Example 11

Part 1: Create class Person that include:

1. name and age attributes
2. constructor that initialize the attributes
3. Method that takes two objects from class person as parameters then print the name of older person.
4. Method that takes string and integer as parameters then return an object from class person using these parameters.
5. method that takes one object from class person as parameter then change the name to “unknown” and age to 18
6. Method that takes one object from class person as parameter then print the name of older person between the current object and the parameter object.

Part 2: in main method:

1. Create two objects from class person.
2. Print the name of older person from above objects.
3. Define variable from class Person then use the class methods to store object in it
4. Use the method change on last object then print name and age.
5. Use the last method you create in class Person to Print the name of older person from the first and second object.
6. Which methods is better to define as static in class person?

public class Person {

 public String name;

 public int age;

 public Person(String name, int age) {

 this.name = name;

 this.age = age;

 }

 public void older(Person p1,Person p2)

 {

 if(p1.age>p2.age)

 System.out.println("older is: "+p1.name);

 else

 System.out.println("older is: "+p2.name);

 }

 public Person create(String s,int i)

 {

 return new Person(s,i);

 }

 public void change(Person p)

 {

 p.name = "unknown";

 p.age = 18;

 }

 public void older2(Person p)

 {

 if(age>p.age)

 System.out.println("older is: "+name);

 else

 System.out.println("older is: "+p.name);

 }

}

public class JavaApplication50 {

 public static void main(String[] args) {

 Person pp1 = new Person("Ahmad",20);

 Person pp2 = new Person("Sami",30);

 pp2.older(pp1, pp2);

 Person pp3;

 pp3 = pp1.create("Morad", 40);

 System.out.println("name: "+pp3.name+" Age:"+pp3.age);

 pp1.change(pp3);

 System.out.println("name: "+pp3.name+" Age:"+pp3.age);

 pp1.older2(pp2);

 }}

Rewrite the program after change methods to static

public class Person {

 public String name;

 public int age;

 public Person(String name, int age) {

 this.name = name;

 this.age = age;

 }

 public static void older(Person p1,Person p2)

 {

 if(p1.age>p2.age)

 System.out.println("older is: "+p1.name);

 else

 System.out.println("older is: "+p2.name);

 }

 public static Person create(String s,int i)

 {

 return new Person(s,i);

 }

 public static void change(Person p)

 {

 p.name = "unknown";

 p.age = 18;

 }

 public void older2(Person p)

 {

 if(age>p.age)

 System.out.println("older is: "+name);

 else

 System.out.println("older is: "+p.name);

 }

}

public class JavaApplication50 {

 public static void main(String[] args) {

 Person pp1 = new Person("Ahmad",20);

 Person pp2 = new Person("Sami",30);

 Person.older(pp1, pp2);

 Person pp3;

 pp3 = Person.create("Morad", 40);

 System.out.println("name: "+pp3.name+" Age:"+pp3.age);

 Person.change(pp3);

 System.out.println("name: "+pp3.name+" Age:"+pp3.age);

 pp1.older2(pp2);

 }}