



NORD ANGLIA EDUCATION



Java for Beginners

British School
of Beijing

Java Task Book 1

Name



- ! Complete all activities in this workbook by including screenshots of your code and working programs where requested.
- ! To use this workbook, you will need to use the PPT/PDF book 'Java for Beginners'.
- ! Save each program created and paste your code/outputs into this booklet
- ! Remember, when you get stuck, C3B4ME and use the support links provided.

This booklet provides a comprehensive set of problems for you to solve using the Java programming language. It is separated into levels, each of which focus on a different aspect of generic programming constructs. They will gradually get more complex towards the end and will likely require some independent research to complete. Some recommended websites to use are below:

- Useful for syntax help: <https://www.w3schools.com/java/>
- Tutorials and examples: <https://www.programiz.com/java-programming>
- Nice tutorials covering the full Java spec: <https://www.javatpoint.com/java-tutorial>
- Nice tutorials covering the full Java spec: <https://www.tutorialspoint.com/java/index.htm>
- Beginners Tutorial: <https://www.guru99.com/java-tutorial.html>
- Glossary: <https://stackify.com/java-glossary/>

The levels in these workbooks are:

Level 1 - Variables

Level 2 - Input/Output

Level 3 - Selection

Level 4 - Iteration

Level 5 - Algorithmic Thinking

Level 6 - Arrays

Level 7 – File Reading/Writing

Level 8 – Methods

Level 9 – Objects

Level 10 – GUI Elements

Contents

GLOSSARY.....	5
Level 1: Variables (easy).....	7
Task 1C – Postman Pat	9
Task 1D – All about me	11
Task 1F Calculations	13
Level 2: Input & Output (easy).....	15
Task 2A Greet Me	15
Task 2D Number Time.....	17
Task 2I Times Tables	19
Task 2 Shout and Whisper.....	21
Task 2L Aplha Omega	23
Level 3: Selection (easy).....	25
Task 3A Temp Check	25
Task 3B Super Hero.....	27
Task 3D Speeding Fine	29
Task 3E GTA Checker	31
Task 3F Find my teacher.....	33
Task 3I Coding system selector	35
Task 3J Brickman.....	37
Task 3K Pay Calc	39
Task 3L Simpsons Checker	41
Task 3M Santa Claus.....	43
Task 3N Odd or Even.....	45
Task 3Q Cinema Age Checker.....	47
Task 3T Number Selector	49
Task 3V Quiz Master.....	51
Task 3W Case Calc.....	54
Level 4: Iteration (easy-medium)	57
Task 4 – Repeat after me	57
Task 4B Spellman	59
Task 4C Alphabet Loop.....	61



Task 4D Doubling Score	64
Task 4E Green Bottles	66
Task 4F Factors	68
Task 4G Loops with numbers	70
Task 4H Block Maker	72
Task 4I Word Pyramid.....	74
Task 4J Plus Me.....	76
Task 4K Word Size Checker	78
Task 4L While Calc.....	80
Task 4M Block Maker 2	83
Level 5: Algorithmic Thinking (medium).....	85
Task 5B Biggest Number	85
Task 5C Maths Trainer	87
Task 5E Lock Picker	89
Task 5F Season Checker	91
Task 5M Name Mixer.....	93
Task 5R InCommon.....	95
Level 6: Arrays.....	97
Task 6A Little Boxes.....	97
Task 6A Little Boxes.....	99
Task 6C Number Hunt.....	101
Task 6D Backward Boxes	103
Task 6F Array Averages.....	105
Task 6G Array Min	107
Task 6A Little Boxes.....	109
Task 6N Letter Sorter	111
Task 6O Super Sorter.....	113
.....	113



GLOSSARY

Add in any key terminology and the definitions as you work through the booklet.

TERM	DESCRIPTION
Generic Programming Constructs	
Algorithm	
Decomposition	
Pattern Recognition	
Abstraction	
Variable	
String	
Integer	
Double	
Function	
Built-in function	
Truncate	
Modulus	
Parameter	
Selection	
Condition	
Flow Chart	
Fixed Loop	
Conditional Loop	
Array	
1D List	
2D List	
Elements	
Append	
Java Principles	
Pop()	
Variable	
Method	
Type + Name	
Identifier	
Signature	
Block of Code	
Class	
Instance	
new Keyword	
Constructor	



new + Constructor	
. (Dot operator)	
Object-Oriented Paradigm	
Object	
Field	
Side Effect	
Reference	
Primitive Type	
State	
Expression	
Statement	
Declaration	
Compiler	



Level 1: Variables (easy)

1B – James Bond

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project called 1B-James Bond. ()
2. Create a new class () called JamesBond within this project. You will be editing the main method inside of this class to create your program.

Task: Create a simple program that has two variables; one called **destination** and one called **days**. You can populate these variables with suitable values. Your program should then output these variables in the following sentence (the red text indicates where the variables should be used).

James Bond is on a top secret mission to DESTINATION.

All we know is that Bond only has DAYS days to save the world!!!

Example Output #1:

```
James Bond is on a top secret mission to California.  
All we know is that Bond only has 15 days to save the world!!!
```

Example Output #2:

```
James Bond is on a top secret mission to Argentina.  
All we know is that Bond only has 3 days to save the world!!!
```

Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

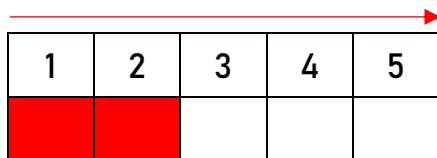
What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour per the example below)

EASY

VERY DIFFICULT





Task 1C – Postman Pat

1C – Postman Pat

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project called 1C-PostManPat. ( Java Project)
2. Create a new class () called PostManPat within this project. You will be editing the main method inside of this class to create your program.

Task: Create a simple program that has two variables; one called **postcode** and one called **towncode**. Enter a postcode as a value into the postcode variable. The value for towncode should be the first letter of your home town. The variables should then be output in the sentence below (the red text should be replaced with the appropriate variable).

My postcode is **POSTCODE** and my town code is **TOWNCODE**.

Example Output #1:

My postcode is Dai 25F and my town code is D.

Example Output #2:

My postcode is MER2 6YH and my town code is M.



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

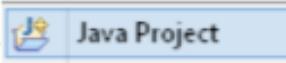
A horizontal scale consisting of five empty square boxes numbered 1 through 5. A red arrow points from the left towards the right, indicating the progression of difficulty.

1	2	3	4	5



Task 1D – All about me

1D – All About Me**Java Programming Tasks (V2)****Preliminary Setup:**

1. Using Eclipse, create a new Java Project called 1D - AboutMe. ()
2. Create a new class () called AboutMe within this project. You will be editing the main method inside of this class to create your program.

Task: Create a simple program that will output the following details about you (or someone else) on the screen:

1. Name (Both forename and surname)
2. Gender (either M or F)
3. Age
4. Height (In meters)
5. Married (true or false)

For each, you will need to create an appropriately named variable with a suitable data type. These variables should then have values added to them so they can be outputted as in the sentences shown in the example below.

Example Output:

```
My name is: Bilbo Baggins
My gender is: M
My age is: 88
My height is: 1.2
Am I married? false
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5

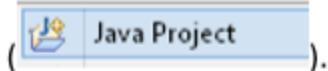
Task 1F Calculations

1F – Calculations

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project called 1F - Calculations.



2. Create a new class () called Calculations within this project. You will be editing the main method inside of this class to create your program.

Task: Start by creating 3 variables containing random whole numbers. One further variable must contain a decimal number. Your program should then run the following calculations, saving the result of each as a new variable.

1. The program will firstly **ADD** all four numbers together.
2. The program will then take this number and **DIVIDE** by 3.
3. The program will then take this number and **MULTIPLY** by 2.
4. The program will then take this number and **SUBTRACT** 9.

NOTE: You should perform all of the calculations and save each result in a new variable. After you have done all of them, proceed to creating the output with system.out.println to resemble the following example:

Example Output #1:

```
Starting numbers = 50,44,101,58.58
Calculation 1 = 253.57999999999998
Calculation 2 = 84.52666666666666
Calculation 3 = 169.05333333333333
Calculation 4 = 160.05333333333333
```

Example Output #2:

```
Starting numbers = 22,37,88,21.058
Calculation 1 = 168.058
Calculation 2 = 56.01933333333333
Calculation 3 = 112.03866666666666
Calculation 4 = 103.03866666666666
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

A horizontal scale consisting of five empty square boxes numbered 1 through 5. An arrow points from the left side of the scale towards the number 5, indicating a progression from easy to very difficult.

1	2	3	4	5

Level 2: Input & Output (easy)

Task 2A Greet Me

2A – Greet Me

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project called 2A-GreetMe. (
2. Create a new class () called GreetMe within this project. You will be editing the main method inside of this class to create your program.

Task: Create a simple program has the user enter their name via the keyboard. The program should then greet the person by their name.

Example Output #1:

```
What is your name?  
Bob  
Hello Bob! How are you?
```

Example Output #2:

```
What is your name?  
Sarah  
Hello Sarah! How are you?
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

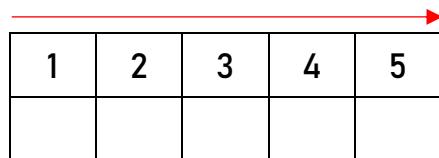
What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT





Task 2D Number Time

2D – Number Time

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project called 2D-NumberTime. ( Java Project)
2. Create a new class ( G) called NumberTime within this project. You will be editing the main method inside of this class to create your program.

Task: Have the user enter a number and then perform the following calculations on it (which it has to display as in the examples):

- Double the number (number \times 2)
- Half the number (number \div 2)
- The number squared (number \times number)
- The number plus half of itself (number + (number \div 2))

Hint: Think carefully about how you need to store the number... as you will be handling decimal places in this exercise.

Example Output #1:

```
Please enter a number:  
8  
Double of 8.0 is 16.0  
Half of 8.0 is 4.0  
Square of 8.0 is 64.0  
One and half times 8.0 is 12.0
```

Example Output #2:

```
Please enter a number:  
6  
Double of 6.0 is 12.0  
Half of 6.0 is 3.0  
Square of 6.0 is 36.0  
One and half times 6.0 is 9.0
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

A horizontal scale consisting of five empty square boxes numbered 1 through 5. A red arrow points from the left towards the right, indicating the direction of increasing difficulty.

1	2	3	4	5

Task 2I Times Tables

2I – Times Tables**Java Programming Tasks (V2)****Preliminary Setup:**

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Create a program that will ask the user to enter a number. The program should then output the times table for that number from 1-12. It should be outputted in the **SAME WAY** as the output examples show below:

Example Output #1:

Please enter a number:

```
2
1 x 2 is 2
2 x 2 is 4
3 x 2 is 6
4 x 2 is 8
5 x 2 is 10
6 x 2 is 12
7 x 2 is 14
8 x 2 is 16
9 x 2 is 18
10 x 2 is 20
11 x 2 is 22
12 x 2 is 24
```

Example Output #2:

Please enter a number:

```
8
1 x 8 is 8
2 x 8 is 16
3 x 8 is 24
4 x 8 is 32
5 x 8 is 40
6 x 8 is 48
7 x 8 is 56
8 x 8 is 64
9 x 8 is 72
10 x 8 is 80
11 x 8 is 88
12 x 8 is 96
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenshot/screenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

A horizontal scale consisting of five empty square boxes labeled 1, 2, 3, 4, and 5 from left to right. A red arrow points from the word 'EASY' towards the number 5, indicating a spectrum of difficulty.

Task 2 Shout and Whisper

2J – Shout & Whisper**Java Programming Tasks (V2)****Preliminary Setup:**

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Create a program that will ask the user to enter a word. The program should then output the word in the following two sentences after using appropriate methods to format the word accordingly:

Example Output #1:

```
Please enter your word:  
qUack  
I SHOUT QUACK WHEN I AM EXCITED!!!  
i whisper quack when i am trying to be quiet.
```

Example Output #2:

```
Please enter your word:  
Chicken  
I SHOUT CHICKEN WHEN I AM EXCITED!!!  
i whisper chicken when i am trying to be quiet.
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

A horizontal scale consisting of five empty square boxes numbered 1 through 5. A red arrow points from the left towards the right, indicating the progression of difficulty.

1	2	3	4	5



Task 2L Aplha Omega

2L – Alpha Omega**Java Programming Tasks (V2)****Preliminary Setup:**

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Create a program that will ask the user to enter a word. The program should then output the number of letters that are in the entered word, along with the first and last letters of it.

Hint: *The word can be any length, so you will need to work out how long the word is in order to retrieve the last letter.*

Example Output #1:

```
Hi there! Please enter a word:  
quack  
There are total of 5 letters in quack  
The first letter of quack is q  
The last letter of quack is k
```

Example Output #2:

```
Hi there! Please enter a word:  
monologue  
There are total of 9 letters in monologue  
The first letter of monologue is m  
The last letter of monologue is e
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

A horizontal scale consisting of five empty square boxes numbered 1 through 5. A red arrow points from the left towards the right, indicating the direction of increasing difficulty.

1	2	3	4	5

Level 3: Selection (easy)

Task 3A Temp Check

3A - TempCheck

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Create a simple program that asks the user to enter the current temperature. If the temperature is above 0, nothing happens (the program just ends). If the temperature is 0 or below, it should output “THAT'S FREEZING!”.

Example Output #1:

```
Please enter the temperature >  
0  
THAT'S FREEZING!
```

Example Output #2:

```
Please enter the temperature >  
15
```

Example Output #3:

```
Please enter the temperature >  
-9  
THAT'S FREEZING!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

A horizontal scale consisting of five empty square boxes numbered 1 through 5. A red arrow points from the left towards the right, indicating the progression of difficulty.

1	2	3	4	5

Task 3B Super Hero

3B – Superhero Check**Java Programming Tasks (V2)****Preliminary Setup:**

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Create a simple program that asks the user to enter their age for evaluation. If the age is 10 or less, the computer will respond by saying “You can be a superhero! :-)”. If the age is greater than 10, the computer must state “You are TOO OLD to be a superhero! :-(“.

Example Output #1:

```
Please enter your age >  
10  
You can be a superhero! :-)
```

Example Output #2:

```
Please enter your age >  
15  
You are TOO OLD to be a superhero! :-(
```

Example Output #3:

```
Please enter your age >  
7  
You can be a superhero! :-)
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

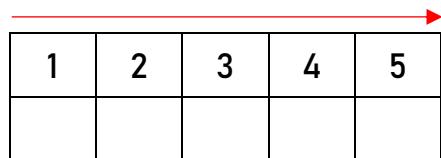
What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT





Task 3D Speeding Fine

3D – Speeding Fine

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: You need to create a program that Dartford police can use to calculate how much to fine speeding motorists. The fines are broken down as follows:

- If the motorist is travelling at less than 40mph, they are not speeding and so simply output a suitable message as such.
- If the speed is between 40-44mph (inclusive), the offender must pay a fixed penalty fine of £90.
- If the speed is over 44mph then the offender will be fined £100 for every mph faster than 44mph that they are travelling.

Example Output #1:

```
Please enter the motorist speed >  
50  
Motorist is travelling 6mph over 44mph.  
Motorist is fined £600
```

Example Output #2:

```
Please enter the motorist speed >  
25  
Motorist is not speeding.
```

Example Output #3:

```
Please enter the motorist speed >  
44  
Motorist is fined £90
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

A horizontal scale consisting of five empty square boxes numbered 1 through 5. A red arrow points from the left towards the right, indicating the progression of difficulty.

1	2	3	4	5



Task 3E GTA Checker

3E – GTA Checker

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Your task is to create a program that will output a different message based upon the age that the user enters. The messages are:

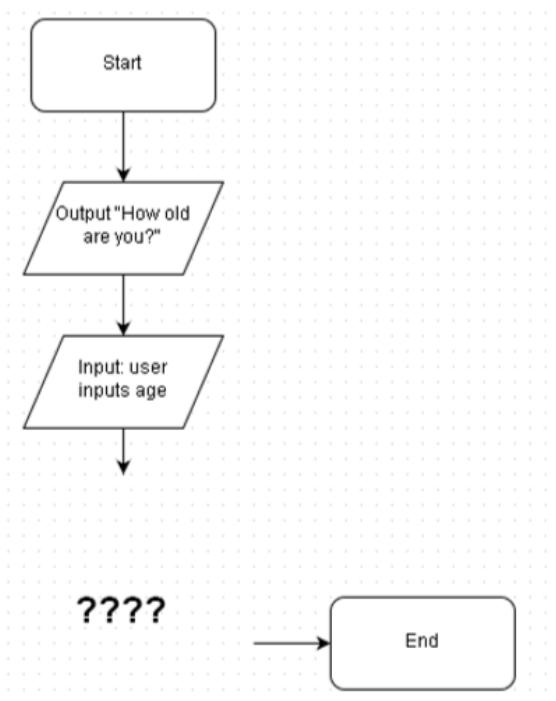
1. If age is greater than or equal to 18, display: *You are old enough to play GTA*
2. If age is greater than or equal to 16, display: *You probably could play GTA if your parents agree*
3. When age does not fall into these two conditions, display: *You should definitely not be playing GTA*

You must then also **COMPLETE** the partial flowchart to the right.

Use: <https://www.draw.io/> or **Flowgorithm** to create the flowchart.

Submission: Submit your COMPLETED FLOWCHART along with the code for this assignment. BOTH files must be turned in. Turn the flowchart in as an image file. Ensure you have attached your completed file.

Submission: Submit your solution (copy the entire contents of the class) in an image format before the deadline. Ensure you have attached your completed file.



NOTE: For this solution, a flowchart must also be completed within flowgorithm or your own choice of tool



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscren/screenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Flowchart

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 3F Find my teacher

3F – Find My Teacher

Java Programming Tasks (V2)

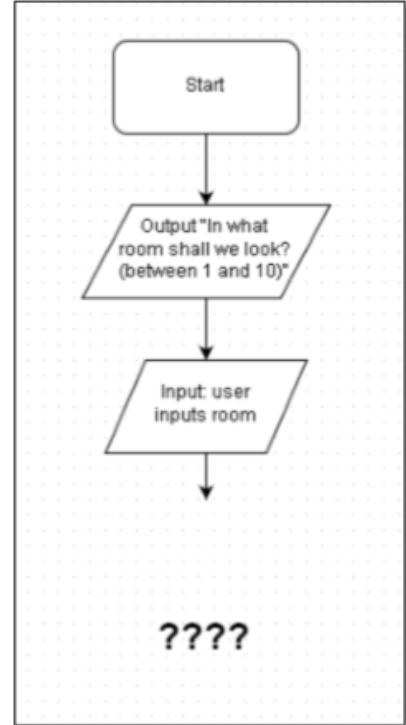
Preliminary Setup:

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Your task is to create a program that will allow the user to guess what room their Computing teacher is in. The user should be asked to enter a number between 1 and 10...

1. If the user enters 1, it should display: *Mr Coetzee is in room 1.*
2. If the user enters 6, it should display: *Mr Taylor is in room 6.*
3. If the user enters 8, it should display: *Mr Lowe is in room 8.*
4. If the user enters any other number (including one greater than 10) the program should simply display: *There is nobody in that room.*

You must then also **COMPLETE** the partial flowchart to the right.



NOTE: For this solution, a flowchart must also be completed within flowgorithm or your own choice of tool



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Flowchart

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5

Task 3I Coding system selector

3I – Coding System Selector**Java Programming Tasks (V2)****Preliminary Setup:**

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: You must make a program that tells the user what character coding system to use, based on how many letters they need to store.

- If they need less than 129, they should use Basic ASCII.
- If they need less than 257, they should use Extended ASCII.
- If they need 257 or more, they should use Unicode.

Example Output #1:

```
How many characters do you need to store?
```

```
73
```

```
I would recommend you use Basic ASCII.
```

Example Output #2:

```
How many characters do you need to store?
```

```
200
```

```
I would recommend you use Extended ASCII.
```

Example Output #3:

```
How many characters do you need to store?
```

```
200000
```

```
I would recommend you use Unicode.
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5

Task 3J Brickman

3J – Brickman**Java Programming Tasks (V2)****Preliminary Setup:**

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: We want to make a row of bricks that is 22 inches long. We have a number of small bricks (1 inch each) and big bricks (5 inches each). Determine if it is possible to make the goal (22 inches) using only the given block sizes (see example output for exact wording)

Little brick:**1 inch****Large Brick:****5 inches****Example Output #1:**

```
Enter the number of little bricks > 2
Enter the number of big bricks > 4
That combination WILL fit perfectly into 22 inches
```

Example Output #2:

```
Enter the number of little bricks > 3
Enter the number of big bricks > 7
OOPS! That combination will NOT fit into 22 inches
```

Submission: Submit your solution (copy the entire contents of the class) in .doc, .docx or .rtf format to Edmodo before the deadline. **Ensure you have attached your completed file.**

**** Extension Challenge – Earn a bonus mark!**

Change the program so that the user can choose how long the goal should be (in other words, not 22 inches, but what the user decides).

Example Extension Output #1:

```
Enter the goal size (in inches) > 25
Enter the number of little bricks > 2
Enter the number of big bricks > 4
OOPS! That combination will NOT fit into 25 inches
```

```
Enter the goal size (in inches) > 16
Enter the number of little bricks > 1
Enter the number of big bricks > 3
That combination WILL fit perfectly into 16 inches
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

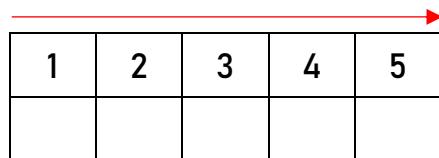
What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT



Task 3K Pay Calc

3K - PayCalc

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: You must create a program that will calculate how much to pay a worker. The program should ask for the number of hours worked (a whole number) and then calculate the pay according to the following rules:

- The first 8 hours are paid at a rate of £8.25 per hour.
- Every hour subsequent to the first 8 hours are paid at an overtime rate of £14.50 per hour.

The program must then output this information as follows:

Example Output #1:

```
Enter the number of hours worked > 8
Standard Pay: £66.0
Overtime Pay: £0.0
Total Pay: £66.0
```

Example Output #2:

```
Enter the number of hours worked > 15
Standard Pay: £66.0
Overtime Pay: £101.5
Total Pay: £167.5
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

A horizontal scale consisting of five empty square boxes numbered 1 through 5. A red arrow points from the left towards the number 5, indicating a progression from easy to very difficult.

1	2	3	4	5



Task 3L Simpsons Checker

3L – Simpsons Checker

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: You must create a program that will ask the user to enter an age and gender. The program should then match this against the characters according to the table:

Age	Gender	Character
34	f	Marge Simpson
39	m	Homer Simpson
8	f	Lisa Simpson
10	m	Bart Simpson
1	f	Maggie Simpson

The program must then output if there is a match or not.

Hint: You will need to use an AND (&&) operator in your statements to join together the conditions.

Hint 2: You will likely need two scanners, one for numbers and one for chars, in order to prevent the error occurring where it skips over the second input.

Example Output #1:

```
Enter the age >
39
Enter the gender (m/f) >
m
This character is Homer Simpson!
```

Example Output #2:

```
Enter the age >
8
Enter the gender (m/f) >
f
This character is Lisa Simpson!
```

Example Output #3:

```
Enter the age >
100
Enter the gender (m/f) >
f
This does not match any characters!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

A horizontal scale consisting of five empty square boxes numbered 1 through 5. A red arrow points from the left towards the right, indicating a progression from 'EASY' to 'VERY DIFFICULT'.

1	2	3	4	5

Task 3M Santa Claus

3M – Santa Claus

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: You must create a program that will determine if Santa is going to leave any presents for the user. Santa's gift policy is as follows:

- If the user has been **GOOD** and **NICE** they will get presents.
- If the user has **NOT** been **GOOD** and **NAUGHTY** they will get coal.
- If the user has been any other combination, they will have to wait and see.

The program must then output if there is a match or not.

Hint: You will need to use an AND (&&) operator in your statements to join together the conditions.

Example Output #1:

```
Have you been good this year? (yes/no) >
yes
Have you been naughty or nice? (naughty/nice) >
nice
You will get presents this year!
```

Example Output #2:

```
Have you been good this year? (yes/no) >
no
Have you been naughty or nice? (naughty/nice) >
naughty
You will get a lump of coal!
```

Example Output #3:

```
Have you been good this year? (yes/no) >
yes
Have you been naughty or nice? (naughty/nice) >
naughty
You will have to wait and see!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenshot/screenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

A horizontal scale consisting of five empty square boxes labeled 1, 2, 3, 4, and 5 from left to right. A red arrow points from the word 'EASY' towards the number 5, indicating a spectrum of difficulty.

Task 3N Odd or Even

3N – Odd or Even**Java Programming Tasks (V2)****Preliminary Setup:**

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Create a program that will ask the user to enter a number between 1 and 100. The program should then determine if that number is ODD or EVEN.

The program must tell the user there is a problem if the number is not between 1 and 100 (inclusive) – it must NOT still run the calculation and output ODD or EVEN.

***Hint:** You will need to use modulus (%) to solve this.*

Example Output #1:

```
Please enter a number between 1 and 100 > 52
Number is EVEN!
```

Example Output #2:

```
Please enter a number between 1 and 100 > 33
Number is ODD!
```

Example Output #3:

```
Please enter a number between 1 and 100 > 105
Number is not between 1 and 100!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

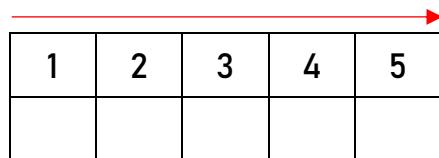
What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT





Task 3Q Cinema Age Checker

3Q – Cinema Age Checker

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Create a program that will ask the user to enter their age. The program should then determine, based upon the following rules, what is the oldest film rating the user may see:

- If they are less than 12 years old:
 - *You may see films rated U*
- If they are 12 years or more but less than 15:
 - *You may see films rated 12*
- If they are 15 years or more but less than 18:
 - *You may see films rated 15*
- If they do not fall into the above categories (so over 18):
 - *You may see any film you like!*

Example Output #1:

```
Please enter your age > 5
You may see films rated U
```

Example Output #2:

```
Please enter your age > 13
You may see films rated 12
```

Example Output #3:

```
Please enter your age > 17
You may see films rated 15
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 3T Number Selector

3T – Number Selector**Java Programming Tasks (V2)****Preliminary Setup:**

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Create a program that will ask the user to enter either 1, 2 or 3. The program will then output that number spelt in capital letters.

If the user enters anything other than 1, 2 or 3 then the program should say the value is invalid.

Note: you *MUST* use switch cases to solve this assignment.

Example Output #1:

```
Please enter either 1, 2 or 3 > 2
You entered TWO!
```

Example Output #2:

```
Please enter either 1, 2 or 3 > 1
You entered ONE!
```

Example Output #3:

```
Please enter either 1, 2 or 3 > 5
You did not enter a valid number!
```

You must also draw a complete flowchart for this program and submit along with the code.



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscren/screenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Flowchart

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 3V Quiz Master

3V – Quiz Master

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Create a program that will ask the user a quiz question. The player should enter the letter which corresponds to their answer. Note that the user's entered letter should be accepted regardless if it is in capitals or lower case ('A' is the same as 'a' etc).

Note: you *MUST* use switch cases to solve this assignment.

Example Output #1:

```
What is the third planet from the sun?  
a. Mercury  
b. Venus  
c. Earth  
d. Neptune  
What is your answer? > C  
Correct! Well done! :)
```

Example Output #2:

```
What is the third planet from the sun?  
a. Mercury  
b. Venus  
c. Earth  
d. Neptune  
What is your answer? > d  
Wrong answer!!
```

**** Turn to next page for extension challenge >>**



3V – Quiz Master

Java Programming Tasks (V2)

** Extension Challenge – Earn a bonus mark!

Modify your program so that when the game starts one of three random questions is presented to the user for them to answer in the same way as above. The question must be randomly selected.

Note: you **MUST STILL** only use switch cases to solve this assignment.

Hint: You will need to use Math Random for this:

```
int random = (int) (Math.random() *max+min);
```

Note in the above example *max* is the largest number you want to possibly appear and *min* the smallest number you want to possibly appear. The numbers should be substituted into the statement.

Example Extension Output #1:

In what year was DGS founded?

- a. 1467
- b. 1576
- c. 1675
- d. 1765

What is your answer? > **c**

Wrong answer!!

Example Extension Output #2:

What is the third planet from the sun?

- a. Mercury
- b. Venus
- c. Earth
- d. Neptune

What is your answer? > **C**

Correct! Well done! :)

Example Extension Output #3:

Which country has no border with France?

- a. Germany
- b. Spain
- c. Andorra
- d. Russia

What is your answer? > **a**

Wrong answer!!



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5

Task 3W Case Calc

3W – CaseCalc

Java Programming Tasks (V2)

Preliminary Setup:

1. Using Eclipse, create a new Java Project with a suitable name and inside of this create a suitably named class.

Task: Create a program that will ask the user to specify the type of operation they wish to perform on two given numbers. They can choose to add, subtract or multiply by entering in the applicable code. **The output should be printed on the screen EXACTLY as shown in the examples.**

If the user enters an invalid operator code then the program should respond by saying it does not understand.

Note: you *MUST* use switch cases to solve this assignment.

Example Output #1:

```
Please select operation (ADD, SUB, MLP) > ADD
Please enter the first number > 5
Please enter the second number > 6
5 + 6 = 11
```

Example Output #2:

```
Please select operation (ADD, SUB, MLP) > SUB
Please enter the first number > 88
Please enter the second number > 6
88 - 6 = 82
```

Example Output #3:

```
Please select operation (ADD, SUB, MLP) > DIV
Please enter the first number > 5
Please enter the second number > 5
I do not understand that operation!
```

Extension task is on the next page



** Extension Challenge – Earn a bonus mark!

Edit your code so that it will ask the user for a second operator and a third number. It will then perform the second operation to the result of the first calculation (the code which you just made for the first part of this assignment). Effectively, doing this:

OUTPUT = (number1 [operator 1] number2) [operator2] number3

The output should be printed on the screen EXACTLY as shown in the examples.

Note: you *MUST STILL* only use switch cases to solve this assignment.

Example Extension Output #1:

```
Please select first operation (ADD, SUB, MLP) > ADD
Please select second operation (ADD, SUB, MLP) > MLP
Please enter the first number > 8
Please enter the second number > 5
Please enter the third number > 3
(8 + 5) x 3 = 39
```

Example Extension Output #2:

```
Please select first operation (ADD, SUB, MLP) > MLP
Please select second operation (ADD, SUB, MLP) > SUB
Please enter the first number > 6
Please enter the second number > 9
Please enter the third number > 2
(6 x 9) - 2 = 52
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5

Level 4: Iteration (easy-medium)

Task 4 – Repeat after me

4A – Repeat After Me

Java Programming Tasks (V2)

Task: Create a program that will ask the user to enter a word. The program must then output that word 10 times.

Note: you *MUST* use a *FOR* loop to solve this.

Example Output #1:

```
Please enter your word > hello
```

Example Output #2:

```
Please enter your word > turkey
```

** Extension Challenge – Earn a bonus mark!

Change the program so that it will ask the user how many times they wish to repeat the given word, and then output the word that many times.

Example Extension Output #1:

```
Please enter your word > chicken
How many times shall I repeat chicken? 3
chicken
chicken
chicken
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 4B Spellman

4B – Spellman**Java Programming Tasks (V2)**

Task: Create a program that will ask the user to enter a 7 letter word. The program must then output the word one letter at a time on subsequent lines.

Note: you *MUST* use a *FOR loop* to solve this.

Hint: You will need to use *.charAt()* to solve this problem.

Example Output #1:

```
Please enter your 7 letter word > updates
u
p
d
a
t
e
s
```

Example Output #2:

```
Please enter your 7 letter word > purpose
p
u
r
p
o
s
e
```

**** Extension Challenge – Earn a bonus mark!**

Change the program so that it will build the word up slowly; adding the next letter onto the last ones on subsequent lines.

Example Extension Output #1:

```
Please enter your 7 letter word > purpose
p
pu
pur
purp
purpo
purpos
purpose
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 4C Alphabet Loop

4C – Alphabet Loop**Java Programming Tasks (V2)**

Task: Create a String variable which contains all 26 letters of the alphabet (A-Z). Your program should output the letters of the alphabet, one per line, with a number showing its position in the alphabet (NOTE – position as humans would use and NOT the index number).

Note: you *MUST* use a *FOR loop* to solve this.

Hint: You will need to use *.charAt()* to solve this problem.

Example Output:

```
1 - A  
2 - B  
3 - C  
4 - D  
5 - E  
6 - F  
7 - G  
8 - H  
9 - I  
10 - J  
11 - K  
12 - L  
13 - M  
14 - N  
15 - O  
16 - P  
17 - Q  
18 - R  
19 - S  
20 - T  
21 - U  
22 - V  
23 - W  
24 - X  
25 - Y  
26 - Z
```

Extension task is on the next page



4C – Alphabet Loop

Java Programming Tasks (V2)

** Extension Challenge – Earn a bonus mark!

Change the program so that it prints the alphabet out but in reverse!

Note: you *MUST* use a *FOR loop* to solve this.

Hint: You do not need to add any additional code to complete this, you must simply re-arrange and alter slightly how the loop functions.

Example Extension Output:

26	-	Z
25	-	Y
24	-	X
23	-	W
22	-	V
21	-	U
20	-	T
19	-	S
18	-	R
17	-	Q
16	-	P
15	-	O
14	-	N
13	-	M
12	-	L
11	-	K
10	-	J
9	-	I
8	-	H
7	-	G
6	-	F
5	-	E
4	-	D
3	-	C
2	-	B
1	-	A



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

A horizontal scale consisting of five empty square boxes numbered 1 through 5. A red arrow points from the left towards the number 5.

1	2	3	4	5



Task 4D Doubling Score

4D – Doubling Scale

Java Programming Tasks (V2)

Task: Ask the user to enter a number. The program should then double this number and output to the screen. The doubled number should then be doubled and outputted. This process should repeat 5 times.

Note: you MUST use a FOR loop to solve this.

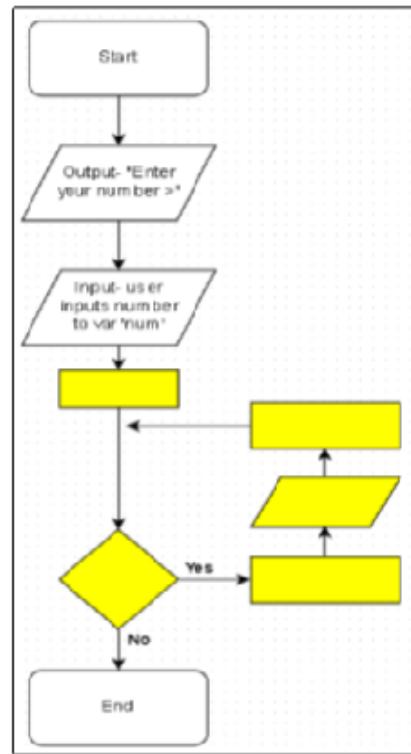
Example Output #1:

```
Enter your number > 2
4
8
16
32
64
```

Example Output #2:

```
Enter your number > 5
10
20
40
80
160
```

You must then also **COMPLETE** the partial flowchart to the right.



NOTE: For this solution, a flowchart must also be completed within flowgorithm or your own choice of tool



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreen/screenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Flowchart

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 4E Green Bottles

4E – Green Bottles (hanging on the wall)

Java Programming Tasks (V2)

Task: Following the popular nursery rhyme “Ten Green Bottles”, create a program that will ask the user how many green bottles are hanging on their wall. It will then produce the lyrics for the song from this number right down to 0. Lyrics for the song can be found here: http://www.bbc.co.uk/learning/schoolradio/subjects/mathematics/countingsongs/G-Z/ten_green_bottles

Note: you **MUST** use a **FOR** loop to solve this.

Example Output:

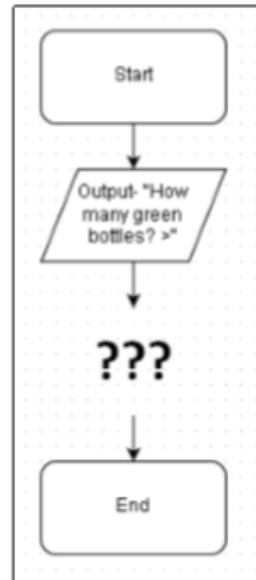
```
How many bottles are on the wall? > 3
3 green bottles, hanging on the wall,
3 green bottles, hanging on the wall,
and if one green bottle, should accidentally fall,
there'll be 2 green bottles, hanging on the wall!

2 green bottles, hanging on the wall,
2 green bottles, hanging on the wall,
and if one green bottle, should accidentally fall,
there'll be 1 green bottles, hanging on the wall!

1 green bottles, hanging on the wall,
1 green bottles, hanging on the wall,
and if one green bottle, should accidentally fall,
there'll be 0 green bottles, hanging on the wall!
```

You must then also **COMPLETE** the partial flowchart to the right.

Use: <https://www.draw.io/> to create the flowchart.



NOTE: For this solution, a flowchart must also be completed within flowgorithm or your own choice of tool

Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot
CODE
(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

Flowchart

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

1	2	3	4	5

Task 4F Factors

4F - Factors**Java Programming Tasks (V2)**

Task: Write a program that will display all of the positive factors for a given number that the user enters.

If you wish to remind yourself what a factors is, or need a calculator to check your program works correctly, try this site:

<https://www.mathsisfun.com/numbers/factors-all-tool.html>

Note: you *MUST* use a *FOR* loop to solve this.

Hint: Not sure how to solve this? Look back at task 3N...

Example Output #1:

```
What is your number? > 54
The factors of 54 are > 1, 2, 3, 6, 9, 18, 27, 54,
```

Example Output #2:

```
What is your number? > 302
The factors of 302 are > 1, 2, 151, 302,
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 4G Loops with numbers

4G – Loops with numbers

Java Programming Tasks (V2)

Task: Create a program that will ask the user to enter a number. The program should then output the preceding 3 even numbers.

Example Output #1:

```
Please enter a number > 5
The next 3 EVEN numbers are:
6
8
10
```

Example Output #2:

```
Please enter a number > 102
The next 3 EVEN numbers are:
104
106
108
```

** Extension Challenge – Earn a bonus mark!

Change the program so that the user can decide if they would like the program to output the next 3 odd OR even numbers.

Example Extension Output #1:

```
Please enter a number > 5
Do you want odd or even numbers? (odd/even) > odd
The next 3 odd numbers are:
7
9
11
```

Example Extension Output #2:

```
Please enter a number > 5
Do you want odd or even numbers? (odd/even) > even
The next 3 even numbers are:
6
8
10
```

Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot
CODE
(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

1	2	3	4	5



Task 4H Block Maker

4H – Block Maker

Java Programming Tasks (V2)

Task: Create a program that will ask the user to enter a number. The program will then create a block of Xs' that is the user's number squared.

Hint: You will need to use a loop within a loop to achieve this.

Example Output #1:

```
Please enter a number > 3
XXX
XXX
XXX
```

Example Output #2:

```
Please enter a number > 8
XXXXXXXXX
XXXXXXXXX
XXXXXXXXX
XXXXXXXXX
XXXXXXXXX
XXXXXXXXX
XXXXXXXXX
XXXXXXXXX
XXXXXXXXX
```

**** Extension Challenge – Earn a bonus mark!**

Modify your program so that it will produce a triangle with the base the size the user entered:

Example Extension Output #1:

```
Please enter a number > 9
X
XX
XXX
XXXX
XXXXX
XXXXXX
XXXXXXX
XXXXXXX
XXXXXXX
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 4I Word Pyramid

4I – Word Pyramid

Java Programming Tasks (V2)

Task: Create a program that will ask the user to enter a word. The program will then display this word, letter by letter, until it is complete as in the example outputs.

Hint: You will need to know the LENGTH of the word to know how many times to run the loop, possibly using CharAt each time to build up the word...

Example Output #1:

```
Please enter a word > Egypt
E
Eg
Egy
Egyp
Egypt
```

Example Output #2:

```
Please enter a word > Tomb
T
To
Tom
Tomb
```

** Extension Challenge – Earn a bonus mark!

Modify your program so that it will remove one letter at a time after the word has been completed, as seen in the example below.

Hint: This will require the use of the Java substring() method which you will need to look up how it works. Then think how you can implement it into your loops.

Example Extension Output #1:

```
Please enter a word > mummy
m
mu
mum
mummo
mummy
mumm
mum
mu
m
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 4J Plus Me

4J – Plus Me**Java Programming Tasks (V2)**

Task: Create a program that will ask the user to enter three numbers. The user will then be asked to enter the sum of the numbers, to which the program will continue to ask until the user enters the correct answer.

Hint: You will need to use a while loop to complete this.

Example Output #1:

```
Please enter no 1 > 7  
Please enter no 2 > 3  
Please enter no 3 > 10  
What is the sum? > 10  
Sorry, try again!  
What is the sum? > 20  
Correct! Well done!
```

Example Output #2:

```
Please enter no 1 > 5  
Please enter no 2 > 10  
Please enter no 3 > 5  
What is the sum? > 20  
Correct! Well done!
```

**** Extension Challenge – Earn a bonus mark!**

Modify your program so that the program will randomly select 3 numbers between 1 and 10. The user must then guess the total.

Hint: You will need to use Math Random for this:

```
(int) (Math.random() *max+min);
```

Example Extension Output #1:

```
Generating random numbers...  
What is the sum? > 5  
Sorry, try again!  
What is the sum? > 20  
Correct! Well done!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 4K Word Size Checker

4K – Word Size Checker

Java Programming Tasks (V2)

Task: Create a program that will read in a word. If the word has 6 letters, it should be outputted in capital letters backwards. If not, it should tell the user how many letters there are. The program should continue to repeat this process until the user enters “Japan” as their word of choice.

Hint: You will need to use a while loop to complete this.

Example Output #1:

```
Enter a word > chess
chess has 5 letters
Enter a word > google
Magic six! ELGOOG
Enter a word > Japan
Goodbye!
```

Example Output #2:

```
Enter a word > rat
rat has 3 letters
Enter a word > DVDrOM
Magic six! MORDDVD
Enter a word > china
china has 5 letters
Enter a word > Japan
Goodbye!
```

Submission: Submit your solution (copy the entire contents of the class) in .doc, .docx or .rtf format to Edmodo before the deadline. **Ensure you have attached your completed file.**

** Extension Challenge – Earn a bonus mark!

Modify your program so that there is a 30% chance it will randomly declare “Game Over”

*Hint: You will need to use Math Random for this: (int)(Math.random() *max+min);*

Example Extension Output #1:

```
Enter a word > Simon
Simon has 5 letters
Enter a word > looroll
looroll has 7 letters
Enter a word > baggins
baggins has 7 letters
I randomly declare... GAME OVER!
```

Example Extension Output #2:

```
Enter a word > happy
happy has 5 letters
I randomly declare... GAME OVER!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5

Task 4L While Calc

4L - WhileCalc

Java Programming Tasks (V2)

Task: Create a program that will ask the user for their name followed by two numbers. The program should then add the two numbers and output the total to the user. The user should then specify if they wish to go again. If they do, then the process should repeat until the user chooses "no".

Note: Pay careful attention to what parts of the program do and do not repeat, as not everything will be inside of your loop.

Example Output #1:

```
Welcome to the calculator!
What is your name?
Marcus
Ok Marcus, what is your first number?
256
Thanks! What is the second number?
333
Calculating...
The total is 589
Would you like to add more numbers? (yes/no)
no
Ok, byebye!
```

Example Output #2:

```
Welcome to the calculator!
What is your name?
Samuel
Ok Samuel, what is your first number?
666
Thanks! What is the second number?
957
Calculating...
The total is 1623
Would you like to add more numbers? (yes/no)
yes
Ok Samuel, what is your first number?
985
Thanks! What is the second number?
12121
Calculating...
The total is 13106
Would you like to add more numbers? (yes/no)
no
Ok, byebye!
```



Extension task is on the next page

4L - WhileCalc

Java Programming Tasks (V2)

** Extension Challenge – Earn a bonus mark!

Modify your program so that each time it will ask the user to decide if they wish to add, subtract or multiply the two numbers together. The answer given should reflect the user's decision.

Example Extension Output #1:

```
Welcome to the calculator!
What is your name?
Jack
Ok Jack, what is your first number?
25
Thanks! What is the second number?
15
And what shall we do with them? (add/sub/mul)
mul
Calculating...
25*15 is 375
Would you like to add more numbers? (yes/no)
yes
Ok Jack, what is your first number?
2144
Thanks! What is the second number?
369
And what shall we do with them? (add/sub/mul)
sub
Calculating...
2144-369 is 1775
Would you like to add more numbers? (yes/no)
no
Ok, byebye!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 4M Block Maker 2

4M – Block Maker 2**Java Programming Tasks (V2)**

Task: Create a program that will ask the user to specify how big they want their block to be. The program must then output a block of this size that is hollow (only Xs' around the edge, as in the examples).

Note: You can use any type of loop to solve this problem, however it is easier with a FOR loop. You may also need to use selection within the loop.

Example Output #1:

```
How big is the block?  
4  
XXXX  
X  X  
X  X  
XXXX
```

Example Output #2:

```
How big is the block?  
10  
XXXXXXXXXX  
X      X  
X      X  
X      X  
X      X  
X      X  
X      X  
X      X  
X      X  
XXXXXXX
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5

Level 5: Algorithmic Thinking (medium)

Task 5B Biggest Number

5B – Biggest Number

Java Programming Tasks (V2)

Task: Create a program that will ask the user to enter 3 whole numbers. The program must then display which is the largest of the three.

Remember: You are looking for the most efficient solution to this problem.

Example Output #1:

```
Enter first number > 23
Enter second number > 16
Enter third number > 27
The third number (27) was the biggest
```

Example Output #2:

```
Enter first number > 100
Enter second number > 101
Enter third number > 99
The second number (101) was the biggest
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 5C Maths Trainer

5C – Maths Trainer**Java Programming Tasks (V2)**

Task: Create a program that will ask the user an addition question which has two random operands (chosen by the computer that are between 1 and 15). The user will enter their answer and the computer will tell them if they are correct or not. The program will keep asking questions until the user specifies they want to stop.

Remember: You are looking for the most efficient solution to this problem.

Hint: You will need to use Math Random for this: (`int`) (`Math.random() *max+min`) ;

Hint #2: Have a look back at 4J if you are stuck.

Example Output #1:

```
Welcome to Maths Trainer!
What is 1 + 13? > 14
Correct! Well done!
Would you like another question? (Y/N) > Y
What is 10 + 11? > 15
Incorrect! The right answer is: 21
Would you like another question? (Y/N) > N
Ok, adios!
```

Example Output #2:

```
Welcome to Maths Trainer!
What is 8 + 10? > 18
Correct! Well done!
Would you like another question? (Y/N) > Y
What is 11 + 9? > 20
Correct! Well done!
Would you like another question? (Y/N) > Y
What is 11 + 3? > 20
Incorrect! The right answer is: 14
Would you like another question? (Y/N) > N
Ok, adios!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

→

1	2	3	4	5



Task 5E Lock Picker

5E – Lockpicker

Java Programming Tasks (V2)

Task: In order to open the electronic lock your program must generate the number 55. Write a program that will generate random numbers between 10 and 99. The program should continue to repeat until the number 55 is created. The program should output how many attempts it made to get 55.

Remember: You are looking for the most efficient solution to this problem.

Hint: You will need to use Math Random for this: (int) (Math.random() *max+min);

Example Output #1:

```
Picking the lock...
38
63
51
46
12
22
30
71
55
That took 9 tries!
```

Example Output #2:

```
Picking the lock...
98
75
32
77
28
50
39
59
81
19
84
98
26
21
17
82
96
55
That took 18 tries!
```

Example Output #3:

```
Picking the lock...
22
76
44
83
85
11
91
41
44
30
21
77
69
68
17
36
55
That took 17 tries!
```

** Extension Challenge – Earn a bonus mark!

Modify your program so the user is asked to enter the lock number (so it's not 55 by default).

Example Extension Output:

```
Enter lock number > 15
Picking the lock...
68
56
```

[screenshot cut down]

```
81
15
That took 247 tries!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 5F Season Checker

5F – Season Checker

Java Programming Tasks (V2)

Task: Your task is to create a program that will read in a date and determine which season it falls within. Your program should read in a DAY and a MONTH integer separately. Assume all months have 31 days so DAY will be 1-31 and MONTH will be 1-12 (1 for January, 2 for February and so on).

Here are the season ranges:

Season	Start Date	End Date
Winter	16/12	15/3
Spring	16/3	15/6
Summer	16/6	15/9
Autumn	16/9	15/12

Remember: You are looking for the most efficient solution to this problem.

Hint: Try writing this down on paper first to see how you could solve it. There are multiple methods of doing so.

Example Output #1:

```
Enter the day > 15
Enter the month > 12
You entered 15/12
That falls in autumn!
```

Example Output #2:

```
Enter the day > 22
Enter the month > 4
You entered 22/4
That falls in spring!
```

Example Output #3:

```
Enter the day > 8
Enter the month > 8
You entered 8/8
That falls in summer!
```

** Extension Challenge – Earn a bonus mark!

Modify your program so the process repeats until the user opts to stop:

Example Extension Output:

```
Enter the day > 5
Enter the month > 12
You entered 5/12
That falls in autumn!
Do you wish to check another date? (Y/N) > Y
Enter the day > 3
Enter the month > 2
You entered 3/2
That falls in winter!
Do you wish to check another date? (Y/N) > N
Enjoy the seasons!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 5M Name Mixer

5M – Name Mixer

Java Programming Tasks (V2)

Task: Your task is to create a program that will ask the user to enter their first and last name. The program should then mix the names up! It should swap the first two letters of the first name with the first two letters of the last name, then output this new name.

Remember: You are looking for the most efficient solution to this problem.

Hint: You will need to use both the `substring(x)` and `substring(x,x)` – you may need to look up how these work.

Example Output #1:

```
Enter first name > Harry
Enter last name > Potter
Hi there Porry Hatter!
```

Example Output #2:

```
Enter first name > Marcus
Enter last name > Parnell
Hi there Parcus Marnell!
```

Submission: Submit your solution (copy the entire contents of the class) in .doc, .docx or .rtf format to Edmodo before the deadline. **Ensure you have attached your completed file.**

** Extension Challenge – Earn a bonus mark!

Modify your program so that it will first calculate the length of the names and simply mix up half of each.

Example Extension Output:

```
Enter first name > Nicole
Enter last name > Sherzinger
Hi there Sherzole Nicinger!
```

**Notes:****SUBMISSION:** Paste your code into the next row along with the 'output' as a printscreen/screenshot**CODE**

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION**Reflections***What problems did you have in producing this solution?**How could you improve for next time?**Overall, how difficult was it? (fill the boxes with colour)***EASY****VERY DIFFICULT**A horizontal scale consisting of five empty square boxes numbered 1 through 5. An arrow points from the left side of the scale towards the number 5, indicating a progression from easy to very difficult.

1	2	3	4	5



Task 5R InCommon

5R – InCommon**Java Programming Tasks (V2)**

Task: Write a program that will ask the user to enter two sentences. The program should then compare them and output a list of all letters that are in common between the two sentences.

Remember: You are looking for the most efficient solution to this problem.

Hint: One way to solve this is to create a variable with the alphabet stored inside of it and then compare the letters of both sentences against this to see what matches...

Example Output #1:

```
Enter sentence 1:  
Life on Mars is fun  
Enter sentence 2:  
Foxes have no morals  
The letters in common are: A E F L M N O R S
```

Example Output #2:

```
Enter sentence 1:  
Never wake a sleeping lion  
Enter sentence 2:  
Duck quacks do not echo  
The letters in common are: A E K N O S
```

**** Extension Challenge – Earn a bonus mark!**

Modify your program so that it not only outputs a list of common letters, but it sorts them into vowels and consonants.

Example Extension Output:

```
Enter sentence 1:  
The pink flamingo  
Enter sentence 2:  
The orange camel  
The letters in common are: A E G H L M N O T  
The vowels in common are: A E O  
The consonants in common are: G H L M N T
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5

Level 6: Arrays

Task 6A Little Boxes

6A – Little Boxes #1

Java Programming Tasks (V2)

Task: Create a program that will ask the user to enter 5 integers. These numbers should be stored into an array (which is 5 boxes long) which is called 'boxes'. After the user has entered all 5 numbers, the program should calculate the average of the numbers entered.

Your program must then ask the user if they want to go again. The process should repeat until the user chooses to stop.

Hint: This task will require a loop to put the numbers into an array.

Hint: The average is not going to be a whole number, which means its datatype is not int.

Example Output #1:

```
Please enter number 1 > 2
Please enter number 2 > 6
Please enter number 3 > 4
Please enter number 4 > 10
Please enter number 5 > 8
The average is 6.0
That was fun! Would you like to do that again? (Y/N) N
Ok then! Zàijiàn!
```

Example Output #2:

```
Please enter number 1 > 67
Please enter number 2 > 87
Please enter number 3 > 54
Please enter number 4 > 110
Please enter number 5 > 99
The average is 83.4
That was fun! Would you like to do that again? (Y/N) Y
Please enter number 1 > 3
Please enter number 2 > 6
Please enter number 3 > 5
Please enter number 4 > 2
Please enter number 5 > 0
The average is 3.2
That was fun! Would you like to do that again? (Y/N) N
Ok then! Zàijiàn!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 6A Little Boxes

6B – Little Boxes #2

Java Programming Tasks (V2)

Task: Create a program that will ask the user to enter 5 characters. These characters should be stored into an array (which is 5 boxes long) which is called ‘boxes’. After the user has entered all 5 characters, the program should then output them as shown in the examples (note the added dash that exists only between characters).

Your program must then ask the user if they want to go again. The process should repeat until the user chooses to stop.

Hint: This task will require a loop to put the characters into an array.

Example Output #1:

```
Please enter character 1 > A
Please enter character 2 > 6
Please enter character 3 > ^
Please enter character 4 > %
Please enter character 5 > /
Here is your output: A - 6 - ^ - % - /
That was fun! Would you like to do that again? (Y/N) N
Ok then! Zàijiàn!
```

Example Output #2:

```
Please enter character 1 > 4
Please enter character 2 > $
Please enter character 3 > £
Please enter character 4 > $
Please enter character 5 > *
Here is your output: 4 - $ - £ - $ - *
That was fun! Would you like to do that again? (Y/N) Y
Please enter character 1 > £
Please enter character 2 > €
Please enter character 3 > ^
Please enter character 4 > <
Please enter character 5 > >
Here is your output: £ - € - ^ - < - -
That was fun! Would you like to do that again? (Y/N) N
Ok then! Zàijiàn!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreen/screenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 6C Number Hunt

6C – NumberHunt

Java Programming Tasks (V2)

Task: Create a program that will create 20 random whole numbers between 10 and 40, adding them to an array called 'boxes'. The program should then return if the array contains the number 19.

Hint: You will need to use Math Random for this:

```
(int) (Math.random() *max+min);
```

Example Output #1:

```
Generating numbers...
21-31-34-26-15-26-37-32-18-28-23-12-14-14-13-10-33-33-32-37
The array does not contain 19!
```

Example Output #2:

```
Generating numbers...
15-23-11-22-26-35-25-38-17-36-19-30-26-13-11-11-22-28-12-16
The array contains 19!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 6D Backward Boxes

6D – Backward Boxes

Java Programming Tasks (V2)

Task: Create a program that will store an array of 8 boxes. The program should read in a symbol, character or number to each box. The program should then output these BACKWARDS.

The program should loop until the user opts to stop.

Hint: You will need to use an array to solve this problem.

Example Output #1:

```
Please enter character 1 > G
Please enter character 2 > n
Please enter character 3 > i
Please enter character 4 > z
Please enter character 5 > @
Please enter character 6 > m
Please enter character 7 > A
Please enter character 8 > #
Here is your output: #Am@ZinG
That was fun! Would you like to do that again? (Y/N) N
Ok then! adjö!
```

Example Output #2:

```
Please enter character 1 > A
Please enter character 2 > ^
Please enter character 3 > a
Please enter character 4 > J
Please enter character 5 > #
Please enter character 6 > L
Please enter character 7 > O
Please enter character 8 > L
Here is your output: L@L#Ja^A
That was fun! Would you like to do that again? (Y/N) Y
Please enter character 1 > D
Please enter character 2 > I
Please enter character 3 > Z
Please enter character 4 > g
Please enter character 5 > %
Please enter character 6 > ^
Please enter character 7 > j
Please enter character 8 > $
Here is your output: $j^%g2iD
That was fun! Would you like to do that again? (Y/N) N
Ok then! adjö!
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 6F Array Averages

6F – Array Averages

Java Programming Tasks (V2)

Task: Create a program that will generate 30 random numbers between 25 and 100 and put them into an array. The program should then output the average number for that series of numbers.

Hint: `(int) (Math.random() * ((Max - Min) + 1)) + Min;`

Example Output #1:

```
Generating numbers...
The average number for that series is 65.16666666666667
```

Example Output #2:

```
Generating numbers...
The average number for that series is 57.8
```

Example Output #3:

```
Generating numbers...
The average number for that series is 60.833333333333336
```

Submission: Submit your solution (copy the entire contents of the class) in .doc, .docx or .rtf format to Edmodo before the deadline. **Ensure you have attached your completed file.**



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreen/screenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 6G Array Min

6G – Array Min

Java Programming Tasks (V2)

Task: Create a program that will generate 40 random numbers between 10 and 200 and put them into an array. The program should then output the smallest number of that series generated.

Hint: `(int) (Math.random() * ((Max - Min) + 1)) + Min;`

Example Output #1:

```
Generating numbers...
The smallest number in this series is 12
```

Example Output #2:

```
Generating numbers...
The smallest number in this series is 11
```

** Extension Challenge – Earn a bonus mark!

Modify your program so that it also prints out the largest number in the series!

Example Extension Output #1:

```
Generating numbers...
The smallest number in this series is 17
The largest number in this series is 193
```

Example Extension Output #2:

```
Generating numbers...
The smallest number in this series is 12
The largest number in this series is 198
```



Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot

CODE

(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

EASY

VERY DIFFICULT

1	2	3	4	5



Task 6A Little Boxes

6I – Number Sorter

Java Programming Tasks (V2)

Task: Create a program that will read in five user-defined numbers to an array. The program should then sort the numbers in ascending order and output the order to the user.

Example Output #1:

```
Enter number #1 > 6
Enter number #2 > 8
Enter number #3 > 1
Enter number #4 > 3
Enter number #5 > 5
Sorted! 1 3 5 6 8
```

Example Output #2:

```
Enter number #1 > 101
Enter number #2 > 50
Enter number #3 > 66
Enter number #4 > 52
Enter number #5 > 3
Sorted! 3 50 52 66 101
```

Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot
CODE
(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

1	2	3	4	5



Task 6N Letter Sorter

6N – Letter Sorter

Java Programming Tasks (V2)

Task: Create a program that will read in user-defined word, of any length, to an array – placing one letter in each part of the array. The program should then sort the letters alphabetically and output the result to the screen.

Hint: You do not know how long the array needs to be until the user enters their word. You may need to consider this in the order in which you create things in the program.

Hint #2: To prevent difficulties with this program, ensure that transform the entirety of the user's word to uppercase before storing in the array.

Example Output #1:

```
Please enter your word > chicken  
Sorted! C C E H I K N
```

Example Output #2:

```
Please enter your word > PAinkillEr  
Sorted! A E I I K L L N P R
```

** Extension Challenge – Earn a bonus mark!

Modify your program so that it will also indicate the number of vowels and consonants that make up the word entered.

Example Extension Output #1:

```
Please enter your word > Francesca  
Sorted! A A C C E F N R S  
FRANCESCA contains 3 vowels and 6 consonants.
```

Example Extension Output #2:

```
Please enter your word > Inquisition  
Sorted! I I I I N N O Q S T U  
INQUISITION contains 6 vowels and 5 consonants.
```

Notes:

SUBMISSION: Paste your code into the next row along with the 'output' as a printscreenscreenshot
CODE
(this is the actual program code including the main method)

OUTPUT

(this is the result of running your code)

EXTENSION

Reflections

What problems did you have in producing this solution?

How could you improve for next time?

Overall, how difficult was it? (fill the boxes with colour)

1	2	3	4	5



Task 60 Super Sorter

6O – Super Sorter

Java Programming Tasks (V2)

Task: Create a program that hold the below information in three appropriate arrays as named below:

```
nameArray = "Superman", "Batman", "Hulk", "Thor", "Iron Man", "Deadpool", "Groot"  
strengthArray = 5,1,10,6,2,3,4  
availableArray = true, true, true, false, true, true, false
```

Your program should then ask your user to choose how they wish to sort the data; either by name, strength rating or availability. Your program should then return the sorted table of data. It should loop until the user wishes to stop.

Hint: To achieve the table layout, consider the below code for 'printf' command:

```
for(int i = 0; i < 7; i++){  
    System.out.printf("%-12s", nameArray[i] );  
    System.out.printf("%-5s", strengthArray[i] );  
    System.out.println( availableArray[i] );  
}
```

Example Output #1:

Superman	5	true
Batman	1	true
Hulk	10	true
Thor	6	false
Iron Man	2	true
Deadpool	3	true
Groot	4	false

How do you want to sort?

- A) Name
- B) Strength
- C) Availability
- X) Quit

A

Batman	1	true
Deadpool	3	true
Groot	4	false
Hulk	10	true
Iron Man	2	true
Superman	5	true
Thor	6	false

How do you want to sort?

- A) Name
- B) Strength
- C) Availability
- X) Quit

X

See you soon!

Example Output #2:

Superman	5	true
Batman	1	true
Hulk	10	true
Thor	6	false
Iron Man	2	true
Deadpool	3	true
Groot	4	false

How do you want to sort?

- A) Name
- B) Strength
- C) Availability
- X) Quit

B

Batman	1	true
Iron Man	2	true
Deadpool	3	true
Groot	4	false
Superman	5	true
Thor	6	false
Hulk	10	true

How do you want to sort?

- A) Name
- B) Strength
- C) Availability
- X) Quit

X

See you soon!