"""This is a module for our Person class.  
.. author: Yilmaz Mustafa <yilmaz.mustafa@intecbrussel.be>  
"""  
  
from datetime import date  
  
class Person:  
 """This is a class which represents a person. It is a bit of a silly class.  
 It stores some personal information, and can calculate a person's age.  
 """  
  
 def \_\_init\_\_(self, name, surname, birthdate, address, telephone, email):  
 """This method creates a new person.  
  
 :param name: first name  
 :type name: str  
 :param surname: surname  
 :type surname: str  
 :param birthdate: date of birth  
 :type birthdate: datetime.date  
 :param address: physical address  
 :type address: str  
 :param telephone: telephone number  
 :type telephone: str  
 :param email: email address  
 :type email: str  
 """  
  
 self.name = name  
 self.surname = surname  
 self.birthdate = birthdate  
  
 self.address = address  
 self.telephone = telephone  
 self.email = email  
  
 def age(self):  
 """This method calculates the person's age from the birthdate and the current date.  
  
 :returns: int -- the person's age in years  
 """  
 today = date.today()  
 age = today.year - self.birthdate.year  
  
 if today < date(today.year, self.birthdate.month, self.birthdate.day):  
 age -= 1  
  
 return age  
  
 def email\_domain(self): # voeg de methode argumenten hier (als ze vereist zijn...)  
 return self.email.split("@")[1].lower()  
  
 def \_\_\_str\_\_\_(self):  
 return "Person: " + self.name.capitalize() + " " + self.surname.capitalize() + ", is " + str(self.age()) + " years old.\n\tContact: " + self.address + "." + str(self.telephone) + ", " + self.email.lower()   
  
  
  
person1 = Person("Justin", "Bieber", date.fromisoformat("1986-03-11"), "Brussel Centraal", 46852145678, "just.in@be")  
print(person1.info())  
  
person2 = Person("Nikola", "Tesla", date.fromisoformat("1967-03-11"), "Stad Gent", 4966321185, "nikola.tesla@legends.com")  
print(person2.info())