

Tutorial 4: Sequence

This tutorial will provide practice at implementing and testing sequences.

Task 1: VAT



Write a program to prompt the user to enter the cost of an item before VAT has been added. The program should calculate and output the amount of VAT payable and the total cost of the item including VAT.

VAT is currently charged at 20% and this should be stored as a constant in your program.

Step 1: Create a NetBeans project

- using NetBeans create a new project called `VATproj` and store it in your T4 folder

Step 2: Write source code

- add a new file called `VAT` to the `VATproj` project
- write the `VAT` code, to calculate and output the VAT payable and the cost of an item after VAT has been added

Step 3: Design a test plan

- using the information given in the lecture (slides 12 – 14), design a test plan that fully tests your program and store it in a file called `VATTest.doc`

Step 4: Test your program and take screen shots

- run your program with each test case you designed in your test plan
- take a screen shot of each test run and save it as a separate file in your project folder called `VAT1.jpg`, `VAT2.jpg`, etc.

Portfolio requirements

- The NetBeans project for this completed task
- `VATTest.doc` from step 3 containing your test plan
- `VAT1.jpg`, `VAT2.jpg`, etc from step 4, containing screen shots of each test run

Task 2: Average



The following code has been written to input 3 integer values and calculate and output the average of those values as a double. However, the programmer is not sure whether or not the code is correct and so has asked you to help test it.

By following the steps below, you will design a test plan to fully test the program, then use it to check the code and correct any errors that you find.

```
//program to output the average of 3 integer values
public class Average
{
    public static void main (String args[])
    {
        Scanner kybd = new Scanner(System.in);

        System.out.print("Enter first integer: ");
        int num1 = kybd.nextInt();
        System.out.print("Enter second integer: ");
        int num2 = kybd.nextInt();
        System.out.print("Enter third integer: ");
        int num3 = kybd.nextInt();

        double ave = num1 + num2 + num3 / 3;

        System.out.println("Average is: " + ave);
    }
}
```

Step 1: Create a NetBeans project

- create a new project called `AverageProj` and store it in your T4 folder

Step 2: Copy source code

- add a new file called `Average` to the `AverageProj` project
- copy the above code from Blackboard and paste it into the `Average` file

Step 3: Design a test plan

- using the information given in the lecture (slides 12 – 14), design a test plan that fully tests the average program and store it in a file called `AverageTest.doc`

Step 4: Test your program

- run the program with each test case you designed in your test plan
- using the results from your testing correct any program errors you find

Step 5: Take a screen shot of the output

- take a screenshot of the output from the corrected `Average` program using 3, 5 and 6 as input
- save the screenshot in your project folder as `Average.jpg`

Portfolio requirements

- The NetBeans project for this completed task
- `AverageTest.doc` from step 4 containing your test plan
- `Average.jpg` from step 5, containing a screen shot of the output using 3, 5, 6 as input

Task 3: Distance

The formula to calculate the distance travelled by an object moving with constant acceleration is:

$$s = ut + \frac{1}{2} at^2$$

where s = distance
 u = initial velocity
 t = time taken
 a = acceleration

Write a program to prompt the user for the required input, and output the distance travelled by an object. Develop a full test plan to test your program.

Step 1: Create a NetBeans project

- create a new project called `DistanceProj` and store it in your T4 folder

Step 2: Write source code

- write the `Distance` code, to output distance travelled using the above formula

Step 3: Design a test plan

- design a test plan to test the distance program and store it in `DistanceTest.doc`

Step 4: Test your program and take screen shots

- run your program with each test case you designed in your test plan
- take a screen shot of each test run and save it as a separate file in your project folder called `Distance1.jpg`, `Distance2.jpg`, etc

Portfolio requirements

- The NetBeans project for this completed task
- `DistanceTest.doc` from step 3 containing your test plan
- `Distance1.jpg`, `Distance2.jpg`, etc from step 4, containing screen shots

Task 4: Time to drop

To complete this task, you will need to use some methods from the `Math` class. You could consult the on-line documentation at:

<http://docs.oracle.com/javase/7/docs/api/java/lang/Math.html>

The formula from Task 3 can be used to calculate the time taken for an object that is initially stationary to drop vertically under the influence of gravity from a given height:

$$t = \sqrt{2s/a}$$

where `s` = distance in metres

`t` = time taken in seconds

`a` = acceleration (constant: 9.8 m/s^2)

Write a program to prompt the user for the required input, and output the time taken (to two decimal places) for the object to drop. Develop a full test plan to test your program.

Portfolio requirements

- The NetBeans project for this completed task
- `TimeTaken.doc` containing your test plan
- Image files containing screen shots of the actual results when following your test plan