You are to create a simple banking application using Python classes.

The objects we care about in this bank are:

- Person
- Bank
- Client
- Account
- Transaction

The relationships between these objects are:

- A bank has one or more clients, a client belongs to only one bank. A bank has a name, a number of clients, a location and total amount of transactions
- A client is a Person, not all people are clients
- A person has the following attributes: name, cni_number, neighborhood, profession
- A client has all the above attributes as well as a bank.
- A client has one or more accounts. An account belongs to one client
- An account has a type (current or savings). It also has a balance and a number of transactions attribute.
- A transaction is linked to an account, an account has 0 or more transactions.
- A transaction has an amount, a type (credit or debit), and is linked to an account.

Complete the tasks below to implement this functionality:

Create the Person class

Create the Person class with the attributes and the appropriate `__init__` method.

Create the Bank class

- Create a class attribute called `list_of_banks` which will be an empty list
- Write the `__init__` method, initialize the appropriate attributes for the object.
    - In the `__init__` method, add the object to the `list_of_banks` class attribute using the append method of the list. Number of clients and total amount of transactions are 0 by default.
- Write a `__del__` method that removes the object from the `list_of_banks` class attribute.

Create the Client class

- Write the `__init__` method that takes as argument, a `person` object and a `bank` object. Write the code to create the corresponding object attributes
  - In the `__init__` method, increase the `number_of_clients` attribute of the bank by 1
- Write a **factory method** called `create_client` that takes as argument: name, cni_number, neighborhood, profession, and bank.
  This method will create a person object with the attributes, and then creates and returns the client object with this person object and the bank.
- Write a destructor that reduces the `number_of_clients` attribute of this client's bank by 1.

Create the Account class

- Write the `__init__` method that takes as argument a client, a type (current or savings), a balance and a number of transactions and creates the corresponding object attributes
- Write 2 factory methods, `create_savings_account` and `create_current_account` that take as argument a client and create `Account` objects with attributes balance (0), number of transactions (0), the client and the corresponding type (current or saving)

Create the Transaction class

- Write the appropriate `__init__` method. After creating the attributes do the following in the `__init__` method (similar to the Transaction class created in class)
  - Add code that increases the number of transactions attribute of the account object by 1.
  - Add code that increases the balance of the account object by the transaction's amount.
- Write a destructor that reduces the number of transactions attribute of the account by 1 and also decreases the account's balance by the transaction's amount.