## Shop

Create a class called **Shop**. Upon initialization it should receive a **name (str)**, **type (str)**, **capacity (int)**. The store should also have an **attribute** called **items** (**dictionary** that stores **name** of an item and its **quantity**). The class should have **4 methods**:

* **small\_shop(name: str, type: str)** – a **new shop with capacity of 10** should be created
* **add\_item(item\_name:str)** - adds **1** to the quantity of the given **item**. On **success**, the method should **return "{item\_name} added to the shop"**. If the addition is **not possible**, the following message should be returned **"Not enough capacity in the shop"**
* **remove\_item(item\_name:str, amount:int)** - **removes** the given amount from the **item**. On **success**, it should return **"{amount} {item\_name} removed from the shop"**. **Otherwise**, the method should return **"Cannot remove {amount} {item\_name}"**
* **\_\_repr\_\_()** - returns a string representation in the format **"{shop\_name} of type {shop\_type} with capacity {shop\_capacity}"**

### Examples

|  |
| --- |
| **Test Code** |
| fresh\_shop = Shop("Fresh Shop", "Fruit and Veg", 50)  small\_shop = Shop.small\_shop("Fashion Boutique", "Clothes")  print(fresh\_shop)  print(small\_shop)  print(fresh\_shop.add\_item("Bananas"))  print(fresh\_shop.remove\_item("Tomatoes", 2))  print(small\_shop.add\_item("Jeans"))  print(small\_shop.add\_item("Jeans"))  print(small\_shop.remove\_item("Jeans", 2)) |
| **Output** |
| Fresh Shop of type Fruit and Veg with capacity 50  Fashion Boutique of type Clothes with capacity 10  Bananas added to the shop  Cannot remove 2 Tomatoes  Jeans added to the shop  Jeans added to the shop  2 Jeans removed from the shop |