

## Exercises: Defining Classes

### Problem 1. Define a Class Person

Define a class **Person** with **public** fields for **Name** and **Age**.

Bonus\*

Try to create a few objects of type Person, using the constructor. Print the objects in the main

Name	Age
John	20
Luke	18
Jane	16

### Problem 2. Creating Constructors

Add 3 constructors to the **Person** class from the last task, use constructor chaining to reuse code:

1. The first should take no arguments and produce a person with name "**No name**" and age = **1**.
2. The second should accept only an integer number for the age and produce a person with name "**No name**" and age equal to the passed parameter.
3. The third one should accept a string for the name and an integer for the age and should produce a person with the given name and age.

### Problem 3. \*Oldest Family Member

Use your Person class from the previous tasks. Create a class Family. The class should have a surname and a list of people, a method for adding members (void AddMember(Person member)) and a method returning the oldest family member (Person GetOldestMember()). Write a program that reads the names and ages of N people and adds them to the family. Then print the name and age of the oldest member.

Example

Input	Output
3 Jane 3 Luke 4 Annie 5	Annie 5

Input	Output
5 Steve 10 Christopher 15 Annie 4 Ivan 35 Maria 34	Ivan 35

### Problem 4. \*Opinion Poll

Using the **Person** class, write a program that reads from the console **N** lines of personal information, stores them in a List and then prints all people whose **age** is **more than 30** years, **sorted in alphabetical order**.

Example

Input	Output
3 Huey 12 Dewey 31 Louie 48	Dewey - 31 Louie - 48
5 Huey 33 Dewey 88 Louie 22 Donald 44 Mickey 11	Donald - 44 Huey - 33 Dewey - 88