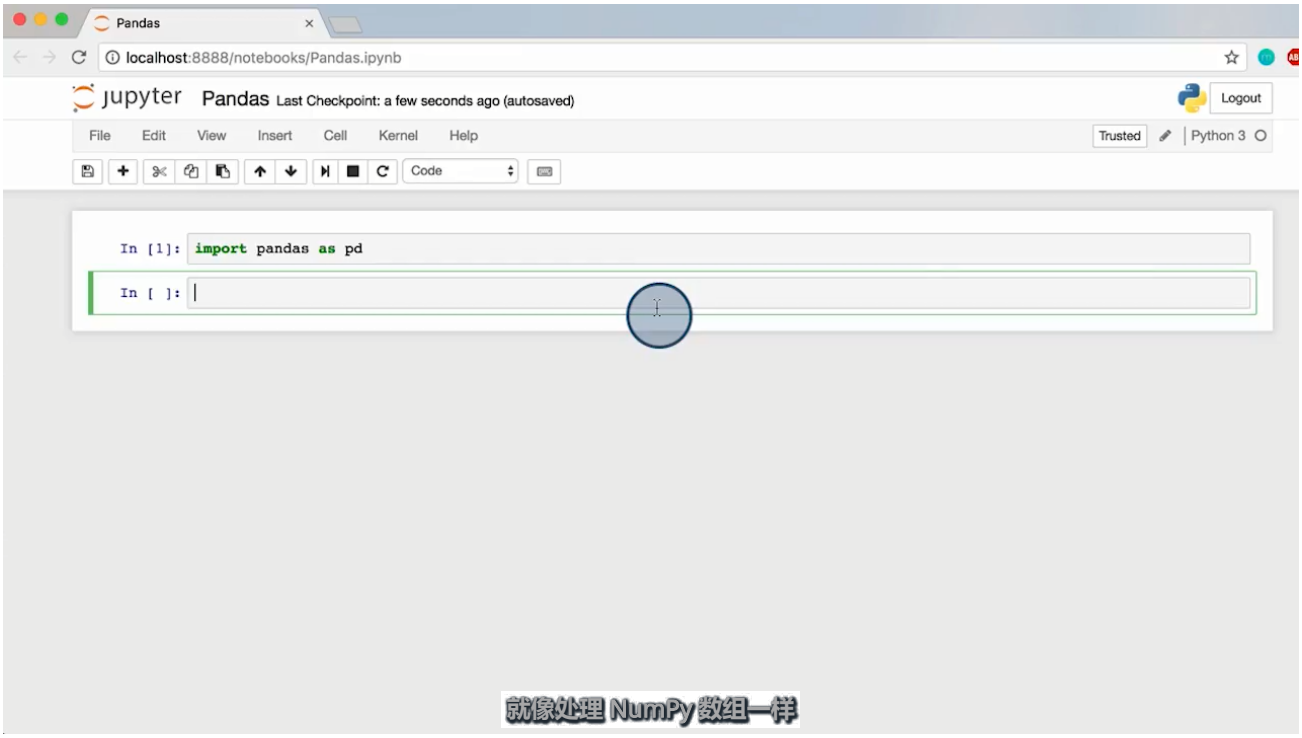
对 Pandas Series 执行算术运算



00:00 / 02:11 1x CC

和 NumPy ndarray 一样，我们可以对 Pandas Series 执行元素级算术运算。在这节课，我们将了解 Pandas Series 和单个数字之间的算术运算。我们创建一个新的 Pandas Series，用于存储只有水果的购物清单。

```
# We create a Pandas Series that stores a grocery list of just fr
fruits= pd.Series(data = [10, 6, 3,], index = ['apples', 'oranges

# We display the fruits Pandas Series
fruits
```

```
apples      10
oranges      6
bananas      3
dtype: int64
```

```
# We print fruits for reference
print('Original grocery list of fruits:\n ', fruits)

# We perform basic element-wise operations using arithmetic symbo
print()
print('fruits + 2:\n', fruits + 2) # We add 2 to each item in fru
print()
print('fruits - 2:\n', fruits - 2) # We subtract 2 to each item i
print()
print('fruits * 2:\n', fruits * 2) # We multiply each item in fru
print()
print('fruits / 2:\n', fruits / 2) # We divide each item in fruit
print()
```

```
Original grocery list of fruits:
 apples     10
oranges      6
bananas      3
dtype: int64

fruits + 2:
apples     12
oranges     8
bananas     5
dtype: int64

fruits - 2:
apples     8
oranges    4
bananas    1
dtype: int64
```

> *oranges 12*
> *bananas 6*
> *dtype: int64*

> *fruits / 2:*
> *apples 5.0*
> *oranges 3.0*
> *bananas 1.5*
> *dtype: float64*

我们还可以对 Pandas Series 中的所有元素应用 NumPy 中的数学函数，例如 `sqrt(x)` 。

```
# We import NumPy as np to be able to use the mathematical functi
import numpy as np

# We print fruits for reference
print('Original grocery list of fruits:\n', fruits)

# We apply different mathematical functions to all elements of fr
print()
print('EXP(X) = \n', np.exp(fruits))
print()
print('SQRT(X) =\n', np.sqrt(fruits))
print()
print('POW(X,2) =\n',np.power(fruits,2)) # We raise all elements
```

> *Original grocery list of fruits:*
> *apples 10*
> *oranges 6*
> *bananas 3*
> *dtype: int64*

> *oranges 403.428793*
> *bananas 20.085537*
> *dtype: float64*

> *SQRT(X) =*
> *apples 3.162278*
> *oranges 2.449490*
> *bananas 1.732051*
> *dtype: float64*

> *POW(X,2) =*
> *apples 100*
> *oranges 36*
> *bananas 9*
> *dtype: int64*

Pandas 还允许我们仅对 fruits 购物清单中的部分条目应用算术运算。我们来看一些示例：

```
# We print fruits for reference
print('Original grocery list of fruits:\n ', fruits)
print()

# We add 2 only to the bananas
print('Amount of bananas + 2 = ', fruits['bananas'] + 2)
print()

# We subtract 2 from apples
print('Amount of apples - 2 = ', fruits.iloc[0] - 2)
print()

# We multiply apples and oranges by 2
print('We double the amount of apples and oranges:\n', fruits[['a
print()
```

> *Original grocery list of fruits:*
> *apples 10*
> *oranges 6*
> *bananas 3*
> *dtype: int64*

> *Amount of bananas + 2 = 5*

> *Amount of apples - 2 = 8*

> *We double the amount of apples and oranges:*
> *apples 20*
> *oranges 12*
> *dtype: int64*

> *We half the amount of apples and oranges:*
> *apples 5.0*
> *oranges 3.0*
> *dtype: float64*

你还可以对具有混合数据类型的 Pandas Series 应用算术运算，前提是该算术运算适合 Series 中的*所有*数据类型，否则会出错。我们来看看将购物清单乘以 2 会发生什么

```
# We multiply our grocery list by 2
groceries * 2
```

> *eggs 60*
> *apples 12*
> *milk YesYes*

可以看出，在上述示例中，我们乘以了 2，Pandas 使每个条目的数据翻倍，包括字符串。Pandas 能够这么操作是因为，乘法运算 `*` 对数字和字符串来说都可行。如果你要应用对数字有效但是对字符串无效的运算，例如 `/`，则会出错。如果 Pandas Series 中有混合类型的数据，确保对于*所有*的元素数据类型，这些算术运算都有效。