

IMD0033 - Probabilidade

Aula 05 - Introdução a Python II

Ivanovitch Silva
Março, 2018



Agenda

Conjuntos (set)
Imputação de dados (faltantes ou perdidos)
Tratamento de erro
Enumeradores (enumerate)
Compreensão de listas
Desafio

Legislators.csv
nfl_suspensions_data.csv

Atualizar o repositório

```
git clone https://github.com/ivanovitchm/imd0033_2018_1.git
```

Ou

```
git pull
```

Conjuntos

Quando trabalhamos com dados, em alguns momentos pode ser interessante extrair dados únicos de um determinado conjunto.

["Dog", "Cat", "Hippo", "Dog", "Cat", "Dog", "Dog", "Cat"]



["Dog", "Cat", "Hippo"]

Conjuntos (set)

```
unique_animals = set(["Dog", "Cat", "Hippo", "Dog", "Cat", "Dog", "Cat"])  
print(unique_animals)
```

```
unique_animals.add("Tiger")
```

```
unique_animals.remove("Dog")
```

```
unique_animals.add("Tiger")
```

Imputação de dados

O que fazer quando os dados estão incompletos?

- Remover
- Popular com um dado valor
- Estimar valores

```
rows = [  
    ["Bassett", "Richard", "1745-04-02", "M", "sen", "DE", "Anti-Administrat  
ion"],  
    ["Bland", "Theodorick", "1742-03-21", "", "rep", "VA", ""]  
]  
  
for row in rows:  
    if row[6] == "":  
        row[6] = "No Party"
```

Tratamento de erros

```
numbers = [1,2,3,4,5,6,7,8,9,10]
for i in numbers:
    try:
        int('')
    except Exception:
        print("There was an error")
```

Enumeradores

```
animals = ["Dog", "Tiger", "SuperLion", "Cow", "Panda"]  
viciousness = [1, 5, 10, 10, 1]  
for animal in animals:  
    print("Animal")  
    print(animal)  
    print("Viciousness")
```

???? como imprimir *viciousness*??

Enumeradores

```
animals = ["Dog", "Tiger", "SuperLion", "Cow", "Panda"]
viciousness = [1, 5, 10, 10, 1]
for i, animal in enumerate(animals):
    print("Animal")
    print(animal)
    print("Viciousness")
    print(viciousness[i])
```

Compreensão de Listas

```
animals = ["Dog", "Tiger", "SuperLion", "Cow", "Panda"]  
animal_lengths = []  
for animal in animals:  
    animal_lengths.append(len(animal))
```



```
animal_lengths = [len(animal) for animal in animals]
```



```
index.js
import React, { useState } from 'react';
import './index.css';

function App() {
  const [contacts, setContacts] = useState([]);
  const [name, setName] = useState('');
  const [phone, setPhone] = useState('');
  const [email, setEmail] = useState('');

  const handleSubmit = (e) => {
    e.preventDefault();
    setContacts([...contacts, { name, phone, email }]);
    setName('');
    setPhone('');
    setEmail('');
  };

  return (
    <div>
      <h1>Contact Manager</h1>
      <div>
        <input type="text" value={name} onChange={e => setName(e.target.value)} />
        <input type="text" value={phone} onChange={e => setPhone(e.target.value)} />
        <input type="text" value={email} onChange={e => setEmail(e.target.value)} />
        <button onClick={handleSubmit}>Add Contact</button>
      </div>
      <ul>
        {contacts.map((contact) => (
          <li>
            {contact.name} {contact.phone} {contact.email}
            <button onClick={() => setContacts(contacts.filter(c => c !== contact))}>Remove</button>
          </li>
        ))}
      </ul>
    </div>
  );
}

export default App;
```

```
index.html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>Contact Manager</title>
  </head>
  <body>
    <div id="root">
      <div>
        <h1>Contact Manager</h1>
        <div>
          <input type="text" value="" />
          <input type="text" value="" />
          <input type="text" value="" />
          <button>Add Contact</button>
        </div>
        <ul>
          <li></li>
        </ul>
      </div>
    </div>
  </body>
</html>
```