

## TUTORIAL 3 SOLUTION

1. Describe the main difference between the way information is stored in:

- an array of primitive data values, for example integers
- an array of objects

primitive: array stores **values**

objects: array stores **references** to objects

2. Describe the contents of the array `players` after the following lines of code have been executed:

```
Player[] players;  
players = new Player[4];
```

array contains 4 **null references**, because array has been created but no `Player` objects have been created

3. What is the effect of the keyword `static` in a variable declaration?

variable is a **class variable**

value is shared between all instances of that class

changing value through one instance changes value for all instances

can be used without creating an object, e.g. `Math.PI`

4. What keyword indicates that a value cannot be changed once it is initialised?

`final`

5. What coding pattern is implemented by declaring a field in one class which is an array of objects whose type is the name of another class?

Has-as (array)

6. Explain the difference between a class diagram and an object diagram.

Class diagram shows templates which can be used to create objects

Each class has to be coded as a Java class

Each class appears **once** in class diagram

Object diagram shows "snapshot" of objects which have been created from the classes

Represents state at a particular time when the program is running

There can be **more than one** object of the same class in the object diagram

7. What kind of loop would you use if:

- you don't know how many times the loop will be executed

AND

- you want to ensure that the code inside the loop is executed at least once

**do-while**

condition is checked at end of loop so code inside loop will be executed at least once even if the condition is false initially

compare to while loop, where condition is checked at beginning of loop - if condition is false initially code inside loop will never be executed

8. What is name of the method which is the entry point for a standalone Java application?

**main**

Signature: `public static void main(String[] args)`

9. Look at the following code:

```
int[] myArray = {1,1,0,0,0};

for(int i=2;i<5;i++)
{
    myArray [i] = myArray [i-2] + myArray [i-1];
}
```

What values does the array `myArray` contain after this code has executed?

**{1,1,2,3,5}**

Loop variable `i` starts at 2, so array elements from `myArray[2]` onward (i.e. third element) are changed

Each element becomes the sum of the two preceding elements, e.g. `myArray[2] = myArray[0] + myArray[1]`

10. Explain the difference between unit testing and system testing.

Unit testing tests individual parts of a system, typically classes

System testing tests functionality of system as a whole

11. The following method calculates an employee's pay.

```
/**
 * calculates the pay for an hourly paid employee
 *
 * @param emp the employee
 * @param hours the number of hours worked
 * @param rate the hourly pay rate
 * @return the calculated pay for the employee
 */
public double calculatePay(Employee emp,int hours,double rate)
{
    // detail omitted
}
```

Write a Javadoc comment which would correctly document this method.

12. Look at the following Java classes. For brevity, all fields are public and getters/setters are not included.

```
public class Point
{
    public int x, y;

    public Point(int x, int y)
    {
        this.x = x;
        this.y = y;
    }
}

public class Circle
{
    public Point centre;
    public Integer radius;

    public Circle(int x, int y, int radius)
    {
        this.centre = new Point(x,y);
        this.radius = new Integer(radius);
    }
}
```

```
public class TestClass
{
    public static void main(String[] args)
    {
        Circle circ = new Circle(4,5,3);
    }
}
```

How many objects are created when the `main` method runs?

Three objects

A Circle, a Point and an Integer

Main method causes constructor of Circle to be executed (**new** Circle)

Constructor of Circle in turn creates two objects (**new** Point and **new** Integer)

Integer is a class which encapsulates a primitive integer value and has a set of methods which are useful for working with integers