|  |
| --- |
| **Program 01** |
| **Output** |
| Please enter the name: Scrooge McDuck  Please enter the age: 82  Please enter the address: 123 Easy Street  Please enter the type of person: Millionaire  How many vacation homes does he/she have?4  Go again?Y  Please enter the name: John Miller  Please enter the age: 30  Please enter the address: Ramelle, France  Please enter the type of person: Teacher  How much mortgage is remaining?30000  Go again?Y  Please enter the name: Susan Stressed  Please enter the age: 19  Please enter the address: Apt. 101, Faulty Towers  Please enter the type of person: Student  How much is rent this month?500  Go again?N  Millionaire Scrooge McDuck  Restaurant: Driver, take me to Mannys Steakhouse  Order: Caviar, filet mignon, lobster, and several bottles of your best wine!  What is the bill?300  Bill: Here you go $450.0! And keep the change!  Teacher John Miller  Restaurant: Honey, how about Chilis tonight?  Order: Can I have the special? And how much for a tall beer?  What is the bill?30  Bill: Are you sure 30.0 is correct? OK, here is 34.5  Student Susan Stressed  Restaurant: McDonalds or Culvers?  Order: Burger and fries please!  What is the bill?10  Bill: Can I owe you ten bucks or do the dishes? |
| **Source Code** |
| #This program utilizes the concept of inheritance and has one  #Parent class Person with three child classes Millionaire,  #Student, and Teacher. Based on the information entered the  #program will use that specific class and by polymorphism  #will choose the overloaded methods for that class and not  #the parent class.  class Person(object):  def \_\_init\_\_(self, name, age, address, typePerson):  self.name = name  self.age = age  self.address = address  self.typePerson = typePerson  def restaurant(self):  print('Restaurant: Generic Choice')  def order(self):  print('Order: Generic Order')  def pay\_bill(self, bill):  total = (bill + (bill \* 0.2))  print('Bill: {}'.format(bill))  #Millionaire class inheriting from Person class  class Millionaire(Person):  def \_\_init\_\_(self, name, age, address, typePerson, vacationHomes):  self.name = name  self.age = age  self.address = address  self.typePerson = typePerson  self.vactionHomes = vacationHomes  def restaurant(self):  print('Restaurant: Driver, take me to Mannys Steakhouse')  def order(self):  print('Order: Caviar, filet mignon, lobster, and several bottles of your best wine!')  def pay\_bill(self, bill):  total = (bill + (bill \* 0.5))  print('Bill: Here you go ${}! And keep the change!'.format(total))  #Teacher class inheriting from Person class  class Teacher(Person):  def \_\_init\_\_(self, name, age, address, typePerson, mortgage):  self.name = name  self.age = age  self.address = address  self.typePerson = typePerson  self.mortgage = mortgage  def restaurant(self):  print('Restaurant: Honey, how about Chilis tonight?')  def order(self):  print('Order: Can I have the special? And how much for a tall beer?')  def pay\_bill(self, bill):  total = (bill + (bill \* 0.15))  print('Bill: Are you sure {} is correct? OK, here is {}'.format(bill, total))  #Student class inheriting from Person class  class Student(Person):  def \_\_init\_\_(self, name, age, address, typePerson, rent):  self.name = name  self.age = age  self.address = address  self.typePerson = typePerson  self.rent = rent  def restaurant(self):  print('Restaurant: McDonalds or Culvers?')  def order(self):  print('Order: Burger and fries please!')  def pay\_bill(self, bill):  print('Bill: Can I owe you ten bucks or do the dishes?')  def main():  persons = []  again = 'Y'  ## Input data for different people and add appropriate object to list  while(again == 'Y' or again == 'y'):  name = input('Please enter the name: ')  age = input('Please enter the age: ')  address = input('Please enter the address: ')  typePerson = input('Please enter the type of person: ')  #Series of if-elif-esle statements to determine which class will be used  if(typePerson == 'Millionaire' or typePerson =='millionaire'):  vacationHome = input('How many vacation homes does he/she have?' )  temp = Millionaire(name , age , address , typePerson , vacationHome)  persons.append(temp)  again = input('Go again?')  elif(typePerson == 'Teacher' or typePerson == 'teacher'):  mortgage = int(input('How much mortgage is remaining?'))  temp = Teacher(name, age, address, typePerson, mortgage)  persons.append(temp)  again = input('Go again?')  elif(typePerson == 'Student' or typePerson == 'student'):  rent = int(input('How much is rent this month?'))  temp = Student(name, age, address, typePerson, rent)  persons.append(temp)  again = input('Go again?')  else:  print('That is not a valid entry. Please start over!')  again = 'Y'  print()  ## Display information for people in list  for person in persons:  print(person.typePerson, "", person.name)  person.restaurant()  person.order()  bill = float(input(' What is the bill?'))  person.pay\_bill(bill)  print()    if \_\_name\_\_ == '\_\_main\_\_': main() |

|  |
| --- |
| **Program 02** |
| **Output** |
|  |
| **Source Code** |
|  |

|  |
| --- |
| **Program 03** |
| **Output** |
|  |
| **Source Code** |
|  |

|  |
| --- |
| **Program 04** |
| **Output** |
|  |
| **Source Code** |
|  |

|  |
| --- |
| **Program 05** |
| **Output** |
|  |
| **Source Code** |
|  |