The Calculator

Requirements

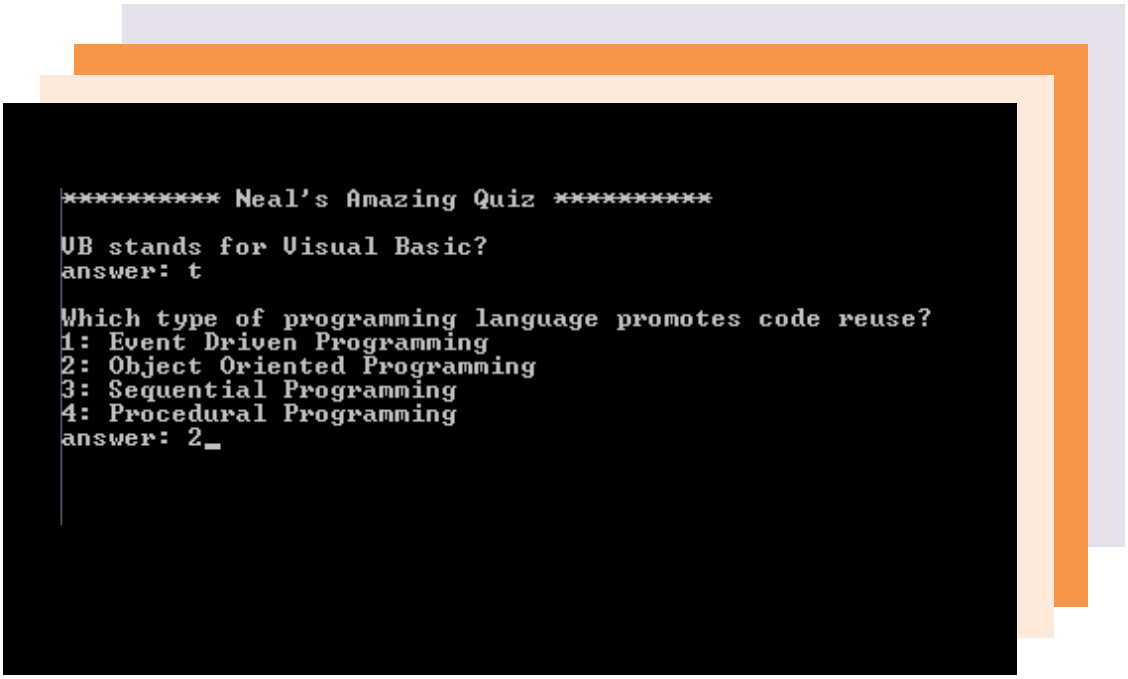
- ☑ Create a calculator application which uses a single procedure (Shared Sub Main)
- ☑ The user should be able to input two numbers and then select a mathematical operation
- ☑ The calculation must then be displayed to the user in full, including the answer

```
Please input your first number: 5
Please input your second number: 2
Please choose one of the following options:
1: +
2: -
3: *
4: /
3
5 * 2 = 10
```

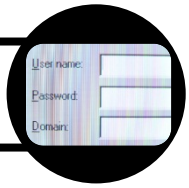
The Quiz

Requirements

- ☑ Create a minimum five-question quiz that uses a single procedure (Shared Sub Main)
- ☑ Use a mixture of SELECT CASE and IF statements
- ☑ The user's score must be displayed at the end of the quiz, along with a message based on how well they performed



The Password App



Requirements

- ☑ Write an application that checks a user's password
- ☑ If the password is correct, your application should write the line 'Access Granted'
- ☑ If the password is incorrect, your application should write the line 'Access Denied'

Additions:

- ☑ Expand the application by adding a username for the user
- ☑ Add another two users to your application

```
Setting environment for using Microsoft Visual Studio 2010 x86 tools.

C:\Program Files\Microsoft Visual Studio 10.0\VC>cd c:\temp

c:\Temp>vbc password.vb /t:exe
Microsoft (R) Visual Basic Compiler version 10.0.30319.233
Copyright (c) Microsoft Corporation.  All rights reserved.


c:\Temp>password.exe
Please enter your username:
Joe
Please enter your password:
letmein

Access Granted

Please enter your username:
```

The Driving Licence

Requirements

- ☑ Create an application that can be used by the DVLA to determine the status of a driver's driving licence
- ☑ The application must allow the user to enter the number of years that the driver has been driving for and the number of penalty points that they have accumulated on their licence
- ☑ The application must then determine the status of the licence based on the following information:

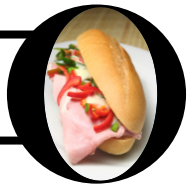
Licence Status:

- ☑ 2 years or less and 6 or more points will result in a disqualification
- ☑ More than 2 years and 12 or more points will result in a time ban
- ☑ Any other combinations will result in the licence status being OK

```
How many years have you been driving for?
6
How many penalty points do you have on your licence?
13

- - - - - - - - LICENCE SUMMARY - - - - - - - - - - - - - -
YOUR LIENCE HAS BEEN TIME-BANNED FOR EXCEEDING 12 POINTS
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

Eat Fresh Subway



Requirements

- ☑ Create an application that can be used in 'Eat Fresh Subway' sandwich restaurants to track and calculate customer orders
- ☑ The application should allow the user to pick the bread type, size and fillings from the set menu. The application should then summarise the order and calculate the bill. The set menu is shown below:

Eat Fresh Menu

Size:

- 6-Inch (£1.65) or 12-Inch (£2.05)

Bread Type:

- Plain (£0.40), Wheat (£0.65), Italian (£0.75), Cheese & Herbs (£0.80)

Fillings:

- Cheese & Tomato (£0.95)
- Italian Bacon & Peperoni (£1.10)
- Tuna & Mayo (£0.95)
- Turkey & Ham (£1.35)
- Chicken Teriyaki (£1.40)
- Steak & Cheese (£1.95)

Desirable Additions:

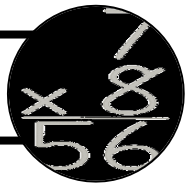
- Add an additional 5% fee if the customer is eating in the restaurant, as opposed to taking it away

```
What size sub do you want (6inch or 12inch)?
6
What bread type do you want (PLAIN, WHEAT, ITALIAN, CHEESE/HERB)?
PLAIN

What filling do you want?
(1) Cheese & Tomato
(2) Italian Bacon & Peperoni
(3) Tuna & Mayo
(4) Turkey & Ham
(5) Chicken Teriyaki
(6) Steak & Cheese
1

- - - - - - - - ORDER SUMMARY - - - - - - - -
```

The Times Table



Requirements

- ☑ Create an application that displays the times table
- ☑ The user must be able to choose the multiplier
- ☑ The user must be able to choose the start and end point of the times table
- ☑ The user must be able to choose to run the application again or exit

```
C:\Temp>multiplier.exe
Please enter the start point number:
3
Please enter the end point number:
9
Please enter the multiplier:
5

********** 5 TIMESTABLE **********

3 X 5 = 15
4 X 5 = 20
5 X 5 = 25
6 X 5 = 30
7 X 5 = 35
8 X 5 = 40
9 X 5 = 45

Do you want to have another go?
1: Yes
2:No
```

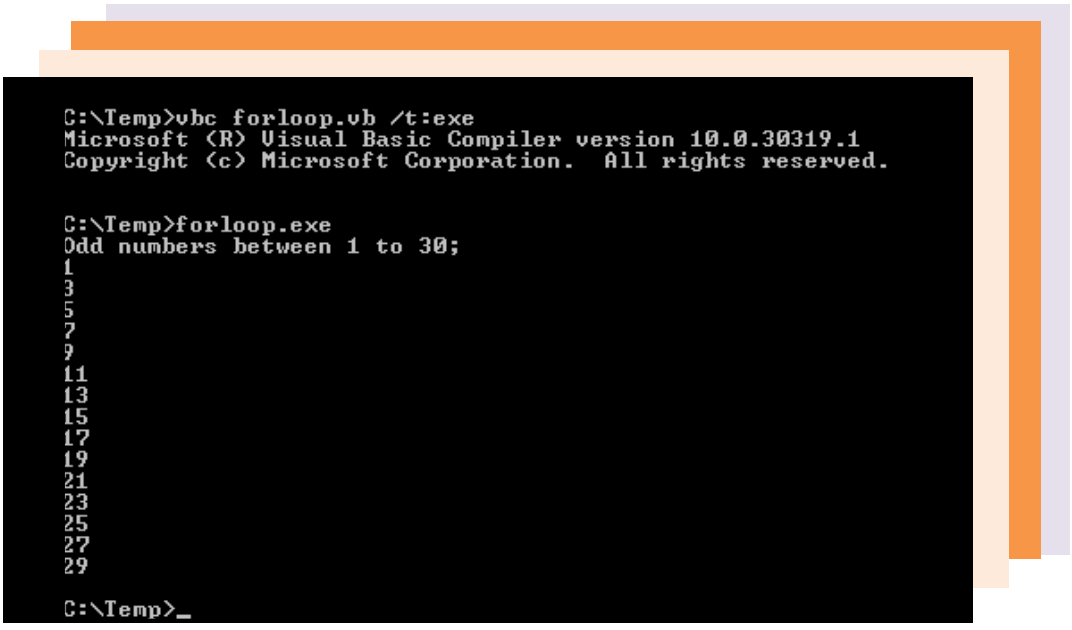
Odds and Evens

Requirements

- ☑ Create an 'odd and even' application which uses a single procedure (Shared Sub Main)
- ☑ The user must be able to input a single number between 1 and 50 into the program
- ☑ The program must then display all odd numbers found in the user's selected number followed by all even numbers found in the user's selected number
- ☑ The program must also display suitable titles

Code Hint:

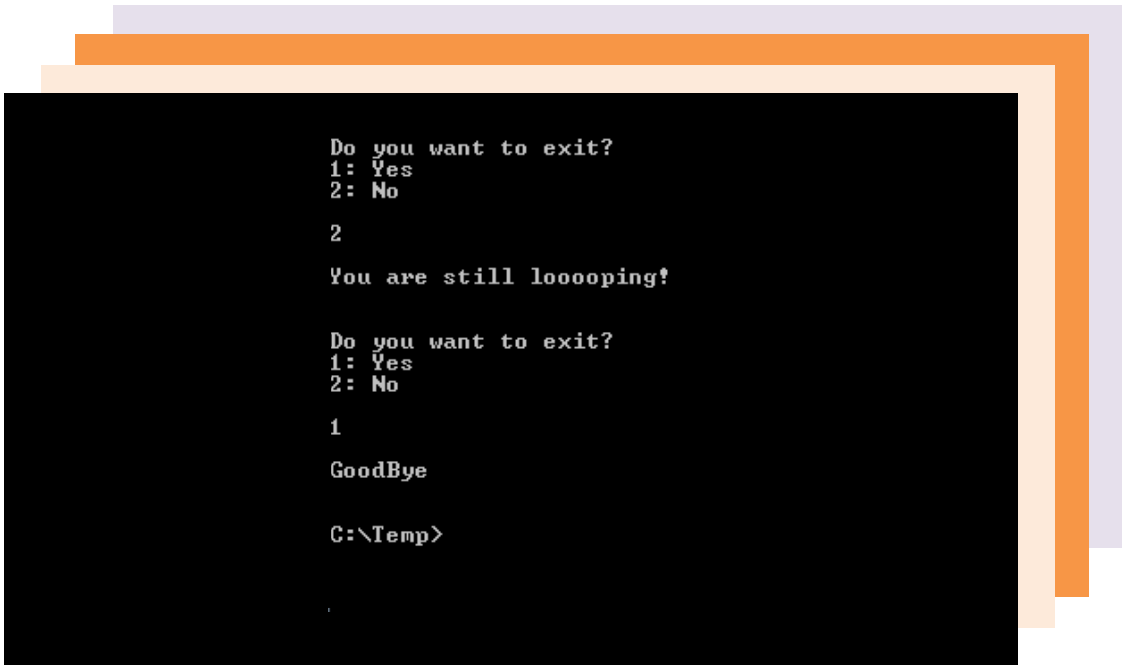
- ☑ For X = 1 to 30 Step 5
- ☑ Step 5 will make the loop increment by 5 as opposed to 1



The Loopy Loop

Requirements

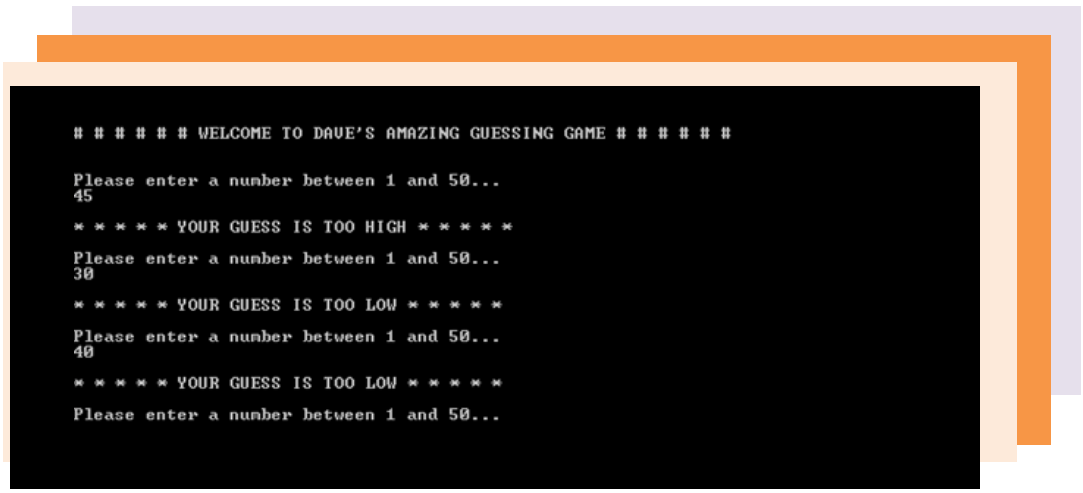
- ☑ Create an application that uses a While Loop to continuously loop, unless the user chooses to exit that loop
- ☑ To do this you will need to use an IF statement nested inside a While Loop



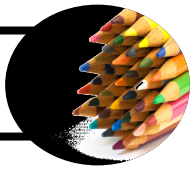
Dave's Amazing Guessing Game

Requirements

- ☑ The game must generate a random number between 1 and 50. The user must then guess what that random number is!
- ☑ The user must keep guessing until their guess matches the answer
- ☑ Every time the user guesses incorrectly they must be given a hint as to whether or not their guess is lower or higher than the answer
- ☑ You will need to use all of the techniques that you have learned over the last few lessons to create this game; variables, IF statements and loops
- ☑ You have your mission; good luck!

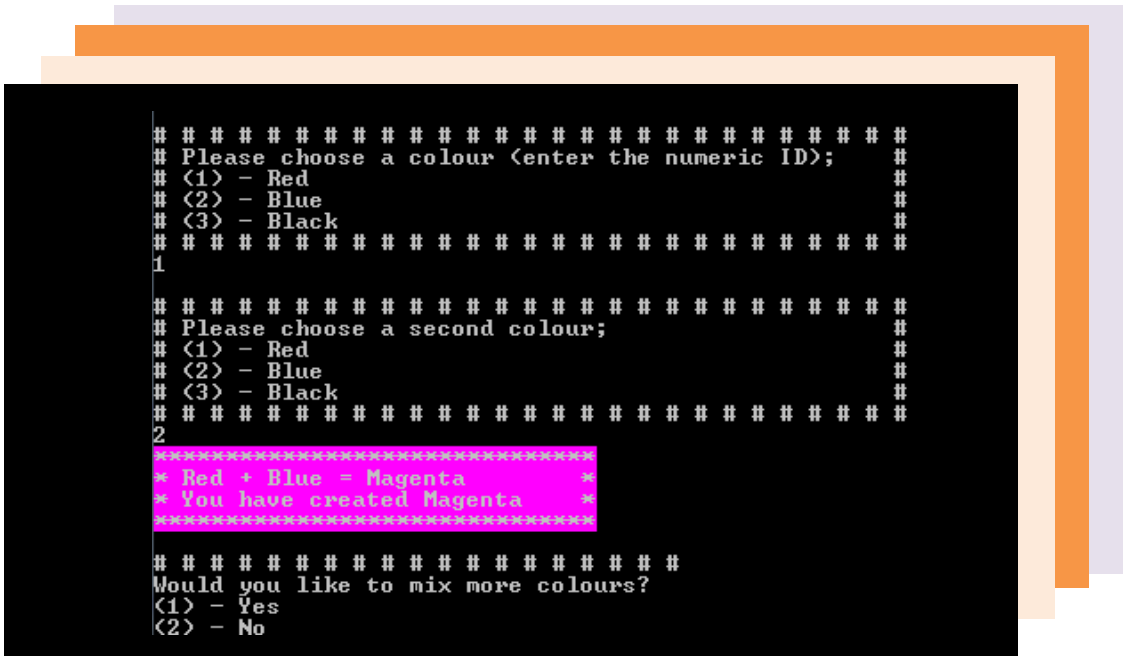


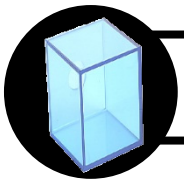
The Colour Mixer



Requirements

- ☑ Create a colour-mixing application (using IF Statements) that will inform the user of the colours they can create when mixing different **colours together**
- ☑ Your application should be able to inform the user of the following **colour mixes**:
  - ✓ Red + Blue = Magenta
  - ✓ Red + Red = Red
  - ✓ Red + Black = Dark Red
  - ✓ Blue + Red = Magenta
  - ✓ Blue + Blue = Blue
  - ✓ Blue + Black = Dark Blue
  - ✓ Black + Red = Dark Red
  - ✓ Black + Blue = Dark Blue
  - ✓ Black + Black = Black





The Cube

REQUIREMENTS

☑ Create an application that allows the user to specify the size of a cube – the application must then use this information to create a cube, of the specified size, using the asterisk or hash symbol

HINT:

✓ You will need to nest a While Loop inside another While Loop to achieve this

✓ The innermost loop will need to use the Write command as opposed to the WriteLine command

```
   at Microsoft.VisualBasic.CompilerServices.Conversions.ToInteger(String Value)

   at nesting.Main()
   at nesting.Main()

C:\Temp>vbc nesting.vb /t:exe
Microsoft (R) Visual Basic Compiler version 10.0.30319.233
Copyright (c) Microsoft Corporation.  All rights reserved.


C:\Temp>nesting.exe

Enter the Width of the quadrilateral:
5
Enter the Length of the quadrilateral:
5


*****
*****
*****
*****
*****
Enter the Width of the quadrilateral:
```

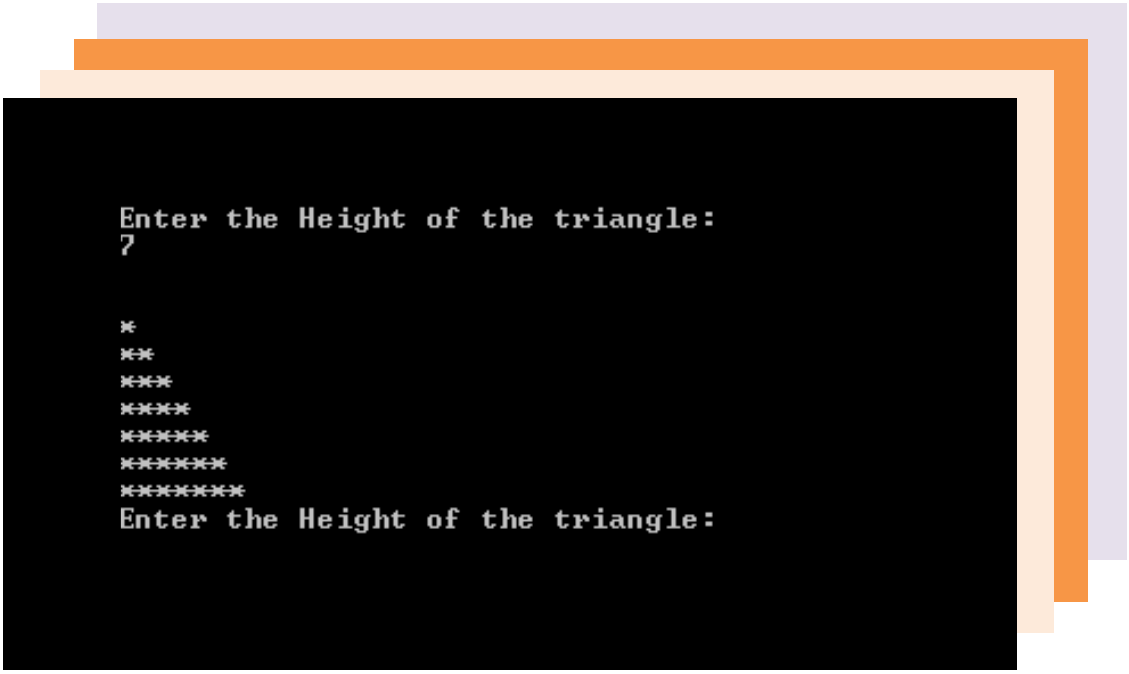
The Triangle

Requirements

- ☑ Create an application that allows the user to specify the height of a triangle – the application must then use this information to create a triangle, of the specified size, using the asterisk or hash symbol

Hint:

- ☑ You will need to nest a For Loop inside a While Loop to achieve this
- ☑ The innermost loop will need to use the Write command as opposed to the WriteLine command



The Currency Converter

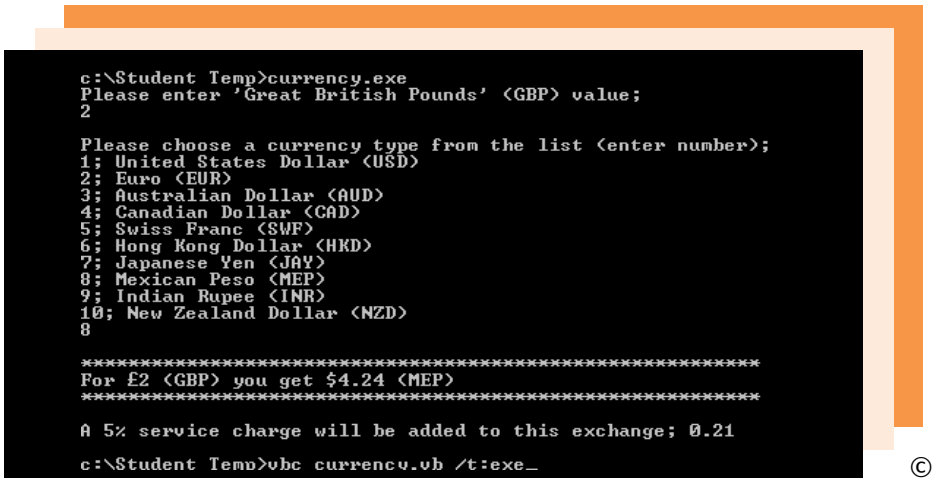
Requirements

- ☑ Write an application that can be used in a travel agency to calculate the exchange rate between various currencies
- ☑ The application must allow the user to enter a value in GBP (Great British Pounds)
- ☑ The user must then be able to choose a currency (from the list below) to exchange the GBP into. The application should display the appropriate information to the user.

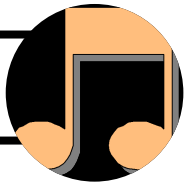
Additions:

- ☑ Calculate a service charge that will be taken from the exchanged currency; the charge is 5% if the amount is below £500, or 3% if the amount is above £500
- ☑ Allow the user to exchange other currencies (from the list below) back into GBP (a 2% service charge is added for this exchange service). Use two methods (subroutines) to achieve this

Exchange Rates			
GBP	United States Dollar (USD)	£1	$1.6117
GBP	Euro (EUR)	£1	€1.2216
GBP	Australian Dollar (AUD)	£1	$1.5578
GBP	Canadian Dollar (CAD)	£1	$1.5985
GBP	Swiss Franc (CHF)	£1	1.4681Fr
GBP	Hong Kong Dollar (HKD)	£1	$1.2507
GBP	Japanese Yen (JPY)	£1	¥1.3173
GBP	Mexican Peso (MXN)	£1	$2.1207
GBP	Indian Rupee (INR)	£1	8.3961Rs
GBP	New Zealand Dollar (NZD)	£1	$1.9793



The Juke Box



Requirements

- ☑ Create an application that can be installed on a modern-day jukebox machine and used by paying customers
- ☑ The application should allow the user to pick a band (from a selection) and then pick a track by that band (again from a selection). The application should then play the song chosen by the user. The bands/songs are listed below (and will be provided for you):

Tracks:
1) 'Come Together' by The Beatles
2) 'She Came In Through The Bathroom Window' by The Beatles
3) 'Eleanor Rigby' by The Beatles
4) 'Lady Madonna' by The Beatles
5) 'Let It Be' by The Beatles

1) 'Beat It' by Michael Jackson
2) 'Billie Jean' by Michael Jackson
3) 'Dont Stop 'Til You Get Enough' by Michael Jackson
4) 'Smooth Criminal' by Michael Jackson

1) 'Is This Love' by Bob Marley
2) 'Could You Be Loved' by Bob Marley
3) 'Get Up Stand Up' by Bob Marley
4) 'I Shot The Sheriff' by Bob Marley

Desirable Requirements:

- ☑ The user must insert more than £1 to play any song. If they insert too much money, then inform the user of how much change they will receive
- ☑ Ask the user if they would like another song; if they answer 'yes', return the user to the start of your program

Code Hints:

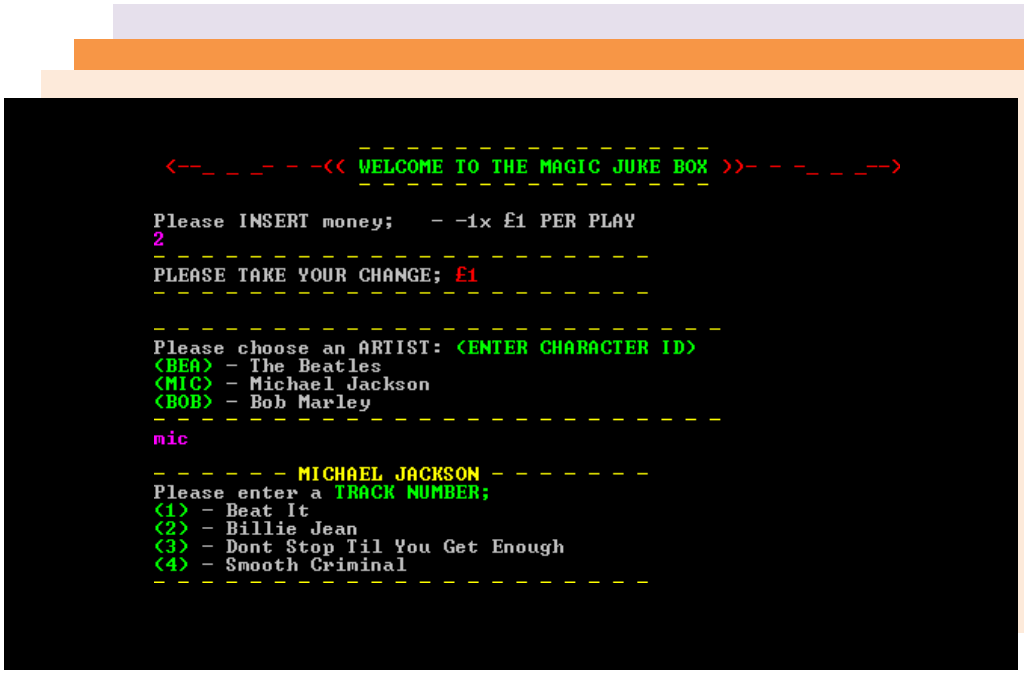
Use this code to play a song (pay attention to the directory of the files):

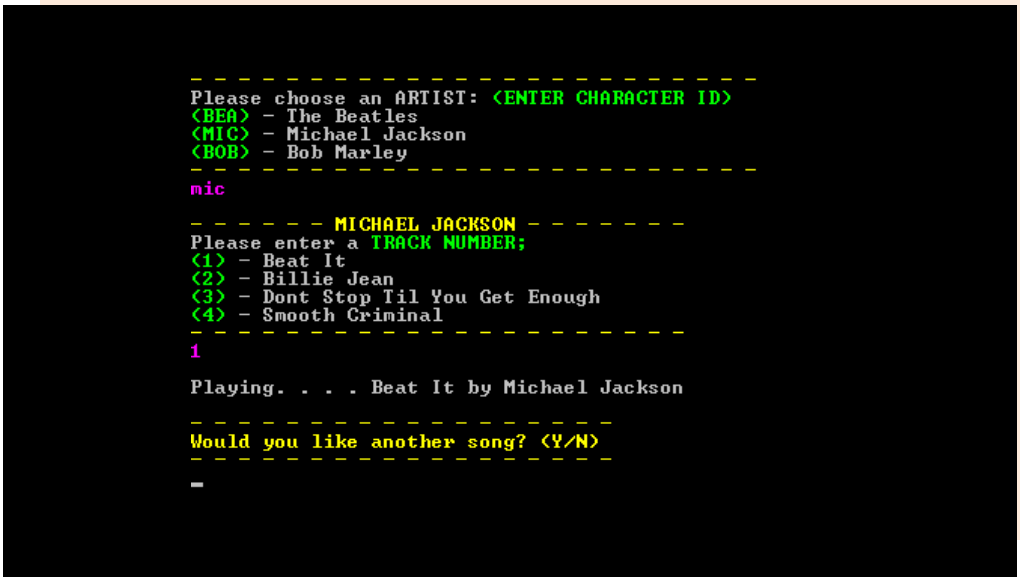
```
My.Computer.Audio.Play(CurDir() & "\JukeBox\BobMarley\04.wav")
```

Use this code to add a fancy title:

```
ForegroundColor = ConsoleColor.Yellow
WriteLine("              ---------------")
ForegroundColor = ConsoleColor.Red
Write(" <-_ _ _-- -{{ ")
ForegroundColor = ConsoleColor.Green
Write("WELCOME TO THE MAGIC JUKE BOX ")
ForegroundColor = ConsoleColor.Red
WriteLine("})- - -_ _ _-> ")
ForegroundColor = ConsoleColor.Yellow
WriteLine("              ---------------")

ReadLine()
```





The Wage Calculator

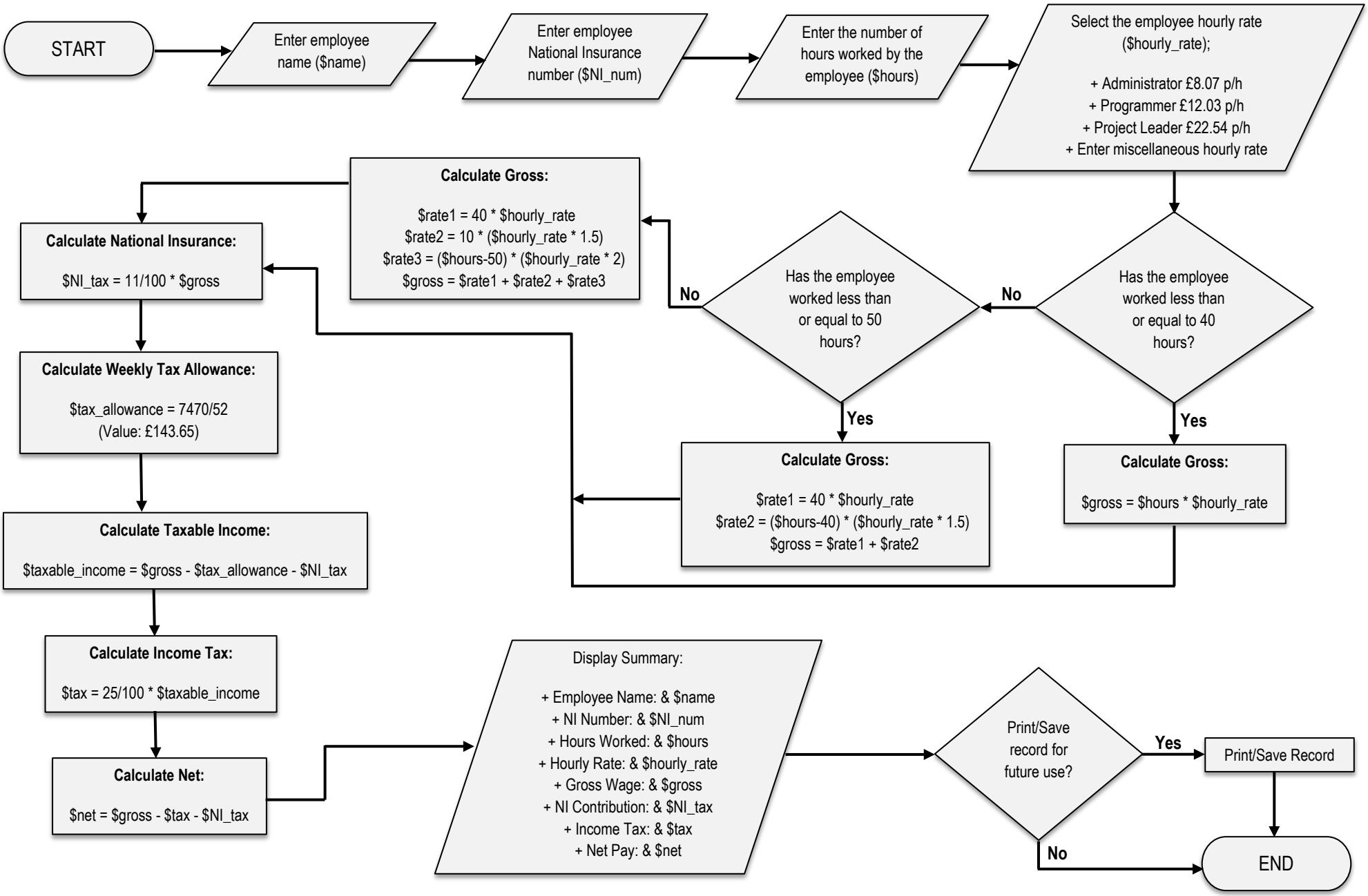
Requirements

- ☑ Write an application that can be used in a small business to calculate the weekly wages (including any overtime) of their employees
- ☑ Any employee who accumulates more than 40 hours per week earns an overtime bonus. The overtime is calculated at 'time and a half' for hours between 41 and 50, and 'double time' for hours over 51
- ☑ The application must be able to display the gross and net total, as well as tax (charged at 25%) and national insurance (charged at 11%) deductions
- ☑ A flow-chart design has been prepared for you; use the diagram to support the building of the application

Extras:

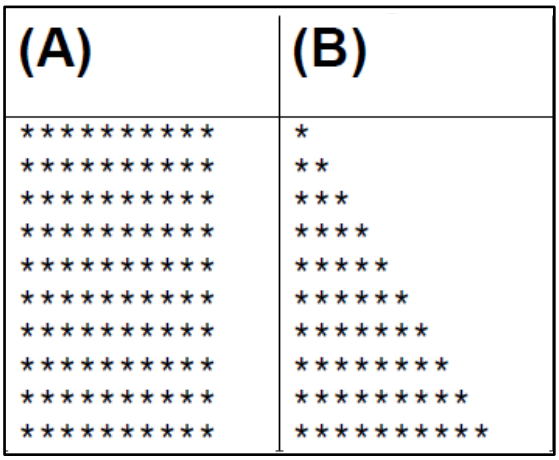
- ☑ Adapt your application so that employees have an annual tax allowance of £7,470 (a tax allowance is a sum of money which employees are allowed to earn before paying taxes)

HINT: divide the tax allowance by 52 to determine the weekly tax allowance



Shapes Program – Simple Version

Create a program that allows the user to choose a shape/pattern (from the list below). Once the user has chosen a shape (either A or B), the application must print the appropriate pattern (with the same dimensions as the shapes below) to the console window.



Code Hints:

To be able to complete this program, the correct use of nested iteration (loops) is required. Use the **For/Next loop** to achieve this. Consider the following code hints before attempting the application:

Use **one loop** with the following code to produce the height of a pattern:

```
For X = 1 To 10
    WriteLine()
    'Insert other code here
Next X
```

Use **another loop, inside** the previous loop, to produce the width of a pattern; for example:

```
'Start of another LOOP
For Y = 1 To 10
    Write("*")
Next Y
'End of another LOOP
```

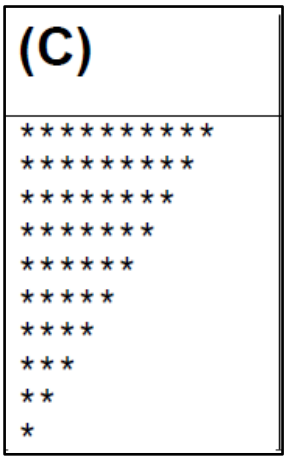
***Note**: 'Write ("*")' will cause the asterisks to be printed side by side as opposed to being printed on a new line.*

Super-Super-Hard Part:

WARNING: Due to the extreme nature and excruciating difficulty of this task, it should only be undertaken by trained professionals and/or superhuman 'whizz-kids'. Please be advised that this this activity poses a serious health risk (risk of dehydration, severe headaches and, in some cases, dizziness) and therefore should only be attempted at your own risk!

Part 1:

Add a third shape/pattern choice (shown below) to your application. If the user chooses this shape, the application must print the appropriate pattern (with the same dimensions as the shape below) to the console window.



Part 2:

Change the application so that once the user has chosen the shape, either A, B or C, they can then specify the size too. The application will then draw the shapes to the specified size, as opposed to drawing them to a fixed size. Use variables (as shown below) to achieve this.

```
        Dim Z As Integer
        Z = ReadLine()
        For X = 1 To Y
            Write("*")
        Next X
```

The Grade Calculator

Requirements

- ☑ Write an application that calculates the end grade for a learner who has completed 18 units on the BTEC National Diploma
- ☑ The application must allow the user to enter their grade (pass, merit, distinction) for each of the 18 units below, before displaying their calculated grade
- ☑ The application must also calculate the number of UCAS points earned by the user's overall grade
- ☑ All of the information needed is displayed below in the appropriate tables:

BTEC National Units

Year 1		
Unit	**Unit Name**	**Credits**
Unit 1	Communication and Employability	10
Unit 2	Computer Systems	10
Unit 3	Information Systems	10
Unit 6	Software Design and Development	10
Unit 9	Computer Networks	10
Unit 11	Systems Analysis and Design	10
Unit 16	Procedural Programming	10
Unit 18	Database Design	10
Unit 30	Digital Graphics	10

Year 2		
Unit	**Unit Name**	**Credits**
Unit 5	Managing Networks	10
Unit 10	Communication Technologies	10
Unit 12	IT Technical Support	10
Unit 13	IT Systems Troubleshooting and Repair	10
Unit 14	Event-Driven Programming	10
Unit 20	Client-Side Customisation of Web Pages	10
Unit 23	Human–Computer Interaction	10
Unit 27	Web Server Scripting	10
Unit 31	Computer Animations	10

BTEC National Points Per Credit Awarded	Pass	Merit	Distinction
Points Per Credit	7	8	9
Points Per Typical 10 Credits	70	80	90

BTEC National Overall Grade (Based on Points Earned):

BTEC Points	Grade
1260–1299	PPP
1300–1339	MPP
1340–1379	MMP
1380–1419	MMM
1420–1459	DMM
1460–1499	DDM
1500–1529	DDD
1530–1559	D*DD
1560–1589	D*D*D
1590 and above	D*D*D*

UCAS Tariff Points Awarded for Extended Diploma (Based on BTEC National Overall Grade):

Grade	UCAS Points
D*D*D*	420
D*D*D	400
D*DD	380
DDD	360
DDM	320
DMM	280
MMM	240
MMP	200
MPP	160
PPP	120

The Mortgage Calculator

Requirements

- ✓ Write an application that can be used in a building society to calculate the interest and repayments of a fixed-mortgage loan

The application **must**:

- ☑ Allow the user to enter the value of the property
- ☑ Allow the user to enter a deposit value (minimum of 5% and maximum of 95% of the property value)
- ☑ Allow the user to enter the term of the mortgage (minimum of 4 years and maximum of 30 years)
- ☑ The application must then calculate the amount of interest to be paid based on the information given (page 2)
- ☑ The application must also summarise the following:
- ☑ The term of the mortgage
- ☑ The value of the property
- ☑ Deposit paid and the percentage of that deposit against the property value
- ☑ The total amount of the money to be loaned (not including interest)
- ☑ The total interest to be paid and the % rate
- ☑ The total amount to be paid back (loan + interest)
- ☑ The value to be paid back for each £1 borrowed (i.e. £1.65 to be paid back for each £1 borrowed)
- ☑ The amount of money to be paid each year and each month (shown with interest) to clear the loan

Interest rates (fixed mortgage)

10-Year Term or Less (Minimum of 4 Years)	
Deposit Percentage:	**Interest Rate:**
50% of Property Value or Less	34.23%
51% Property Value or More	21.10%
11-Year to 20-Year Term	
Deposit Percentage:	**Interest Rate:**
50% of Property Value or Less	41.83%
51% Property Value or More	33.11%
21-Year to 30-Year Term (Maximum of 30 Years)	
Deposit Percentage:	**Interest Rate:**
50% of Property Value or Less	56.92%
51% Property Value or More	42.87%

The E-ticket Machine

Requirements

☑ Create an application that can be installed on an e-ticket machine, inside a train station, to allow users to plan their journey on the 'West Midlands–London Service'

The application should allow the user to pick a start and end destination, the ticket quantity and a ticket type.

The application must then calculate the ticket price based on the following information:

Ticket Prices:

☑ Between 1 and 4 stops it is £1.85 per stop

☑ Between 5 and 6 stops it is £2.60 per stop

☑ Over 7 stops it is £3.10 per stop

Ticket Types:

☑ Single

☑ Return (apply a 13% discount)

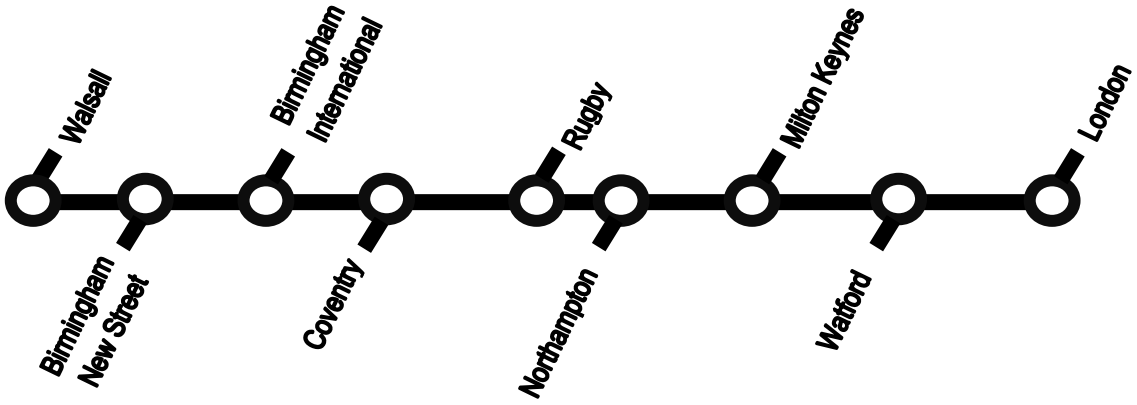
Desirable Additions:

☑ Grant a 25% discount for off-peak tickets (Off-peak is 6pm to 7am)

☑ Grant a 12.5% discount for children

☑ Display the time at the start of the application (i.e. 7:45)

The West Midlands–London



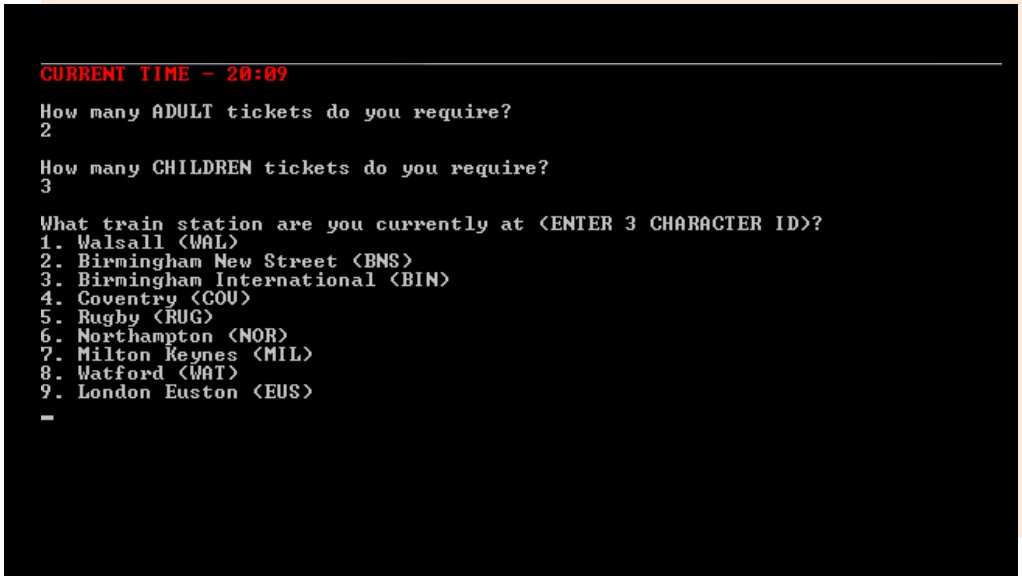
Code Tips

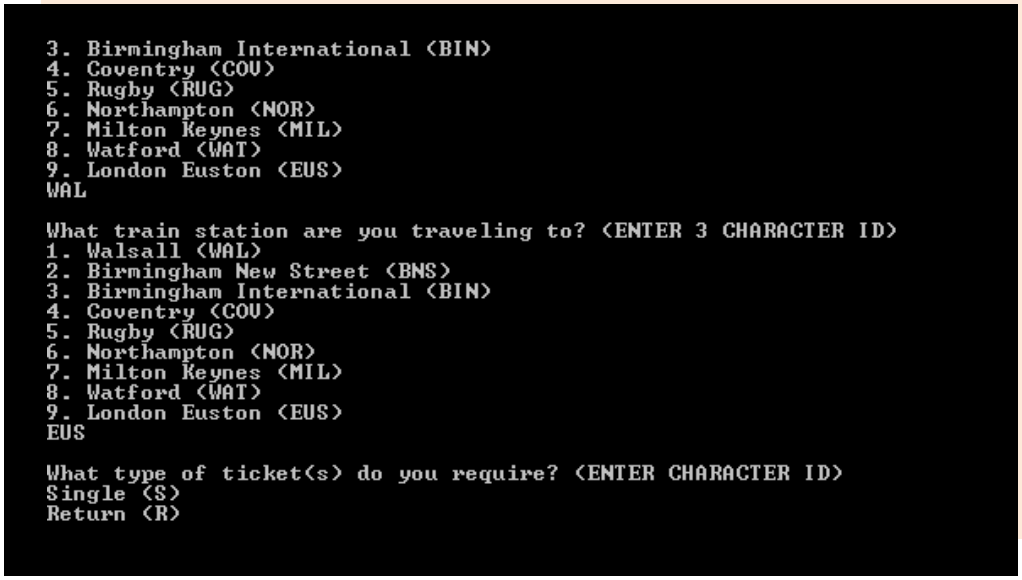
Use this code to display the time:

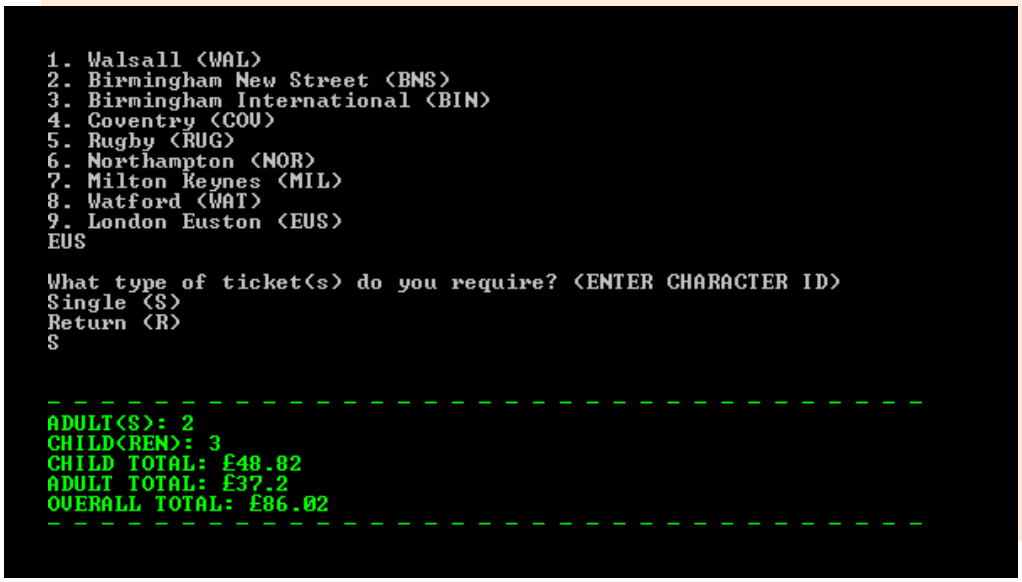
```
Dim rightNow As DateTime = DateTime.Now
Dim Time As String
Time = rightNow.ToString("H:mm")
WriteLine(Time)
```

Use this code to help you calculate the off-peak times:

```
Dim CurrHour As Integer
Dim Currtime As Date
Currtime = DateTime.Now
CurrHour = Currtime.Hour
If CurrHour <= 7 Or CurrHour >= 18 Then
   MsgBox("This is OFF-PEAK")
End If
```







The Pizza Menu

Requirements

☑ Write a pizza application to handle telephone delivery orders for a downtown pizzeria

The application must:

✓ Record the customer's name, address and telephone number

✓ Record details of the pizza order from a set menu (see menu below)

✓ Limit the customer to a maximum of 20 pizzas per order

✓ Summarise the customer's order, including all pizza orders, the price of each pizza, the total cost of the order and details of any delivery charges

Additions:

☑ Use appropriate validation techniques to limit the users' input

Pizza Menu

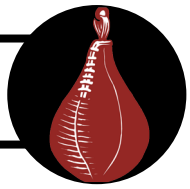
Pizzas are made with tomato and a fresh mozzarella cheese base. All pizzas consist of 16 slices.

1. **Cheese & Tomato** – Italian-style six-cheese blend – £7.50
2. **BBQ Chicken** – Chargrilled chicken, barbeque sauce, bacon, onions – £7.90
3. **Meat Feast** – Ham, peperoni, sausage, bacon, spicy beef – £8.10
4. **Piri-Piri Chicken** – Chilli pepper sauce, chargrilled chicken – £8.80
5. **Hawaii** – Ham, pineapple, mushrooms – £8.90
6. **Mediterranean** – Chorizo, Italian-style sausage, jalapeno sausage – £9.50
7. **The Mexican** – Jalapeno peppers, red peppers, spicy beef, onions – £9.70
8. **The Works** – Pepperoni, sausage, ham, mushrooms, green peppers – £9.90

Special discount: 20% off any order over £20

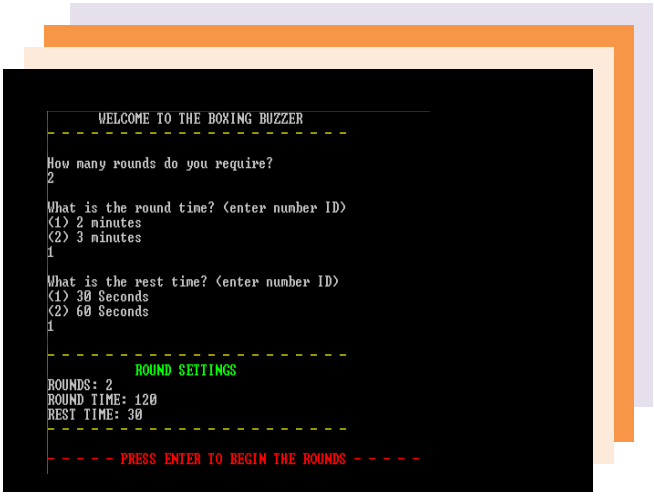
Minimum order is £7 or a delivery charge of £1.50 added

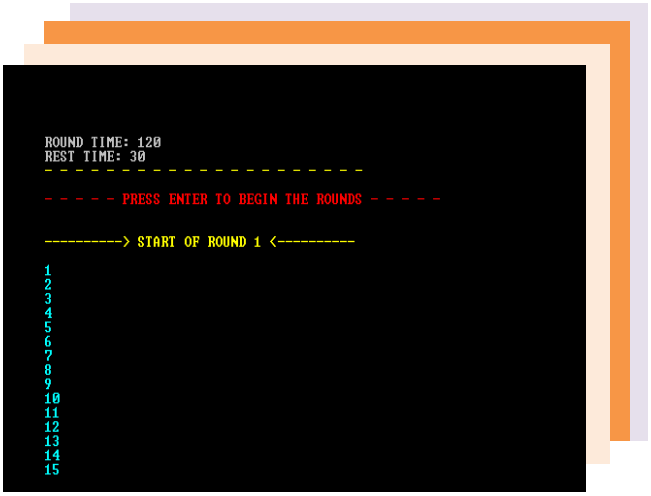
The Boxing Buzzer



Requirements

- ✓ Create an application that can be used in a boxing gym to indicate to boxers (during their training) when rounds begin and end.
- ✓ The application should allow the user to choose the number of rounds, the length of the round (either 2 or 3 minutes) and the rest period in between rounds (either 30 seconds or 1 minute). There should also be a stopwatch to indicate how much time has passed.
- ✓ To complete this activity you will need to use three loops; a For Loop for the rounds, and two While Loops (inside the For Loop) for both the round time and rest time.
- ✓ Desirable Requirements:
- ✓ Use sounds to indicate when rounds start and finish (sounds will be provided for you by your tutor)
- ✓ Summarise the settings before the beginning of the rounds
- ✓ Pause the application (using ReadLine) before the beginning of the rounds





Code Hints:

Use this code to play sounds (pay attention to the directory of the file):

```
My.Computer.Audio.Play("C:\boxing\bell.wav")
```

Use this code to create a stop watch and write it to the screen:

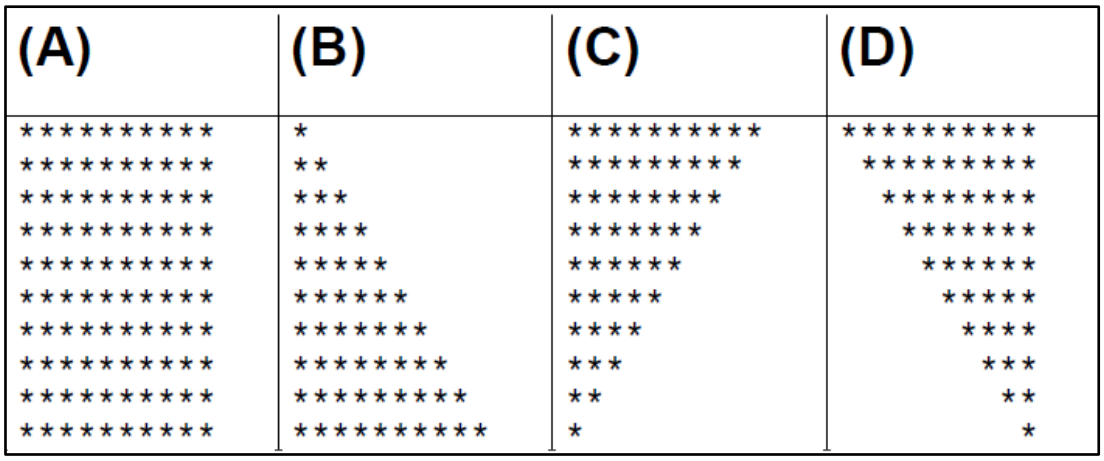
```
Imports system.diagnostics 'add this to the very top of the app
Dim watch As Stopwatch = Stopwatch.StartNew()
Dim X As Integer = 1
watch.Restart()
Do While watch.Elapsed.TotalSeconds < 15
   If watch.Elapsed.TotalSeconds = X Then
        WriteLine(Math.Round(watch.Elapsed.TotalSeconds, 2))
        X = X + 1
   End If
Loop
watch.Stop()
```

Use this code to pause the application until the user presses the Enter key to continue:

```
ReadLine()
```

Shapes Program

Create a program that allows the user to choose a shape/pattern (from the list below). The user must then be able to select the size of the shape (between 0 and 20), before the application prints the pattern (of the chosen size) to the console window.



Code Hints:

To be able to complete this program, the correct use of nested iteration (loops) is required. Use either the **For/Next loop** and/or the **Do While/Until loop** to achieve this. Consider the following code hints before attempting the application:

Use **one loop** with the following code to produce the height of a pattern:

```
For X = 1 To 10
    WriteLine()
    'Insert other code here
Next X
```

Use **another loop, inside** the previous loop, to produce the width of a pattern; for example:

```
'Start of another LOOP
For Y = 1 To 10
    Write("*")
Next Y
'End of another LOOP
```

***Note**: 'Write ("*")' will cause the asterisks to be printed side by side as opposed to being printed on a new line.*

Pattern D requires each line to be started with the appropriate number of blank spaces. This can be achieved using the following code:

```
Write(Space(5))
```

Super-Super-Hard Part:

WARNING: Due to the extreme nature and excruciating difficulty of this task, it should only be undertaken by trained professionals and/or superhuman 'whizz-kids'. Please be advised that this this activity poses a serious health risk (risk of dehydration, severe headaches and in some cases, dizziness) and therefore should only be attempted at your own risk!

Add a fifth shape/pattern choice (shown below) to your application. The user should be able to choose the size of the pattern (3 to 20), before the application prints the pattern (of the chosen size) to the console window.

Note: *the size of this pattern is determined by the number of rows to the middle point; for example, the size of this diamond is* ***6****.*