Introdução a manipulação de dados em Pandas

Dataset Titanic

Nesta webaula vamos utilizar um dos datasets (conjunto de dados) clássicos para quem inicia o estudo na área de ciência de dados, o Titanic. Essa base foi inicialmente lançada em um desafio do portal kaggle.com (https://www.kaggle.com/) e possui 8 colunas, conforme a tabela a seguir.

Amostra do dataset Titanic.csv

	Survived	Pclass	Name	Sex	Age	Siblings/Spouses Aboard	Parents/Children Aboard	Fare
0	0	3	Mr. Owen Harris Braund	male	22.0	1	0	7.2500
1	1	1	Mrs. John Bradley (Florence Briggs Thayer Cumings	female	38.0	1	0	71.2833
2	1	3	Miss. Laina Heikkinen	female	26.0	0	0	7.9250
3	1	1	Mrs Jacques Heath (Lily May Peel) Futrelle	female	35.0	1	0	53.10000
4	0	3	Mr. William Henry Allen	male	35.0	0	0	8.0500

Fonte: adaptada de https://stanford.io/3kDR5Tn.

Seleção de colunas

1 . Para selecionar uma coluna usa-se a sintaxe: `meu_df['coluna']`

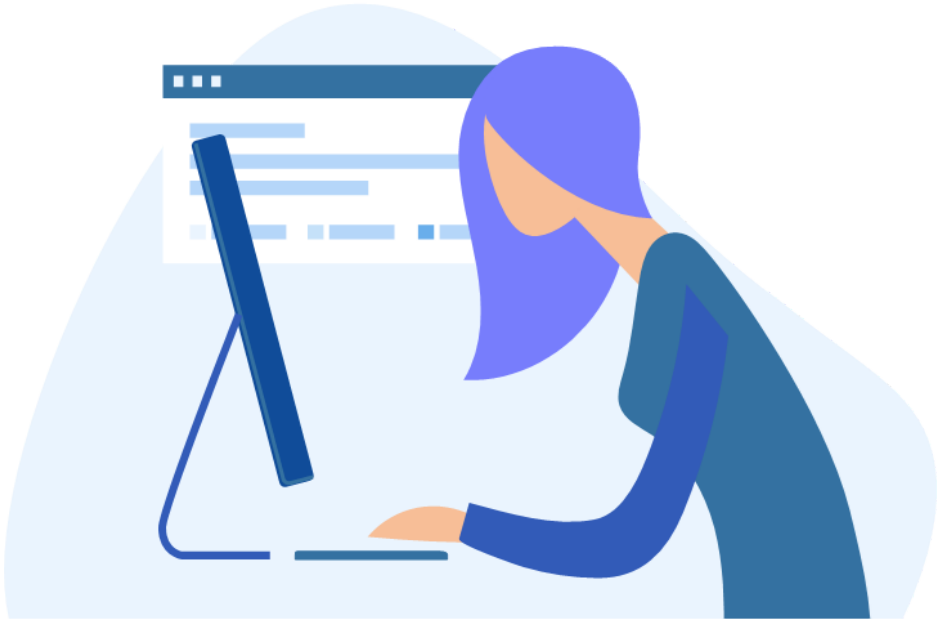
```
df_titanic['Age']
df_titanic['Survived']
```

2. Para selecionar mais de uma coluna é preciso passar uma lista de colunas: `meu_df[['coluna1', 'coluna2', 'coluna3']]`

```
df_titanic[['Age', 'Fare']]
df_titanic[['Name', 'PClass', 'Fare']]
```

Seleção de linhas - Filtros

Um dos recursos mais utilizados por equipes das áreas de dados é a aplicação de filtros. Imagine a seguinte situação: uma determinada pesquisa quer saber qual é a média de idade de todas as pessoas na sua sala de aula, bem como a média de idades somente das mulheres e somente dos homens. A distinção por gênero é um filtro! Esse filtro vai possibilitar comparar a idade de todos com a idade de cada grupo e entender se as mulheres ou os homens estão abaixo ou acima da média geral.



Fonte: Shutterstock.

DataFrames da biblioteca pandas possuem uma propriedade chamada **loc**. Essa propriedade permite acessar um conjunto de linhas (**filtrar linhas**), por meio do índice ou por um vetor booleano (vetor de True ou False).

Saiba mais

» Ao criar uma condição booleana para os dados de uma colun, obtém-se uma Series de valores True ou False.

» Ao usar essa Series como parâmetro no loc, somente os registros que tiverem valor True são exibidos.

Exemplo:

Filtrar somente os homens que estavam a bordo e guardar dentro de um novo DataFrame.

```
filtro_homem = df_titanic['Sex'] == 'male'
df_titanic_homens = df_titanic.loc[filtro_homem]
```

Ou ainda, criar um novo DataFrame somente com os passageiros que sobreviveram:

```
filtro_sobreviventes = df_titanic['Survived'] == 1
df_titanic_sobreviventes =
df_titanic[filtro_sobreviventes]
```

O filtro sempre é criado com base em condições sobre uma ou mais colunas.

Filtros com operadores relacionais e lógicos

É possível criar filtros usando operadores relacionais e lógicos para criar condições compostas. Cada condição deve estar entre parênteses e deve ser conectada pelos operadores lógicos **AND** (**&**), **OR** (**|**).

Por exemplo, criar um novo DataFrame contendo todos os homens que sobreviveram.

```
filtro_sobreviventes = df_titanic['Survived'] == 1
filtro_homem = df_titanic['Sex'] == 'male'

df_titanic_homens_sobreviventes = df_titanic.loc[(filtro_sobreviventes) &
(filtro_homem)]
```

Ou um novo DataFrame com todos os passageiros do sexo feminino que estavam na primeira ou segunda classe. Nesse caso, é preciso utilizar um parênteses extra para garantir a ordem de execução: primeiro faz o **OU** entre pessoas que estavam na primeira e segunda classe e depois faz o **E** com pessoas do sexo feminino.

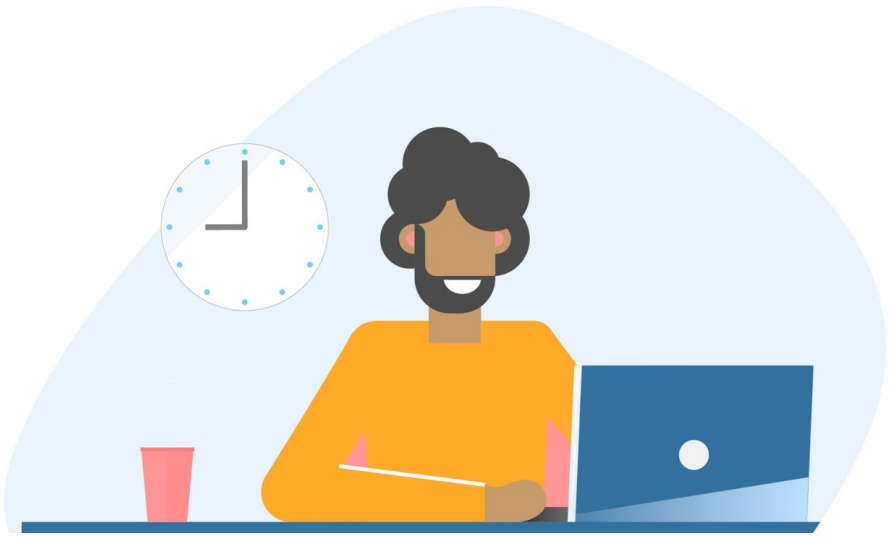
```
filtro_mulher = df_titanic['Sex'] == 'female'
filtro_classe1 = df_titanic['Pclass'] == 1
filtro_classe2 = df_titanic['Pclass'] == 2

df_titanic_mulheres_c1_c2 = df_titanic.loc[(filtro_mulher) & ((filtro_classe1) |
(filtro_classe2))]
df_titanic_mulheres_c1_c2
```

Pesquise mais

Na documentação oficial da biblioteca pandas você encontrará mais exemplos de como utilizar a propriedade loc. Veja como selecionar linhas e colunas em uma única linha de comando.

pandas Team. **pandas.DataFrame.loc()**. Disponível em: https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.loc.html. Acesso em: 24 jun. 2020.



Fonte: Shutterstock.

Desafio extra

Titanic: Machine Learning from Disaster

Para praticarmos e aprendermos um pouco mais sobre os DataFrames, vamos continuar usando a base de dados do Titanic, do famoso desafio “Titanic: Machine Learning from Disaster” (https://www.kaggle.com/c/titanic), o qual utiliza um algoritmo de machine learning para tentar prever quem sobreviveu no desastre do Titanic em 15 de abril 1912.

Acesse aqui o desafio extra