

12章 関数の引数・戻り値を理解しよう

12章 関数の引数・戻り値を理解しよう

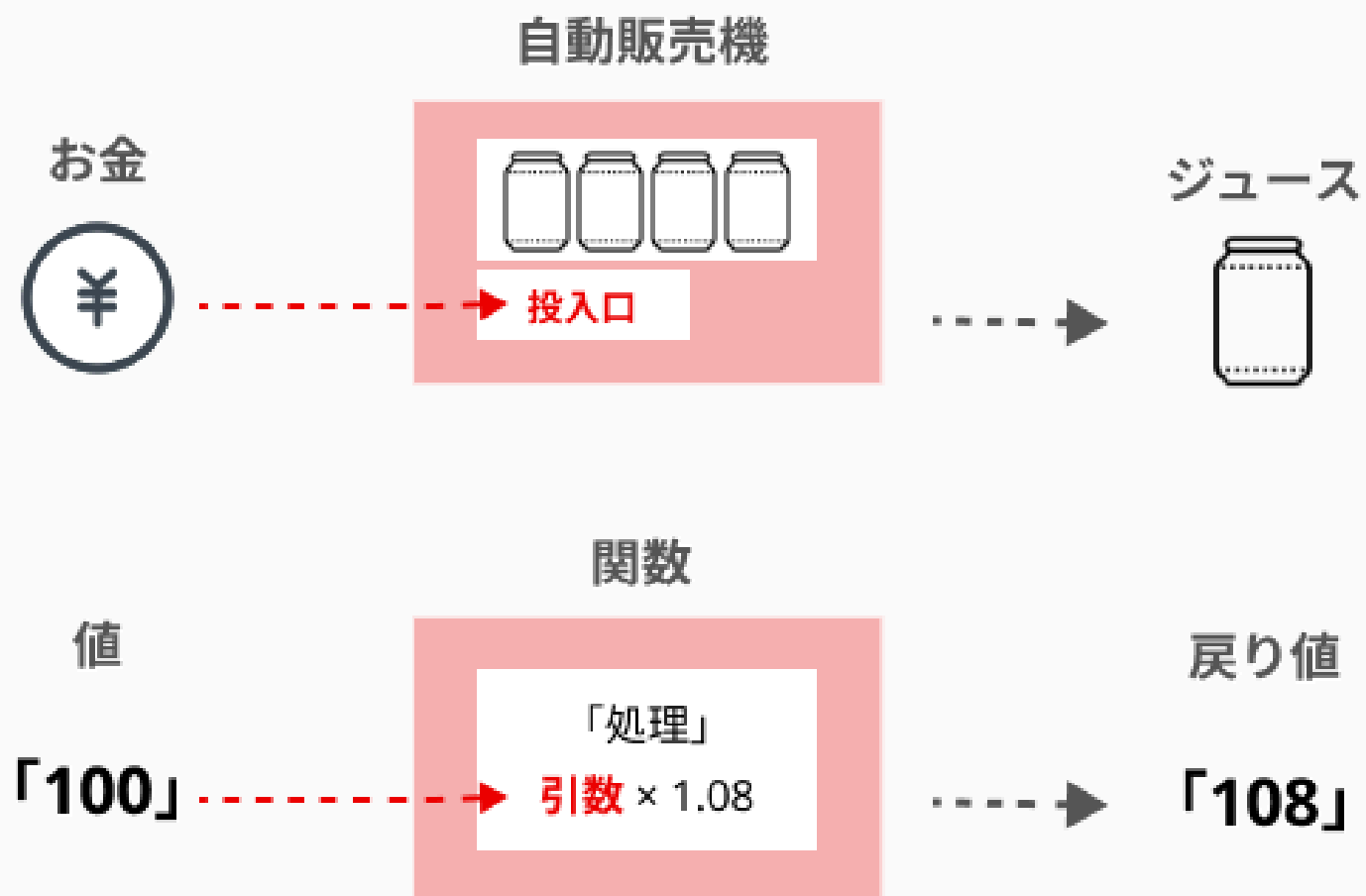
関数で使う引数・戻り値と型ヒントについて学びます

本章の目標

- 引数について学び、関数をより柔軟に使いこなせるようになること
- 戻り値について学び、関数から処理結果を受け取る方法を知ること
- 引数や戻り値のデータ型を指定する「型ヒント」について知ること

| 引数（ひきすう）とは、関数に与えるデータのことです

12章 引数とは



12章 引数が必要な場面

決まっていない値を使って処理を行いたいときに引数が必要です

- 朝のあいさつを出力する関数 → 決まった処理なので引数は不要
- 購入金額に送料を加算する関数 → 購入金額は決まっていない値なので引数が必要

購入金額に送料を加算する関数



関数の丸括弧()内に引数名を記述します

```
def 関数名(引数名):  
    引数を使った一連の処理
```

- 関数の定義時にはまだ引数の具体的な値がわからない
- 変数を使って処理を作っておくイメージ

12章 引数の使い方

購入金額に送料を加算する関数の例

```
def calculate_total(price):  
    # 与えられた引数priceに送料を加算し、変数totalに代入する  
    total = price + 500  
  
    # 変数totalの値を出力する  
    print(f"{total}円")
```


12章 引数を渡して関数を呼び出す

関数を呼び出すときに引数を記述します

```
def calculate_total(price):  
    total = price + 500  
    print(f"{total}円")  
  
# 関数を呼び出し、引数として購入金額を渡す  
calculate_total(1200)
```

- 関数の()内に引数を記述することを「引数を渡す」と表現する

12章 引数を渡して関数を呼び出す

✓
0
秒



与えられた引数\$priceに送料を加算し、その値を出力する関数を定義する

```
def calculate_total(price):
```

与えられた引数priceに送料を加算し、変数totalに代入する

```
total = price + 500
```

変数totalの値を出力する

```
print(f"{total}円")
```

関数を呼び出し、引数として購入金額を渡す

```
calculate_total(1200)
```

1700円

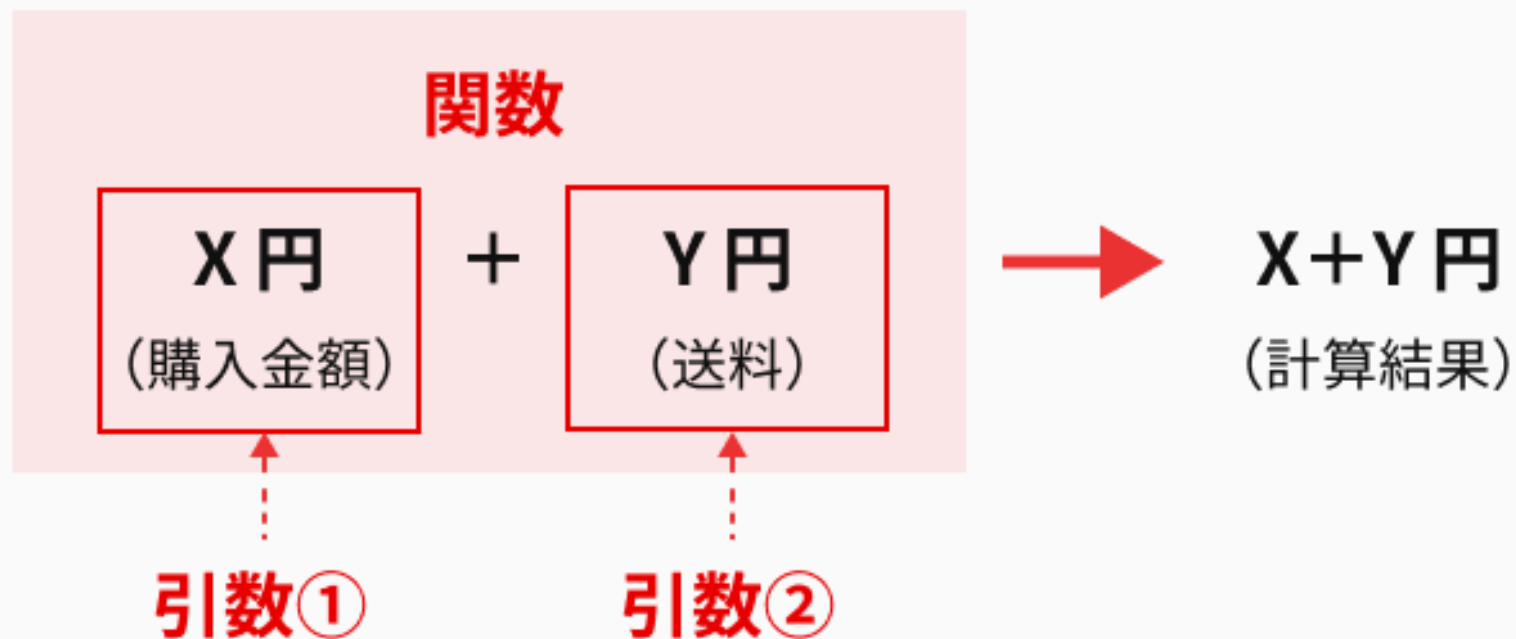
12章 引数を渡して関数を呼び出す

(例) 購入金額が1200円の場合



12章 複数の引数を受け取る関数

引数が複数の場合



12章 複数の引数を受け取る関数

カンマ区切りで引数を記述します

```
def add_two_arguments(price, shipping_fee):  
    # 与えられた引数priceと引数shipping_feeを加算  
    total = price + shipping_fee  
    print(f"{total}円")  
  
# 関数を呼び出し、引数として購入金額と送料を渡す  
add_two_arguments(1200, 500)
```

12章 複数の引数を受け取る関数

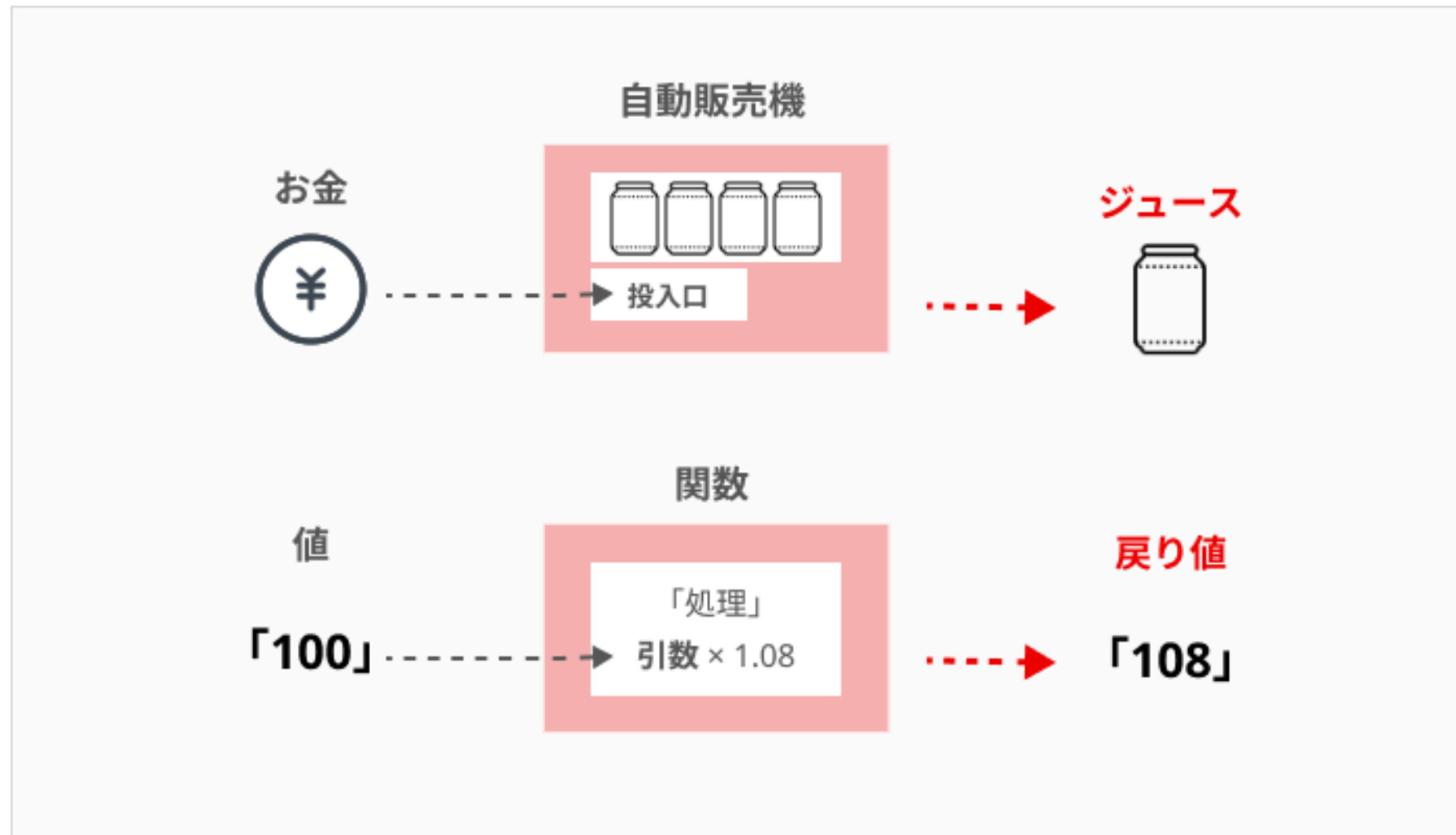
✓
0
秒



```
def add_two_arguments(price, shipping_fee):  
    # 与えられた引数priceと引数shipping_feeを加算し、変数totalに代入する  
    total = price + shipping_fee  
  
    # 変数$totalの値を出力する  
    print(f"{total}円")  
  
# 関数を呼び出し、引数として購入金額と送料を渡す  
add_two_arguments(1200, 500);
```

1700円

12章 関数の戻り値



12章 戻り値とは

関数から返ってくる値のことを戻り値（または返り値）といいます

- 戻り値を返さなければ関数だけで処理が終わる
- 戻り値を返すことで他のコードに活用できる
- `return`を記述すれば戻り値を返せる

12章 戻り値を使った関数の例

商品が購入されたかどうかチェックする関数

```
# 購入済みかどうかを判定する変数
purchased = True

# 戻り値を返す関数を定義する
def is_purchased():
    if (purchased):
        return True
    else:
        return False

# 戻り値を返す関数を条件式に使う
if (is_purchased()):
    print("商品は購入済みです。")
```

12章 戻り値を使った関数の例

✓
1
秒



購入済みかどうかを判定する変数

purchased = True

戻り値を返す関数を定義する

def is_purchased():

if (purchased):

return True

else:

return False

戻り値を返す関数を条件式に使う(Trueであれば処理が実行される)

if (is_purchased()):

print("商品は購入済みです。")



商品は購入済みです。

| 本章では以下の内容を学習しました

引数

- 引数（ひきすう）とは、関数に与えるデータのこと
- 関数には複数の引数を渡すこともできる

戻り値

- 関数内でreturnを記述すれば、戻り値を返せる
- 戻り値を返すことで他のコードに活用できる