AP Chapter 13 Worksheet 6 ArrayList Name_____

 Pts: 10

class hour _____

I. Suppose data is an ArrayList<Double> of size > 0.
Using ArrayList methods increment by one the element with index 0 and store the new result back into position 0 by overriding the previous contents.

Complete the following that will do the above. (2 pts)

Answer: data**.**_____ (_____, data**.**_____(0) + 1);

II. Write the complete declaration statement for an ArrayList named fish whose component type is **Locatable** (using Java 5.0 notation). (Note: Do both the reference declaration and the object declaration in the same line).

Answer: _____

III. Given the code fragment below, what is the content of names and the output after doing the following print statements?

(3 pts)

```
ArrayList<String> names = new ArrayList<String>( );
names.add("A");
names.add(0, "B");
names.add("C");
names.remove(1);
```

Answer: System.out.print(names.get(0)); _____

System.out.print(names.get(1)); _____

System.out.print(names.get(2)); _____

Over

III. Questions #1 and #2 on pages 3 and 4 are based on the Coin and Purse classes given below:

```
public class Coin
{
    private double myValue;
    private String myName;

    // constructor
    public Coin(double value, String name)
    {
        myValue = value;
        myName = name;
    }

    // Return the value of the coin
    public double getValue( )
    {
        return myValue;
    }

    // Return the name
 of the coin
    public String getName( )
    {
        return myName;
    }

    // Define equals method for Coin objects
    public boolean equals(Object obj)
    {
        // implementation not shown
    }
    // Other methods not shown
}

// A purse holds a collection of coins
public class Purse
{
    private ArrayList coins;

    // constructor - creates an empty purse
    public Purse( )
    {
        coins = new ArrayList( );
    }

    // Adds aCoin to the purse
    public void add(Coin aCoin)
    {
        coins.add(aCoin);
    }

    //Returns total value of coins in purse
    public double getTotal( )
    {
        // implementation code not shown – see next page for the implementation
```

}

continued

1) Here is the getTotal method from the Purse class: (2 pts)

```
// returns total value of coins in purse
public double getTotal( )
{
    double total = 0;
    < more code >
    return total;
}
```

Which of the following is a correct replacement for *< more code >*? 1) Answer: _____

A)
```
for (int i = 0; i < coins.length; i++)
{
    Coin c = (Coin) coins.get(i);
    total += c.getValue( );
}
```

B)
```
for (int i = 0; i <coins.length; i++)
{
    Coin c = (Coin) coins.[i];
    total += c.getValue( );
}
```

C)
```
for (int i = 0; i < coins.size( ); i++)
{
    Coin c = coins.get(i) ;
    total += c.getValue( );
}
```

D)
```
for (int i = 0; i < coins.size( ); i++)
{
    Coin c = coins.get(i);
    total += coins * getValue( );
}
```

E)
```
for (int i = 0; i < coins.size( ); i++)
{
    Coin c = (Coin) coins.get(i);
    total += c.getValue( );
}
```

2) A boolean method find( ) is added to the Purse class: (2 pts)

```
// Returns true if the purse has a coin that matches aCoin, false otherwise
public boolean find(Coin aCoin)
{
   for (int i = 0; i < coins.size( ); i++)
   {
     < code to find match >
   }
   return false;
}
```

Which is a correct replacement for < *code to find match* >? 2) Answer: _____

I.
```
Coin c = (Coin) coins.get(i);
if (c.equals(aCoin))
  return true;
```

II.
```
Coin c = (Coin) coins.get(i);
if (c.getName( ).equals(aCoin.getName( )))
  return true;
```

III.
```
Coin c =  (Coin) coins.get(i);
if (c.getValue( ).equals(aCoin.getValue( )))
  return true;
```

A) I only
B) II only
C) III only
D) I and II only
E) I, II, and III

KEY
size = 3
first account number = 1008
last account number = 1729

names contains the strings "B" and "C" at positions 0 and 1 and null at 2

Key
1) E
2) D