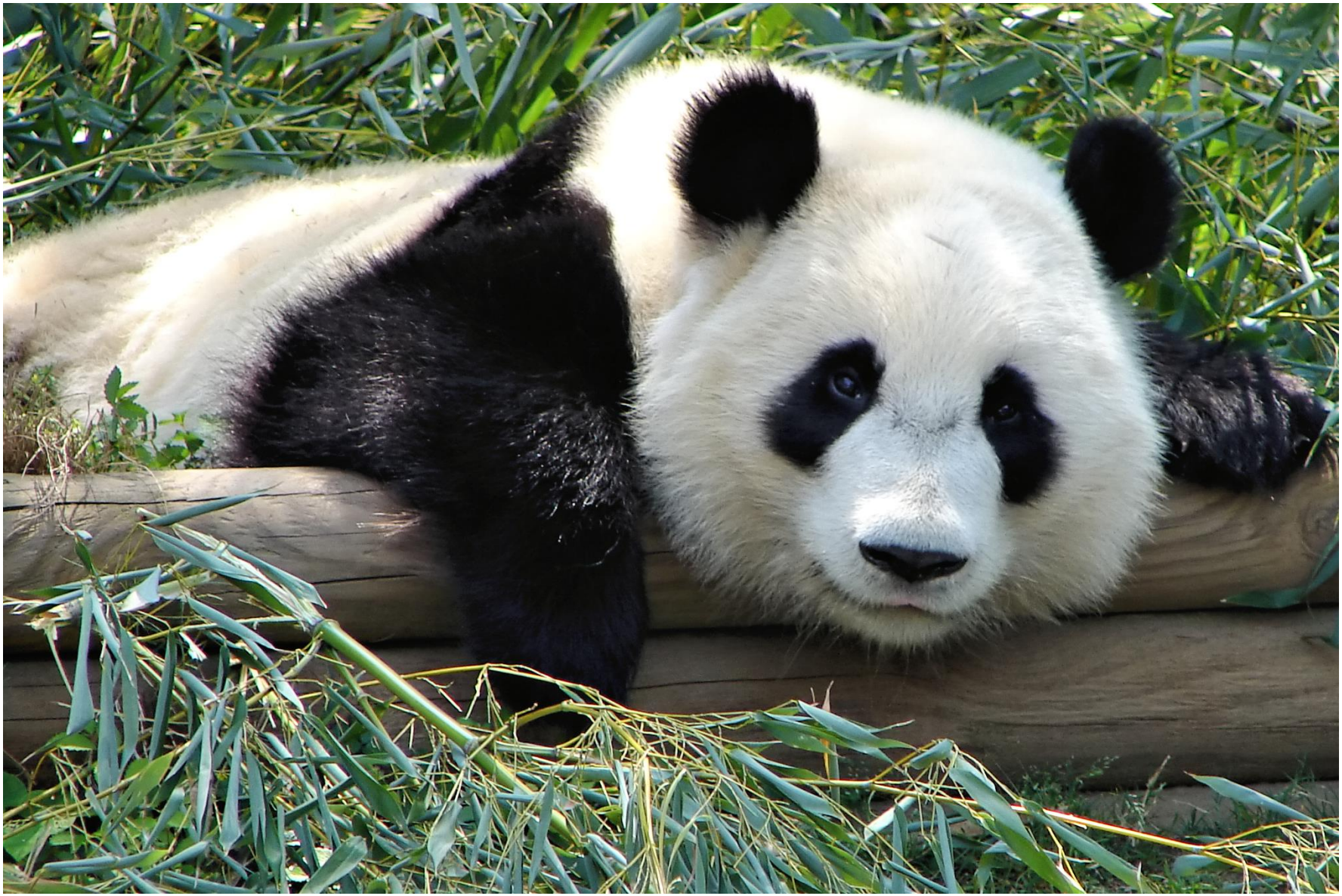
Principais Objetivos

Como avaliar?

Análise dos modelos (cross-validation, error analysis etc)

Análise Subjetiva – análise gráfica, dashboard

Voltando para Python…



Pandas – Introdução

- Pandas é **uma das principais bibliotecas** usadas em Data Science.
- É uma **ferramenta de análise e manipulação de dados**, rápida, flexível e fácil de utilizar, construída com base na linguagem Python.

pandas

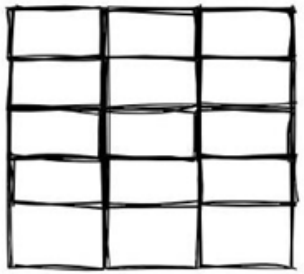
Pandas – *"import pandas as pd"*

Os principais objetos Pandas são as Series e os Dataframes:

Series

- Matriz (array) com 1 só dimensão
- Pode conter qualquer tipo e formato de dados (int, float, str…)
- É declarado com pd.Series()

DataFrame



- Matriz (array) com 2 dimensões (linhas e colunas)
- Cada coluna pode conter diferentes formatos de dados (int, float, str…)
- É declarado com pd.DataFrame()

Pandas - Series

Exemplo: média final das notas de matemática

```
import pandas as pd
```

→ Importar biblioteca

```
new_list = [10,8,12,17,7,19,16,11,13]
new_series = pd.Series(new_list)
new_series
```

→ Definir uma Serie

```
0    10
1     8
2    12
3    17
4     7
5    19
6    16
7    11
8    13
dtype: int64
```

→ Podemos perceber que a Serie tem o tipo integer (int)

Pandas - Series

Exemplo: anos de nascimento dos trabalhadores de uma empresa

```
import pandas as pd
```

Importar biblioteca

```
new_list = [1992, 1999, 2002, 1980, "2000"]
new_series = pd.Series(new_list)
new_series
```

Definir uma Serie

```
0    1992
1    1999
2    2002
3    1980
4    2000
dtype: object
```

Caso os elementos da Serie tenha mais que 1 formato, toda a Serie é convertida para o mesmo formato. Neste caso, para string (object)
No pandas, o formato “str” é “object”

Pandas - DataFrame

- Um **DataFrame é um conjunto de Series**, onde cada Serie corresponde a uma coluna no DataFrame.
- Podemos atribuir um nome a cada coluna, permitindo-nos **guardar e identificar conjuntos de dados** de grande dimensão.

Series

```
0    A
1    B
2    C
Name: id, dtype: object
```

+

Series

```
0    10
1    20
2    30
Name: value, dtype: int64
```

=

DataFrame

	id	value
0	A	10
1	B	20
2	C	30

Pandas - DataFrames

Exemplo: informação sobre os trabalhadores de uma empresa

```
import pandas as pd
```

```
genero = ['M', 'F', 'F', 'M', 'M']
idade = [24, 30, 31, 28, 40]
cargo = ['Analista Junior', 'Manager', 'RH', 'Analista Senior', 'CEO']

df = pd.DataFrame({'genero': genero,
                   'idade': idade,
                   'cargo': cargo})

df
```

O DataFrame é definido com um dicionário.

	genero	idade	cargo
0	M	24	Analista Junior
1	F	30	Manager
2	F	31	RH
3	M	28	Analista Senior
4	M	40	CEO

Vamos praticar!