

05

Os códigos abaixo produzem o mesmo resultado.
Você consegue identificar as linguagens usadas?
O que fazem estes códigos?
Quais as semelhanças e diferenças entre eles?

```
people = [{'name': 'John', 'age': 25}, {'name': 'Jane', 'age': 30}, {'name': 'Doe', 'age': 22}]

for i in range(len(people)):
    for j in range(i + 1, len(people)):
        if people[i]['age'] > people[j]['age']:
            people[i], people[j] = people[j], people[i]

print(people)
```

```
people = [{'name': 'John', 'age': 25}, {'name': 'Jane', 'age': 30}, {'name': 'Doe', 'age': 22}]

sorted_people = sorted(people, key=lambda person: person['age'])

print(sorted_people)
```

```
import Data.List (sortBy)

people :: [(String, Int)]
people = [("John", 25), ("Jane", 30), ("Doe", 22)]

sortedPeople :: [(String, Int)]
sortedPeople = sortBy (\(_, age1) (_, age2) -> compare age1 age2) people

main :: IO ()
main = print sortedPeople
```

```
people = [{"John", 25}, {"Jane", 30}, {"Doe", 22}]

sorted_people = Enum.sort_by(people, fn {_, age} -> age end)

IO.inspect(sorted_people)
```