TITLE PAGE

Course: CS1073

Section: FR03B

Assignment number: 2

Name: Zohaib Hassan Khan

UNB student number: 3740572

a)

Dog.java:

```
/**
 This class represents a dog.
 @author Zohaib Khan 3740572
*/
public class Dog {

   /**
    The name of the dog.
   */
   private String name;

   /**
    The age of the dog (in years).
   */
   private int age;

   /**
    This method constructs a Dog object with the specified
    name and age.
    @param nameIn the name of the dog.
    @param ageIn the age of the dog (in years).
   */
   public Dog (String nameIn, int ageIn) {
      name = nameIn;
      age = ageIn;
   }

   /**
    This method retrieves the name of the dog.
    @return the name of the dog.
   */
   public String getName () {
      return name;
   }

   /**
    This method retrieves the age of the dog.
    @return the age of the dog (in years).
   */
   public int getAgeDogYears () {
      return age;
   }
   /**
    This method computes and returns the dog's age in
    "person years".
    @return the age of the dog (in person years).
   */
   public int getAgePersonYears () {
      return age * 7;
   }

   /**
    This method is called to change the name of the dog.
    @param nameIn the new name for the dog.
   */
   public void changeName (String nameIn) {
      name = nameIn;
   }

   /**
    This method is called to increase the age of the dog
    by one year.
   */
   public void addYear () {
      age = age + 1;
   }

} //end Dog
```

DogTestDriver.java:

```
/**
 This is a driver program for the dog class.
 @author Zohaib Khan 3740572
*/

public class DogTestDriver {

   public static void main (String[] args){

      // Creating 3 dog objects.
      Dog dog1 = new Dog ("Tom", 3);
      Dog dog2 = new Dog ("Duke", 5);
      Dog dog3 = new Dog ("Max", 8);

      // Changing the name of dog3.
      dog3.changeName ("Milo");

      // Adding a year to dog2's age.
      dog2.addYear();

      // Printing out each dog.
      System.out.println ("Dog 1:\n"
      + "Name: " + dog1.getName()
      + "\nAge in dog years: " + dog1.getAgeDogYears()
      + "\nAge in person years: " + dog1.getAgePersonYears()
      + "\n");

      System.out.println ("Dog 2:\n"
      + "Name: " + dog2.getName()
      + "\nAge in dog years: " + dog2.getAgeDogYears()
      + "\nAge in person years: " + dog2.getAgePersonYears()
      + "\n");

      System.out.println ("Dog 3:\n"
      + "Name: " + dog3.getName()
      + "\nAge in dog years: " + dog3.getAgeDogYears()
      + "\nAge in person years: " + dog3.getAgePersonYears()
      + "\n");

   }

} //end class
```

B)

ASQ1Output.txt:

```
Dog 1:
Name: Tom
Age in dog years: 3
Age in person years: 21

Dog 2:
Name: Duke
Age in dog years: 6
Age in person years: 42

Dog 3:
Name: Milo
Age in dog years: 8
Age in person years: 56
```

C)

FunZoneBadge.java:

```
/**
 This class represents a FunZoneBadge.
 @author Zohaib Khan 3740572
*/

public class FunZoneBadge {

   /**
    The name of the badge holder.
   */
   private String name;

   /**
    The badge number.
   */
   private int badgeNumber;

   /**
    The total amount of charges that
    have been added to the badge.
    */
    private double totalAmountOwed;

   /**
    Constructor class to initialize the
    instance variables.
    @param nameIn the name of the badge holder.
    @param badgeNumberIn the badge number.
   */
   public FunZoneBadge (String nameIn, int badgeNumberIn) {
      name = nameIn;
      badgeNumber = badgeNumberIn;
      totalAmountOwed = 0.00; // It will be zero upon entry.
   }

   /**
    This method retrieves the name of the badge holder.
    @return the name of the badge holder.
   */
   public String getName () {
      return name;
   }
   /**
    This method retrieves the badge number.
    @return the badge number.
   */
   public int getBadgeNumber () {
      return badgeNumber;
   }

   /**
    This method returns the total amount owed.
    @return the total amount owed.
   */
   public double getTotal () {
      return totalAmountOwed;
   }

   /**
    Adding the price of the centre's offering.
    @param priceIn the price of accessing an offering.
   */
   public void addAmount (double priceIn) {
      totalAmountOwed = totalAmountOwed + priceIn;
   }

   /**
    Calculating and returning the amount to be donated.
    @param percent the percentage to be donated.
    @return the donation amount.
   */
   public double getDonation (double percent) {
      return totalAmountOwed * percent ;
   }

} //end class
```

FunZoneDriver.java:

```
/**
 This is a driver program for the FunZoneBadge class.
 @author Zohaib Khan 3740572
*/

public class FunZoneDriver {

   public static void main (String[] args){

      // Creating a badge object for Ben.
      FunZoneBadge bensBadge = new FunZoneBadge ("Ben Landry",
      12341234);

      // Charging Ben for laser tag.
      bensBadge.addAmount(6.00);

      // Creating a badge object for Maria.
      FunZoneBadge mariasBadge = new FunZoneBadge ("Maria Lopez",
      2468135);

      // Charging Maria for trampoline park.
      mariasBadge.addAmount(7.25);

      // Creating a badge object for Karl.
      FunZoneBadge karlsBadge = new FunZoneBadge ("Karl Wagner",
      3451016);

      // Charging Karl for Mandalorian pinball.
      karlsBadge.addAmount(2.75);

      // Creating a badge object for Lori.
      FunZoneBadge lorisBadge = new FunZoneBadge ("Lori Evans",
      5798642);

      // Charging Lori for wall climbing.
      lorisBadge.addAmount(12.50);

      // Charging Lori for axe throwing.
      lorisBadge.addAmount(18.75);

      // Charging Maria for sundae bar.
      mariasBadge.addAmount(6.50);

      // Charging Ben and Karl for pool.
      bensBadge.addAmount(9.50);
      karlsBadge.addAmount(9.50);


      // Printing out each badge holder.
      System.out.println ("Badge holder 1:\n"
      + "Name: " + bensBadge.getName()
      + "\nBadge number: " + bensBadge.getBadgeNumber()
      + "\nTotal charges: " + bensBadge.getTotal()
      + "\nAmount donated: " + bensBadge.getDonation(0.18)
      + "\n");

      System.out.println ("Badge holder 2:\n"
      + "Name: " + mariasBadge.getName()
      + "\nBadge number: " + mariasBadge.getBadgeNumber()
      + "\nTotal charges: " + mariasBadge.getTotal()
      + "\nAmount donated: " + mariasBadge.getDonation(0.12)
      + "\n");

      System.out.println ("Badge holder 3:\n"
      + "Name: " + karlsBadge.getName()
      + "\nBadge number: " + karlsBadge.getBadgeNumber()
      + "\nTotal charges: " + karlsBadge.getTotal()
      + "\nAmount donated: " + karlsBadge.getDonation(0.20)
      + "\n");

      System.out.println ("Badge holder 4:\n"
      + "Name: " + lorisBadge.getName()
      + "\nBadge number: " + lorisBadge.getBadgeNumber()
      + "\nTotal charges: " + lorisBadge.getTotal()
      + "\nAmount donated: " + lorisBadge.getDonation(0.20)
      + "\n");

   }

} //end class
```

d)

AS2Q2Output.txt:

```
Badge holder 1:
Name: Ben Landry
Badge number: 12341234
Total charges: 15.5
Amount donated: 2.79

Badge holder 2:
Name: Maria Lopez
Badge number: 2468135
Total charges: 13.75
Amount donated: 1.65

Badge holder 3:
Name: Karl Wagner
Badge number: 3451016
Total charges: 12.25
Amount donated: 2.45

Badge holder 4:
Name: Lori Evans
Badge number: 5798642
Total charges: 31.25
Amount donated: 6.25
```