OBJECT ORIENTED PROGRAMMING WITH PYTHON

Second Class

1st Semester

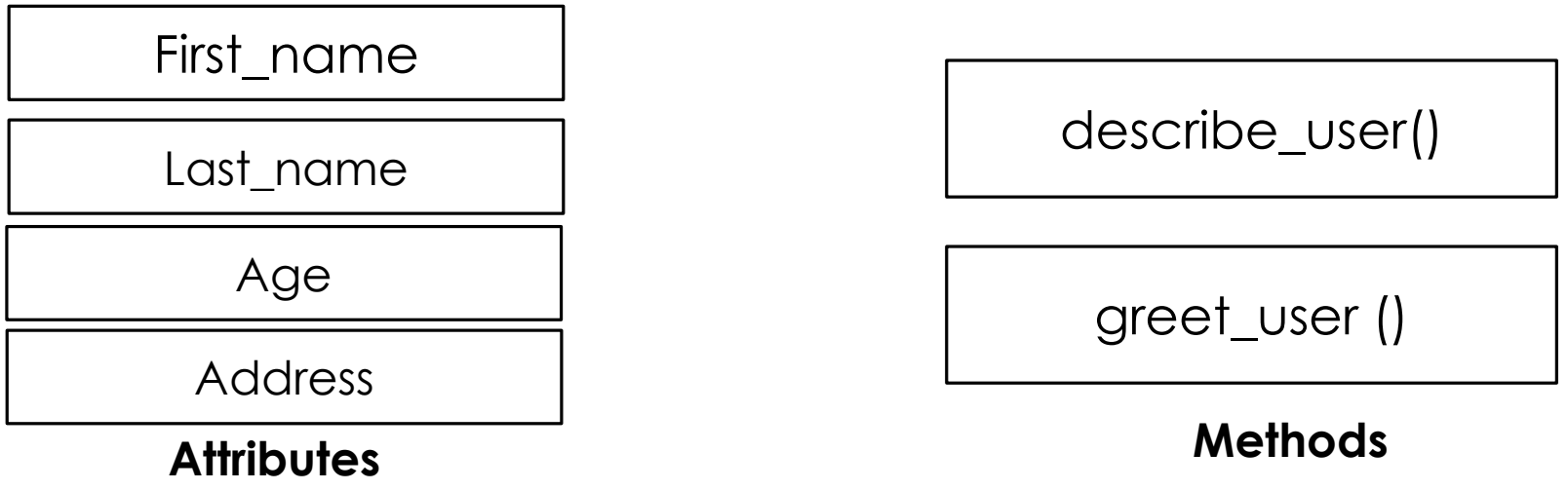
EXAMPLES ON CLASSES AND OBJECTS

- Make a class called **User**.
- Create two attributes called **first_name** and **last_name**, and then create several other attributes that are typically stored in a user profile.
- Make a method called **describe_user()** that prints a summary of the user's information. Make another method called **greet_user()** that prints a personalized greeting to the user.
- Create several instances representing different users, and call both methods
- for each user.

CREATE USER CLASS

Example 1 : Create User Class
class : User
attributes : **first_name** and **last_name**,
methods : **describe_user()** and **greet_user()**

objects : **Ali, Samir, Sara**



IMPROVED USER EXAMPLE

- **Login Attempts**:
- Add an attribute called **login_attempts** to your User class from the previous class
- Write a method called **increment_login_attempts()** that increments the value of login_attempts by 1.
- Write another method called **reset_login_attempts()** that resets the value of **login_attempts** to 0.
- Make an instance of the User class and call **increment_login_attempts**() several times.
- Print the value of **login_attempts** to make sure it was incremented properly, and then call reset_login_attempts().
- Print login_attempts again to make sure it was reset to 0

ADMIN CLASS

- Admin: An administrator is a special kind of user.
- Write a class called **Admin** that inherits from the User class you wrote previously
- Add an attribute, **privileges**, that stores a list of strings like **"can add post", "can delete post", "can ban user"**, and so on.
- Write a method called **show_privileges()** that lists the administrator's set of privileges.
- Create an instance of Admin, and call your method.