education
Department:
Education
PROVINCE OF KWAZULU-NATAL

CURRICULUM GRADE 10 -12 DIRECTORATE

NCS (CAPS)

LEARNER SUPPORT DOCUMENT

STEP AHEAD MATERIAL

GRADE 10

TERM 2

INFORMATION TECHNOLOGY

2022

education
Department:
Education
PROVINCE OF KWAZULU-NATAL

PREFACE

This support document serves to assist Information Technology teachers and learners on how to deal with curriculum gaps and learning losses as a result of the impact of COVID – 19 in 2021. It also captures the challenging topics in the Grade 10 work. This document should be used in conjunction with the 2021 Recovery Teaching Plan and the CAPS document. Activities should serve as a guide to assess the topics dealt with in this document. It covers the following:

TERM 2

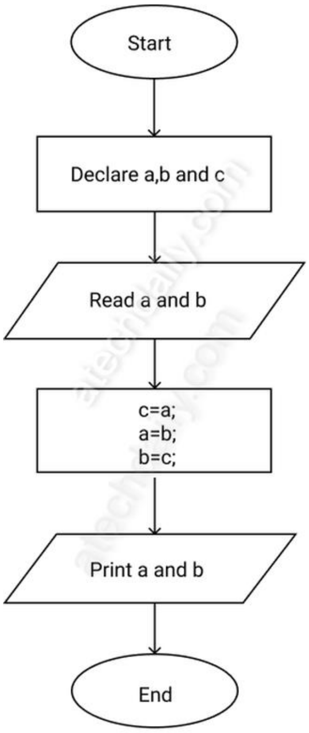
WEEK 1

SOLUTION DEVELOPMENT

Lesson 1

Activity 1: Pseudocode – Swapping values

Use the algorithm below to write pseudocode with appropriate variable names to swap two values



Activity 2: Aggregate Calculations – Sum and Average

The data files are in a folder called Sum and Average in the learner resource folder - Term 2 Week 1.

- Extract the data from the two edit components.
- Write code for the button captioned Sum to calculate the sum of the two numbers extracted. Display the answer using a dialog box.
- Write code for the button captioned Average to calculate the sum of the two numbers extracted. Display the answer using a dialog box.

Lesson 2

Part 1: Isolating Digits

Activity: Isolating digits

Below is an algorithm to isolate the digits of a 2 digit number:

For a 2 digit number use mod 10 and div 10 to isolate the tens and units values.

1. Enter an input (num)
2. x ← num div 10; e.g. 30 div 10 = 3 (isolate first digit)
3. y ← num mod 10; e.g. 30 mod 10 = 0 (isolate last digit)
4. Display x and y

- Write an algorithm to isolate the digits of a 3-digit number using DIV and MOD.

Part 2: Event Handlers

- Watch the video on event handlers and follow the steps to create a simple Delphi project.
- In the OnClick event handler of the button you have placed on the form, write code to display a dialog that says ‘Hello World’.

WEEK 2

SOLUTION DEVELOPMENT

Lesson 1

- Follow the video *Rad Studio Delphi 002 Button - 001 OnClick* to develop the same application following all the steps as indicated in the video.

Lesson 2

Activity: OnActivate Event

The data files are in a folder called Changing Properties in the learner resource folder - Term 2 Week 2.

Write code in the **FormActivate** (OnActivate) event to change the properties of the components.

- Change the **name** of the **Form1** to **frmQuestion1**
- Change the **caption** of **Form1** to **Question 1**
- Change the **color** of **Form1** to **Lime**

- Change the following properties of **pnlHeading**:

a) Caption : **GAME OF THRONES**
b) Font name : **Comic Sans MS**
c) Font Style : **Bold**
d) Font Size : **36**
e) Font color : **Fuchsia**

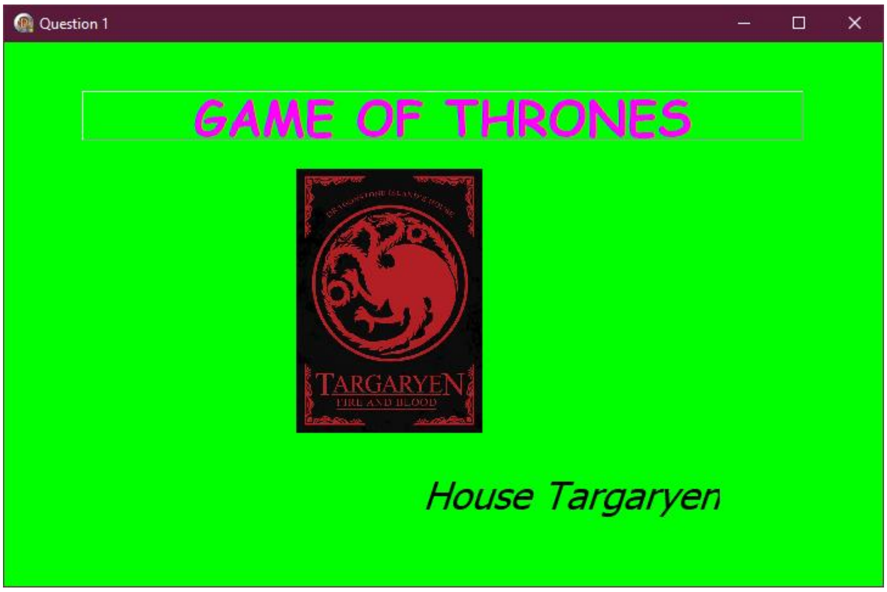
- Change the following properties of the TImage component **imgPicture:**

a) Load the picture file **dragon.jpg**
b) Set Proportional to **true**
c) Set Stretch to **true**

- Change the properties of **lblDisplay** to the following:

a) Caption : **House Targaryen**
b) Font style : **Italic**
c) Font size: **26**

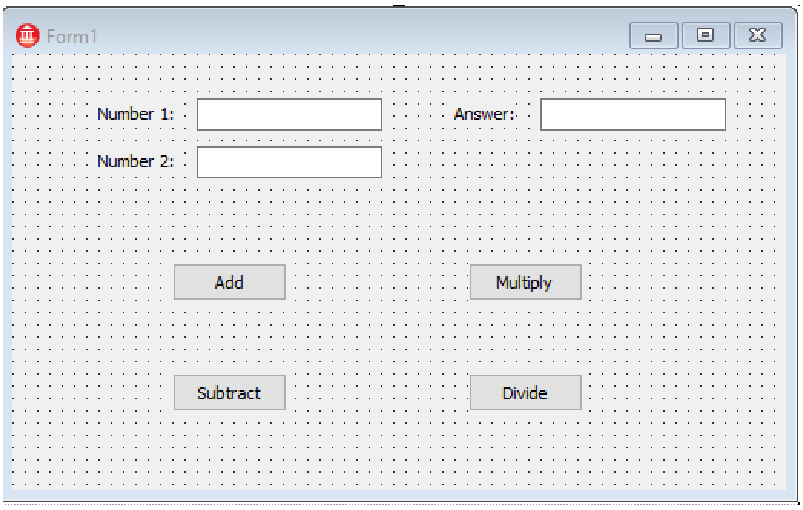
SAMPLE RUN:



Lesson 3

Activity 1: OnClick Event

The data files are in a folder called Simple Calculator in the learner resource folder - Term 2 Week 2.



Write code on the OnClick events for each button to do the following:

1. btnAdd:

- Extract values entered into edtNum1 and edtNum2
- Add the two values
- Display the answer in edtAnswer

2. btnSubtract:

- Extract values entered into edtNum1 and edtNum2
- Subtract the two values
- Display the answer in edtAnswer

3. btnMultiply:

- Extract values entered into edtNum1 and edtNum2
- Multiply the two values
- Display the answer in edtAnswer

4. btnDivide:

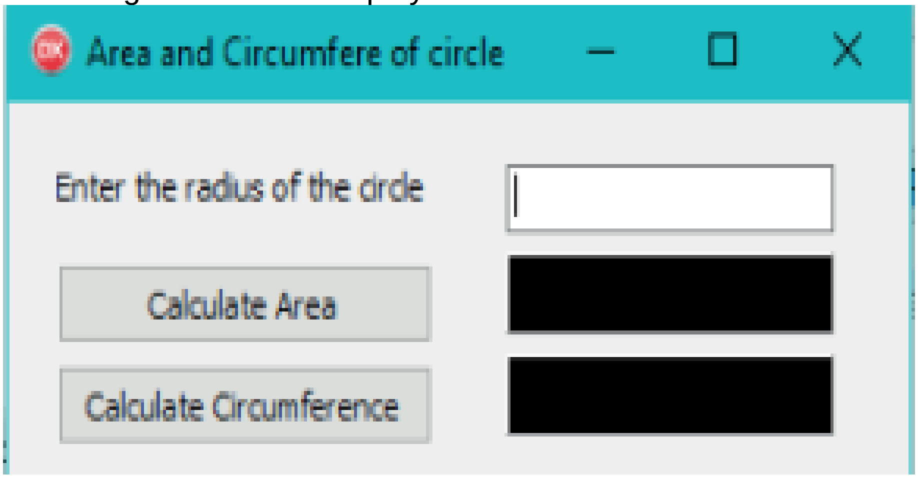
- Extract values entered into edtNum1 and edtNum2
- Divide the two values
- Display the answer in edtAnswer

Activity 2: OnClick Event

The data files are in a folder called '*Circle*' in the learner resource folder - Term 2 Week 2.

Open the Circle_p project located in the Circle data file folder provided.

The design screen will display:



Write code that will calculate and display the radius and circumference in the following buttons with the following formulae:

1. btnArea

- Extract the value entered into edtRadius
- Area := Pi * radius * radius
- Display the area in lblArea

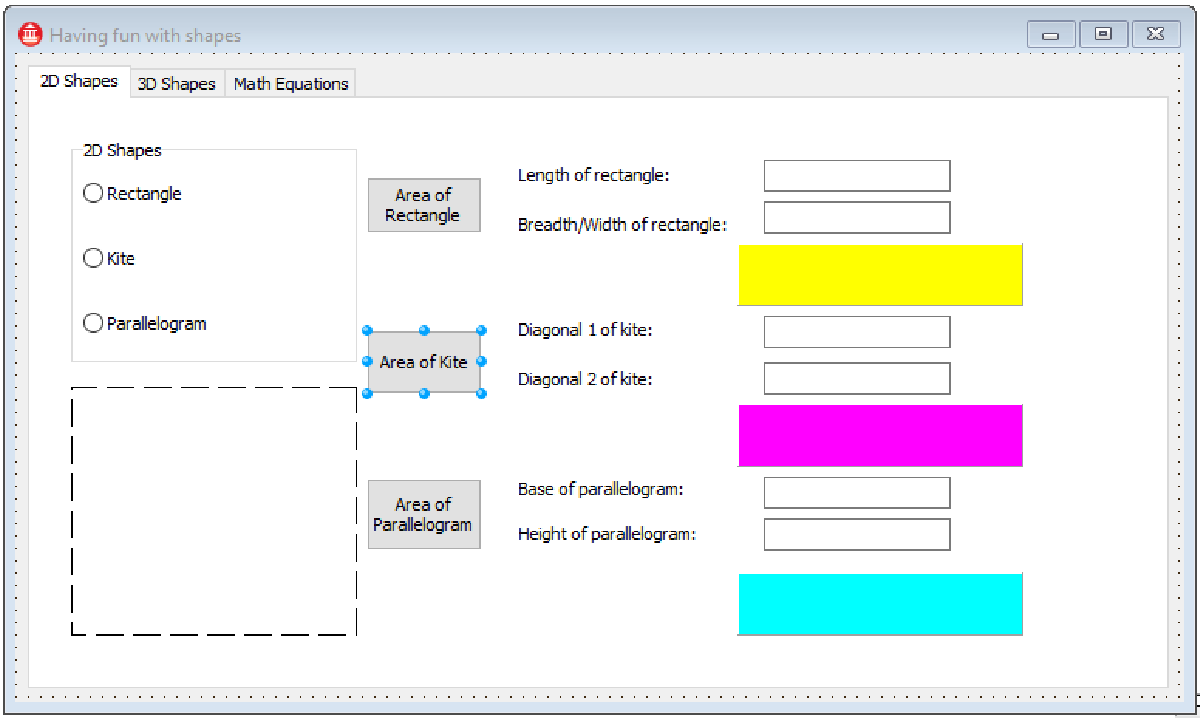
2. btnCircumference

- Extract the value entered into edtRadius
- Circumference := 2 *Pi * radius
- Display the area In lblCircumference

Enrichment Activity: OnActivate and OnClick Events

The data files are in a folder called '*Geometric Shapes*' in the learner resource folder - Term 2 Week 2.

Question 1: Tabsheet [2D Shapes]



Code has been provided to display the images of the 2D shapes that are selected in the radiogroup **rgp2D** using the **OnClick** event. Do not modify code that has been provided.

1.1. Write code for the **OnActivate** event of the form **frmShapes** to do the following:

- Set the stretch property of the TImage **img2D** to true
- Set the proportional property of the TImage **img2D** to true

1.2. Write code for the **OnClick** event of **btnRectArea** to do the following:

- Extract the data entered into **edtLength** and **edtBreadth** and store to relevant data structures
- Calculate the area of the rectangle using the formula:

  Area = Length x Breadth
- Display the area to two decimal places in the panel **pnlRectArea** in the following format:

  Area = ….. square units

1.3. Write code for the **OnClick** event of **btnKite** to do the following:

- Extract the data entered into **edtDiag1** and **edtDiag2** and store to relevant data structures
- Calculate the area of the kite using the formula:

$$\text{Area} = \frac{diagonal\ 1\ x\ diagonal\ 2}{2}$$

- Display the area to two decimal places in the panel **pnlKiteArea** in the following format:

  Area = ….. square units

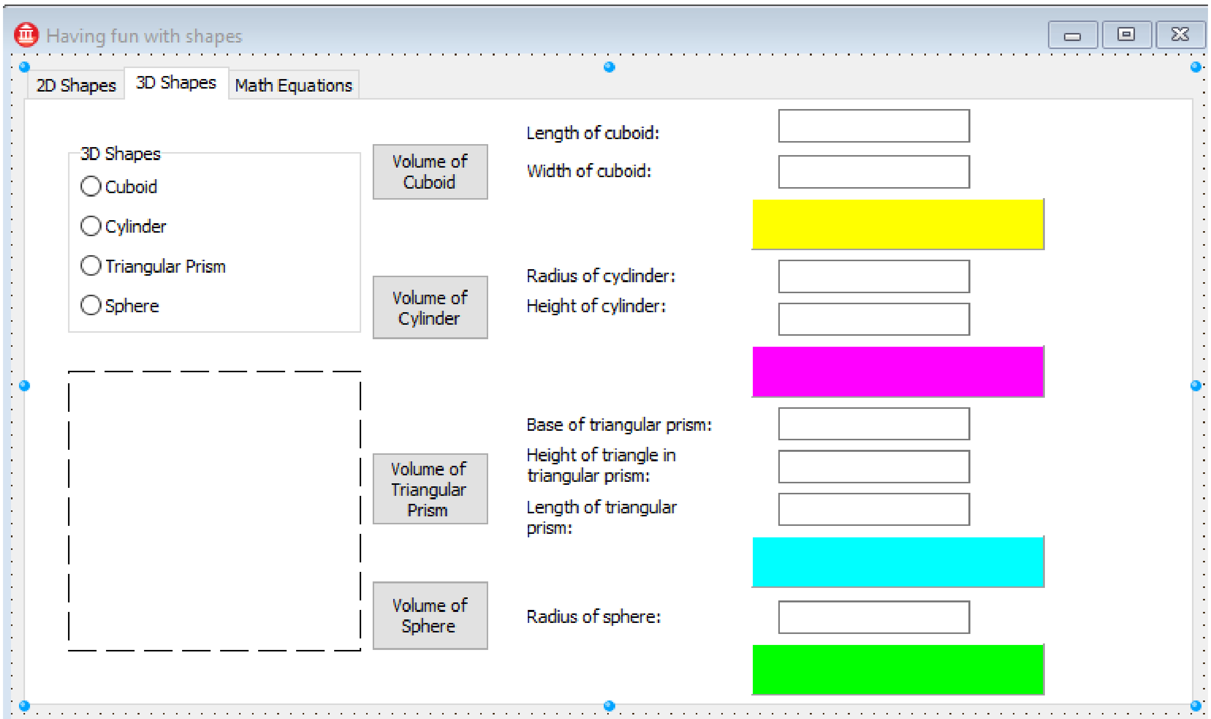
1.4. Write code for the **OnClick** event of **btnParm** to do the following:

- Extract the data entered into **edtBase** and **edtHeight** and store to relevant data structures
- Calculate the area of the kite using the formula:

  Area = base x height
- Display the area to two decimal places in the panel **pnlParmArea** in the following format:

  Area = ….. square units

Question 2: Tabsheet [3D Shapes]



Code has been provided to display the images of the 3D shapes that are selected in the radiogroup **rgp3D** using the **OnClick** event. Do not modify code that has been provided.

2.1. Write code for the **OnActivate** event of the form **frmShapes** to do the following:

- Set the visible property of the TImage **img3D** to true

2.2. Write code for the **OnClick** event of **btnVolCube** to do the following:

- Extract the data entered into **edtCubLength, edtWidth** and **edtCubHeight** and store to relevant data structures
- Calculate the volume of the cuboid using the formula:

  Volume = Length x Width x Height

- Display the volume to one decimal place in the panel **pnlCuboid** in the following format: *Volume = ….. units cubed*

2.3. Write code for the **OnClick** event of **btnVolCyl** to do the following:

- Extract the data entered into **edtRadius** and **edtCylHeight** and store to relevant data structures

- Calculate the volume of the cylinder using the formula:

Volume = π x radius^2 x height

- Display the volume to one decimal place in the panel **pnlCylinder** in the following format:

Volume = ….. units cubed

2.4. Write code for the **OnClick** event of **btnPrism** to do the following:

- Extract the data entered into **edtPrismBase, edtPrismHeight** and **edtPrismLength** and store to relevant data structures
- Calculate the volume of the triangular prism using the formula:

Volume =½ Base x Height x Length

- Display the volume to one decimal place in the panel **pnlPrism** in the following format:

Volume = ….. units cubed

2.5. Write code for the **OnClick** event of **btnSphere** to do the following:

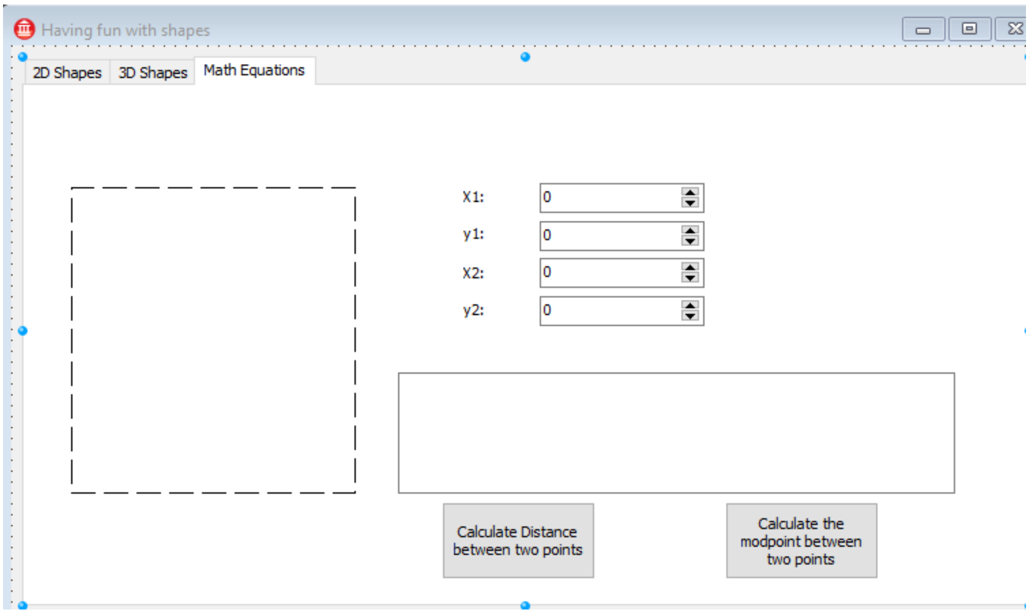
- Extract the data entered into **edtSphereRad** and store to relevant data structures
- Calculate the volume of the sphere using the formula:

Volume = 4/3 π x radius^3

- Display the volume to one decimal place in the panel **pnlSphere** in the following format:

Volume = ….. units cubed

Question 3: Tabsheet [Math Equations]


Having fun with shapes
2D Shapes
3D Shapes
Math Equations
X1:
0
y1:
0
X2:
0
y2:
0
Calculate Distance between two points
Calculate the modpoint between two points

3.1. Write code for the **OnCreate** event of the form **frmShapes** to do the following:

- Load the image called **equations.png** in the TImage component **imgEquation**

3.2. Write code for **btnDistance** to do the following:

- Extract the coordinates (x_1, y_1, x_2, y_2) from the spinedits **spnX1, spnY1, spnX2, spnY2** and store to relevant components.
- Use the distance formula to calculate the distance between the set of coordinates:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

- Display the answer rounded up to a whole number in **memDisplay** in the format:

The distance between the two points are: ……. units

3.3. Write code for **btnMidpoint** to do the following:

- Extract the coordinates (x_1, y_1, x_2, y_2) from the spinedits **spnX1, spnY1, spnX2, spnY2** and store to relevant components.
- Use the moidpoint formula to calculate the distance between the set of coordinates:

$$M = (\frac{x_1 + x_2}{2}; \frac{y_1 + y_2}{2})$$

- Display the answer rounded up to a whole number in **memDisplay** in the format:

The midpoint between the two points are: …….

WEEK 3

CONDITIONALS

Lesson 1

- Complete Activity 5.2 on page 93 from the electronic practical learner textbook.

Lesson 2

- Complete Activity 5.7 on page 103-104 from the electronic practical learner textbook.

Lesson 3

Activity: If-Else Written Exercises

1. Correct the following If statements if they are not correct from a syntax perspective.

1.1. If iNum2 >< iNum1 then iNum1:= iNum1 + 1;

1.2. If iNum1 > iNum2 OR iNum1 < iNum3 then Inc (iNum4);

1.3. If rA > rB > rC then lblOutput.Caption := FloatToStr(rA) + ' is the largest number ';

1.4. If iNum1 => 10 then lblOutput.Caption := 'Yes';

1.5.
```
If (iNum1 > iNum2) then lblOutput.Caption := IntToStr(iNum1) + ' is the largest
number ';
Else
   lblOutput.Caption:= IntToStr(iNum1) + ' is the largest number ';
```

1.6. If chkToppings.Checked := True then rCost = rCost + 10 ;

1.7. If rgpPizzaType = 0 then rCost := 50,

1.8.
```
If iNum1 > iNum2 then
        frmMain.Color := clBlue;
        lblOutput.Caption := IntToStr(iNum1) + ' is the larger number ';
Else
        lblOutput.Caption := IntToStr(iNum2) + ' is the larger number ';
```

1.9. If rgpPizzaType.ItemIndex := -1 then ShowMessage ('No option selected');

2. Write down the just the condition(s) that need to be specified as part of the If statement(s) to accomplish the following tests: Example: *Question: Is iOne a positive number?* Answer: *If iOne* > *0* Then

If Odd(iOne) = False then …………………so much easier ☺

2.1. Is iOne an odd number?
2.2. Is iOne an even number?
2.3. Is iOne a positive odd number?
2.4. Is iOne a positive EVEN number?
2.5. Is rOne a whole number?
2.6. Is 10 a factor of iOne?
2.7. Is 1One a multiple of 5?
2.8. Is iOne a factor of iTwo?
2.9. Is iOne an ODD factor of iTwo?
2.10. Is iOne an EVEN factor of iTwo?
2.11. Is rOne a perfect square?
2.12. Is iOne in the range 1-100?
2.13. Is iOne either negative or more than 100?
2.14. Is sHouse the same sport’s house you are in in the school?
2.15. Is SHouse any one of the sport’s houses in your school?
2.16. Are iOne, iTwo and iThree all the same value?
2.17. Are iOne, iTwo and IThree all unique ('different') values'?
2.18. Is iOne the smallest number of iOne, iTwo and iThree?
2.19. Are these sides (iOne, iTwo and iThree) the sides of an **equilateral** triangle?
2.20. Are these sides (iLongest, iOther1
and iOther2) the sides of a right-angled triangle?
2.21. Are these sides (iOne, iTwo and iThree) the sides of scalene triangle?
2.22. Is the average of three numbers (iOne, iTwo, iThree) is less than -15?
2.23. Is the gradient between 2 points (represented by coordinates iX1,1Y1, iX2, iY2) negative?
2.24. Are any of these marks (iMark1 and iMark2) not a valid mark (1-100)?
2.25. Does the value iGrade represent a valid grade for a high school?

3. Explain for each program segment why the variable iCount1 will never be incremented, irrespective of the values of the other variables used in each case:

3.1 if (iNum1 Mod 2 = 0) OR (Odd(iNum1)) then Inc(iCount2) else Inc(iCount1);

3.2 if Int (Frac (iNum1 / iNum2)) = 0 then Inc(iCount2) else Inc(iCount1);

4. Consider the following Delphi code segment, which displays a grade symbol or a message based on a
student's mark. For a mark from 80 to 100, an A symbol will be displayed, for 70 to 79, a B symbol, for 60 to 69 a C symbol. For any mark below 60 the message 'Grade is below C' will be displayed.

The code is not correct because when marks such as 75 and 85 are entered, the message 'Grade is below C'
is displayed instead of the correct symbol.

```
If iMark >= 80 then edtGrade.Text := 'A' ;
If iMark >= 70 then edtGrade.Text := 'B' ;
If iMark >= 60 then edtGrade.Text := 'C'
else
   edtGrade.Text := 'Grade is below C' ;
```

4.1. Briefly explain why the code produces the wrong output for marks such as 75 and 85.

4.2. Rewrite the code so that the program works correctly for all possible values for iMark from 1 to 100 (inclusive).

5. Write code (not a whole program) for the following. You may assume that

- All components and variable names are 'declared' and follow the usual naming conventions
- You do not have to input the values unless asked to.
- You may use additional variables if need be with showing their declarations
- All output can be done via a label called lblOutput

E.g. Decide if a year (iYear) is a leap year or not (just assume if it is divisible by 4 for now)

```
If Year Mod 4 = 0 then lblOutput.Caption := 'Yes'
Else
    lblOutput.Caption := 'No';
// It actually not that simple but leave it for now ;)
```

5.1 Work out if the fencing around a rectangular plot of land is more than 150 (metres) or not giving *iLength* and *iBreadth.*

5.2 Work out if the area of the netting needed to cover a circular pool is under 100 (m^2) or not given the diameter of the pool (*rDiameter*).

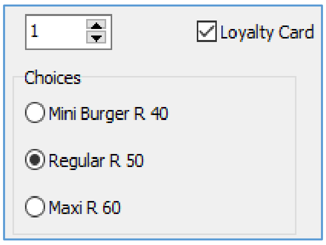
5.3 Work out if the length of fencing needed to go round a cover a circular pool is 400 (m) or more or not given the radius of the pool (*rDiameter*).

5.4 A radio group called **rgpChoices** is used in a program. The program must output the number of the option selected (starting at 1,2,3,4 etc. NOT as 0,1,2,3). In other words if the user chooses the first option then the option number output must be a 1 etc.

The text ‘No choice’ made must be shown if no choice was selected by the user.

5.5 Consider the following GUI of a program used to work out Burger orders – it has the following components:

Name	**Type**	**Purpose**
rgpChoices	Radio group	To choose from 1 of 3 burger types
sedQty	Spin edit	To choose number of burgers
chkBox	Checkbox	To specify whether customer has a loyalty card
memOutput	Memo	Used for outputting results - memOutput.Lines.add(……………..);



1 ☑ Loyalty Card

Choices

○ Mini Burger R 40

● Regular R 50

○ Maxi R 60

Write the following code segments (no variables need be declared)

5.5.1 Clear the memo control

5.5.2 Read the number of pizzas required from the spin edit into a variable called *iOrdered.*

5.5.3 Based on the choice made via the radio group, assign the cost of a single burger of that type to a variable called *rCost4One.*

5.5.4 If the customer orders **Maxi** Burgers then for ever **four** ordered they do not pay for
one of them as follows:

Number of Maxi Burgers ordered	Number free	Number to be paid for
1	0	1
3	0	1
4	1	3
7	1	6
9	2	7
12	3	9

Based on this write code to adjust the value of *iOrdered* value

5.5.5 Use the values of the *rCost4One* and *iOrdered* variables to work out the subtotal and store the value in a variable called *rSubTotal*.

5.5.6 If the customer has a loyalty card then they receive 10% off the price (*rSubTotal*).

5.5.7 Work out the final cost by adding 15% VAT to the subtotal (*rSubTotal*) and store in a variable called *rTotalCost*.

5.5.8 Output the following information to the memo control in the format shown:

Cost of nn burgers = R xxxx.xx

Where nn is the number of burgers (*iOrdered*),

WEEK 4

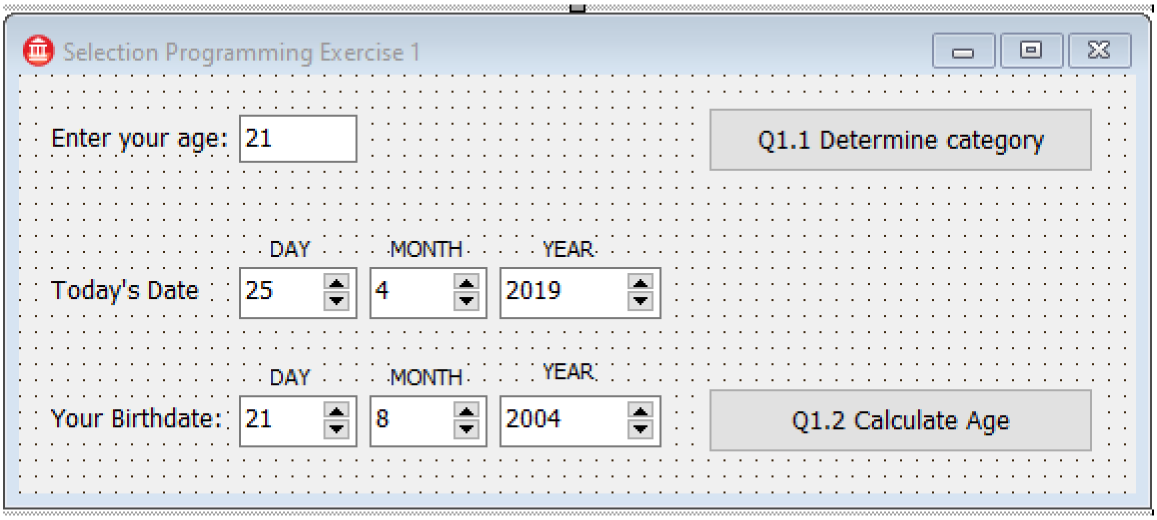
CONDITIONALS

Learners are to make use of the Summary Notes located within learner resource folder for Term 2 Week 4.

Lesson 1

- ➢ Complete Exercise 1 found in the Term 2 Week 4 learner resource folder. The data files for Exercise 1 are found within the folder named: <u>*“Grade 10 IT - Selection Programming Content”*</u>.

GUI: Exercise 1



Enter your age: 21

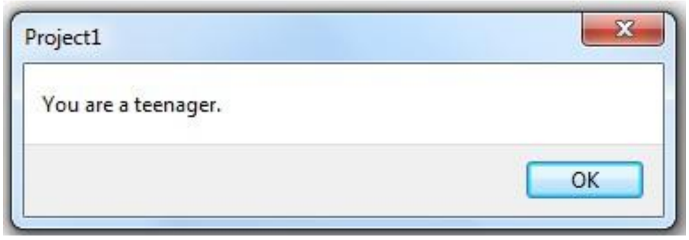
Today's Date: DAY 25, MONTH 4, YEAR 2019

Your Birthdate: DAY 21, MONTH 8, YEAR 2004

1.1 Complete the code for the **btnQ11** button that will get the user’s age from the **edtAge** edit control and then displays a message based on the age given as input:

- ➢ Ages that are 18 or below are for a teenager.
- ➢ Ages between 19 and 39 (inclusive) are for an adult.
- ➢ Ages between 40 and 59 (inclusive) are for middle aged.
- ➢ Ages 60 or above are for a senior citizen.

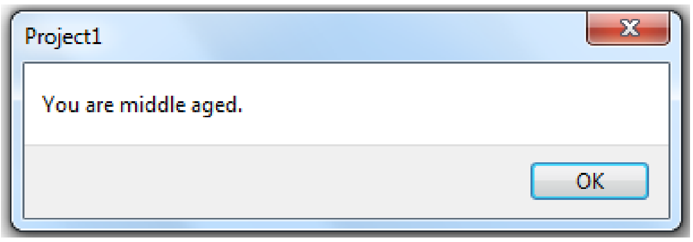
The following outputs are examples of how the message must be displayed:



Project1

You are a teenager.

Or

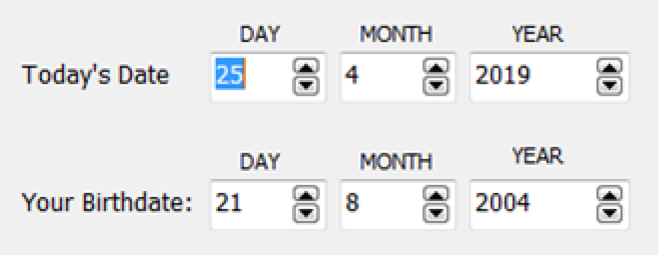


Project1

You are middle aged.

1.2 Complete the code for the **btnQ12** button that will take as input the day, month and year of today's date via three separate spin edit controls AND the day, month and year of your birthdate via three separate spin edit controls.

An example of the graphical user interface is shown below:



	DAY	MONTH	YEAR
Today's Date	25	4	2019
Your Birthdate:	21	8	2004

Use the inputs provided to calculate and display the age of the user. The age of the user is calculated as:

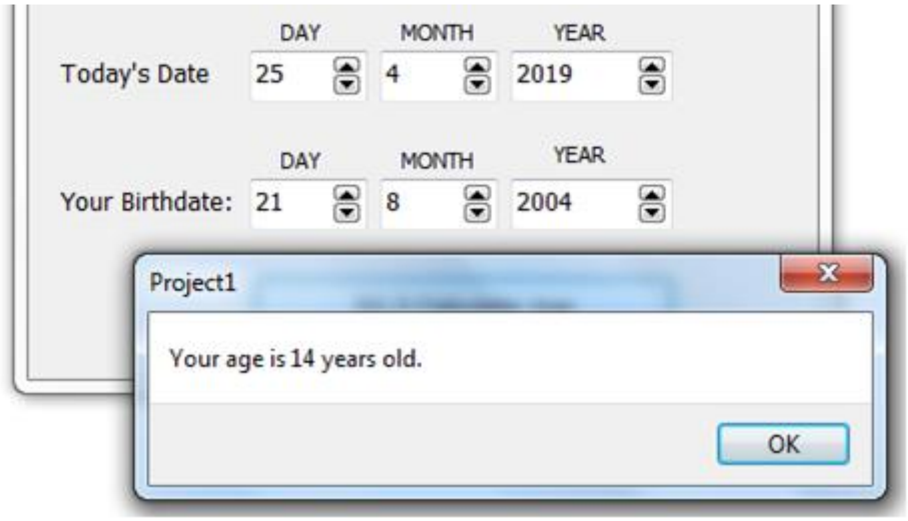
- If your birth month is before today's month, then your age is the difference between today's year and your birth year,
  for example 2019 – 2004 = 15
- If your birth month is after today's month, then your age is the difference between today's year and your birth year MINUS 1 (Because you haven't had your birthday this year yet),
  for example 2019 – 2004 - 1 = 14
- If your birth month and today's month are the same, then

- If your birth day is after today's day then your age is the difference between today's year and your birth year MINUS 1 (Because you haven't had your birthday this year yet),

  for example 2019 – 2004 - 1 = 14
- If your birth day is on or before today's day then your age is the difference between today's year and your birth year,

  for example 2019 – 2004 = 15.

Display the age in a show message procedure as shown in the diagram below:



Today's Date: DAY 25, MONTH 4, YEAR 2019

Your Birthdate: DAY 21, MONTH 8, YEAR 2004

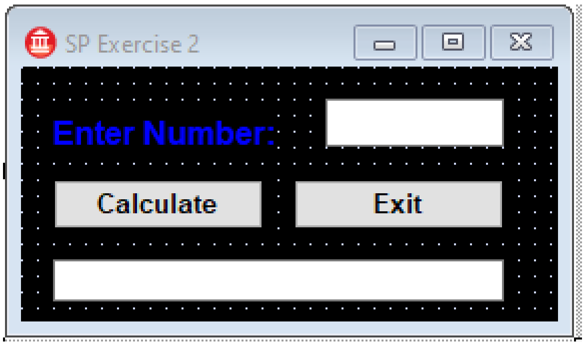
Project1

Your age is 14 years old.

Lesson 2

➢ Complete Exercise 2 and Exercise 3 found in the Term 2 Week 4 learner resource folder. The data files for Exercise 2 and Exercise 3 are within the folder named: *"Grade 10 IT - Selection Programming Content"*.

GUI: Exercise 2



Open the ***Project2*** file and complete the program by answering the following questions:

This program has an edit box ***edtNumber*** that takes as input an integer and another edit box ***edtResult*** that displays a result once the user clicks on the Calculate button, ***btnQ1****.* When the user clicks on this button, the program must determine if the number given as input is an ABC number, an XYZ number or a normal number.

An ABC number is any **EVEN** number that:

- Has 7 as a **FACTOR** and
- When it is divided by 5, it has a **REMINDER** of less than 3.

An XYZ number is any number that:

- Is a **PERFECT SQUARE** and
- Can only be a number from 0 to 50, 100 to 150 or 200 to 250.
- Does **NOT** have 3 as a **FACTOR**.

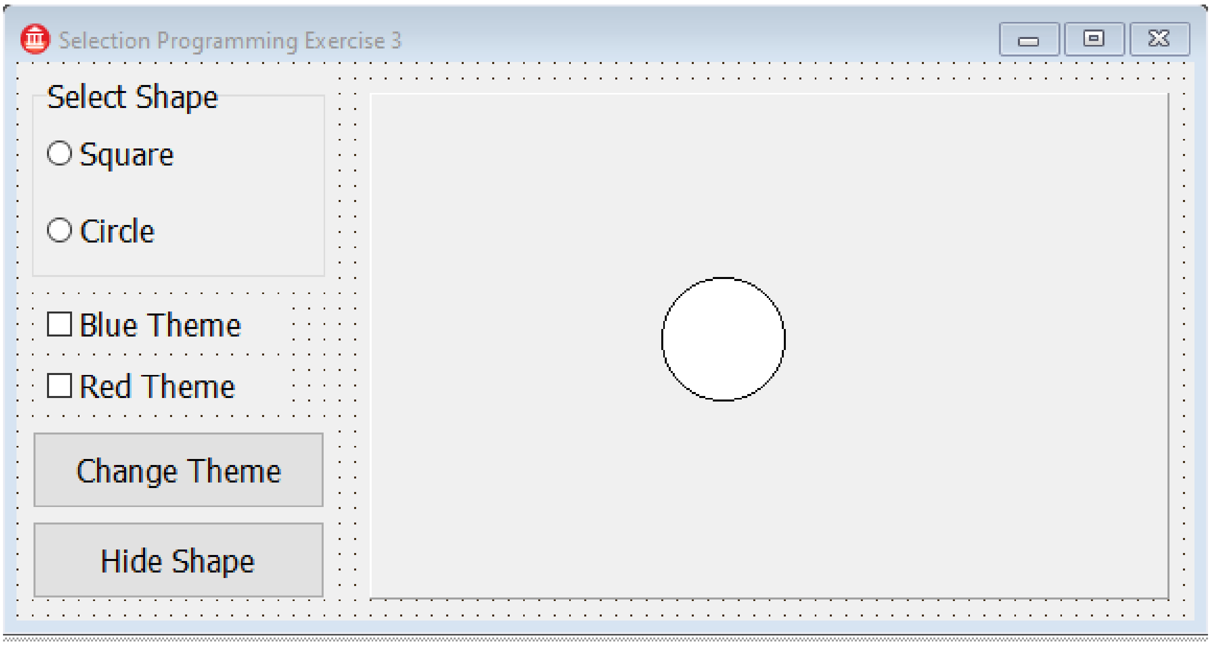
Any other number is a normal number.

Write the code that will calculate what type of number it is and then display the result in the ***edtResult*** edit box.

For example:

- 42, 112, 210 are ABC numbers
- 25, 49, 121 are XYZ numbers
- 3, 144, 168 is a Normal number

GUI: Exercise 3



Open the ***Project1*** file and complete the program by answering the following questions:

1.1. Complete the code for the ***rgpQ11*** radio group's OnClick event handler that will change the shape of the ***shpShape*** shape component to a square shape if the first option in the radio group is selected and to a circle shape if the second option in the radio group is selected.

1.2. Complete the code for the ***btnQ12*** button's OnClick event handler that will change the colour of the ***pnlDisplay*** panel based on which checkboxes are selected based on the following criteria:

- If the ***cbxBlue*** checkbox is selected, then the panel must be changed to *blue*.
- If the ***cbxRed*** checkbox is selected, then the panel must be changed to *red*.
- If BOTH checkboxes are selected, then the panel must be changed to *maroon*.
- If NO checkboxes are selected, then the panel must be changed to *black*.

1.3. Complete the code for the ***btnQ13*** button's OnClick event handler that will do the following:

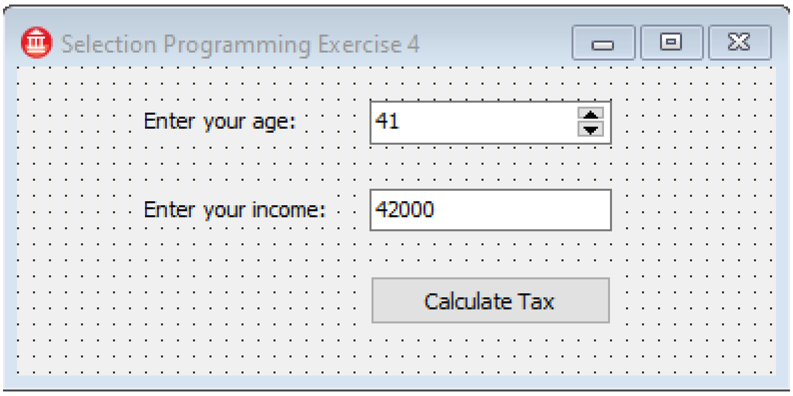
- If the ***shpShape*** shape is visible, then
  - The shape must become invisible AND
  - The ***btnQ13*** button's text must change to '*Show Shape*'

- And if the ***shpShape*** shape is invisible then
    - The shape must become visible AND
    - The ***btnQ13*** button's text must change to '*Hide Shape*'

Lesson 3

- Complete Exercise 4 and Final Challenge found in the Term 2 Week 4 learner resource folder. The data files for Exercise 4 and Final Challenge are found within the folder named: *"Grade 10 IT - Selection Programming Content"*.

GUI: Exercise 4



Selection Programming Exercise 4

Enter your age: 41

Enter your income: 42000

Open the ***Project1*** file and complete the program by answering the following questions:

This program has a spin edit ***spnAge*** that takes as input the user's age and an edit box ***edtIncome*** that takes as input the user's monthly income. Once the user clicks on the Calculate Tax button, ***btnQ1,*** the program must determine the tax amount needed to be paid based on the following table:

<table>
<tr><th rowspan="2">Ages</th><th colspan="5">Monthly Income Brackets</th></tr>
<tr><th>0</th><th>13000</th><th>18 000</th><th>40 000</th><th>80 000 -></th></tr>
<tr><td>66 to 100</td><td>5%</td><td colspan="2">16%</td><td colspan="2">32%</td></tr>
<tr><td>31 to 65</td><td colspan="2">10%</td><td>15%</td><td>28%</td><td>40%</td></tr>
<tr><td>19 to 30</td><td>0</td><td colspan="4">12% on ONLY everything over R13 000</td></tr>
<tr><td>Below 18</td><td colspan="5">0</td></tr>
</table>

Make use of at least ONE case statement for this example.

For example:

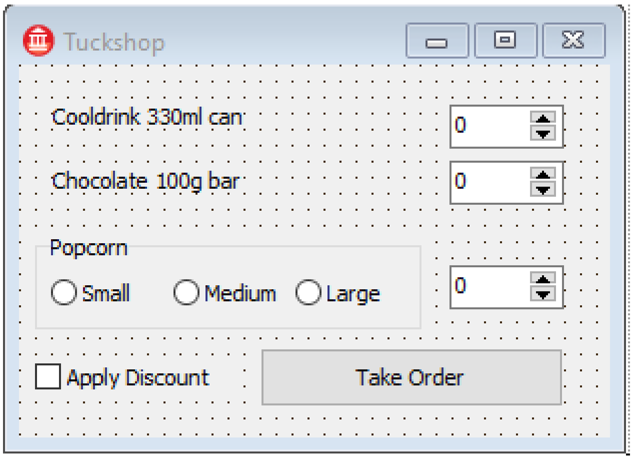
- **21** year old earning **R20 000** pays 12% on everything over R13 000, therefore 12% of R7 000 = **R840.00**
- **35** year old earning **R20 000** pays 15% on their income, therefore 15% of R20 000 = **R3 000.00**
- **50** year old earning **R50 000** pays 28% on their income, therefore 28% of R50 000 = **R14 000.00**
- **70** year old earning **R15 000** pays 16% on their income, therefore 16% of R15 000 = **R2 400.00**
- **73** year old earning **R42 000** pays 32% on their income, therefore 32% of R42 000 = **R13 440.00**

Use a ***ShowMessage*** procedure to display the answer in the following format:

Tax Amount = R XXXXX.XX

(Amount is displayed as a currency to TWO decimal places).

GUI: Final Challenge



Open the ***Project1*** file and complete the program by answering the following questions:

1. Complete the code for the ***cbxDiscount*** checkbox's OnClick event handler (when the user clicks on the checkbox) that will do the following:

- If the ***cbxDiscount*** checkbox is ticked, then the user will be prompted (Inputbox) for a number security code and the program must determine if the code is valid. The code is valid if it meets ALL of the following requirements.
  - The code is a FOUR DIGIT number.
  - The code is an EVEN number.
  - The code has THREE or SEVEN as a factor.

1008, 1368 and 9996 are all examples of valid codes.

If the user gives an invalid code then an error message must be displayed (showmessage) and the ***cbxDiscount*** checkbox must remain <u>unchecked.</u>

2. Complete the code for the ***btnQ2*** button's OnClick event handler for the following questions:

2.1. Calculate the total amount for the cooldrinks. This is calculated by using the quantity (number) of cooldrinks ordered given by the user via the ***spnCooldrink*** spin edit control.

Cooldrinks cost R10 each unless you order 6 or more, then they cost R8.99 each.

Example:

- If 4 cooldrinks are ordered, then the total amount for cooldrinks is 4 x R10 = R40
- If 7 cooldrinks are ordered, then the total amount for cooldrinks is 7 x R8.99 = R62.93

Store this answer in the ***rCoolDrinkTotal*** real variable.

2.2. Calculate the total amount for the chocolates. This is calculated by using the quantity (number) of chocolates ordered given by the user via the ***spnChoc*** spin edit control.

Chocolates costs R6.99 each but every THIRD chocolate bought is free.

Example:

- If 2 chocolates are ordered, then the total amount for chocolates is: 2 x R6.99 = R13.98
- If 3 chocolates are ordered, then that means that one is free and you only pay for the first two so the total amount for chocolates is: (3 x R6.99) - (1 x R6.99) = R13.98
- If 11 chocolates are ordered, then that means the third, sixth and ninth chocolates are free so the total amount for chocolates is: (11 x R6.99) - (3 x R6.99) = R55.92

Store this answer in the ***rChocTotal*** real variable.
HINT: Use DIV to determine every THIRD chocolate

2.3. Calculate the total amount for the popcorns. This is calculated by using the quantity (number) of popcorns ordered by the user via the ***spnPopcorn*** spin edit control and the size of the popcorns via the ***rgpPopcorn*** radio group.

Small popcorns cost R9.99 each, medium popcorns cost R19.99 each and large popcorns cost R25.99 each.

Example:

- If 2 small popcorns are ordered, then the total amount for popcorns is: 2 x R9.99 = R19.98
- If 3 medium popcorns are ordered, then the total amount for popcorns is: 3 x R19.99 = R59.97

- If 4 large popcorns are ordered, then the total amount for popcorns is: 4 x R25.99 = R103.96

  Store this answer in the ***rPopcornTotal*** real variable AS WELL AS the name of the size of the popcorn (Small, Medium or Large) in the **sPopcornType** string variable. *(If no popcorn is selected then 'None' must be stored in the string variable)*

2.4. Calculate the total amount for the order, which is calculated as the sum of the cooldrink total, chocolate total and popcorn totals. Store this answer in the ***rTotal*** real variable.

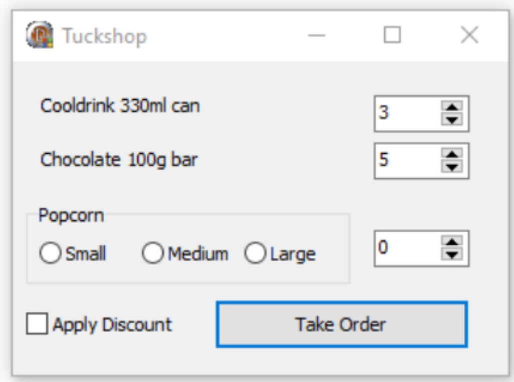
2.5. If the ***cbxDiscount*** check box is selected, the order receives a 10% discount. Display all the details from Questions 2.1 to 2.4 using a showmessage in the following format:

```
format:    Cooldrinks: Rxx.xx
           Chocolate: Rxx.xx
           Popcorn (Type): Rxx.xx
           ORDER TOTAL: Rxx.xx
```

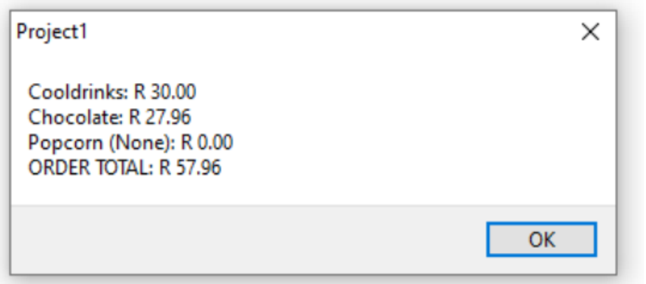
All values must be displayed as currency to TWO decimal places. The words '*with discount*' must appear at the end of the order total if the user received the 10% discount.

HINT: Use #13 to start a new line in a showmessage
Example: showmessage('First line' + #13 + 'Second line' + #13 + 'Third line') ;

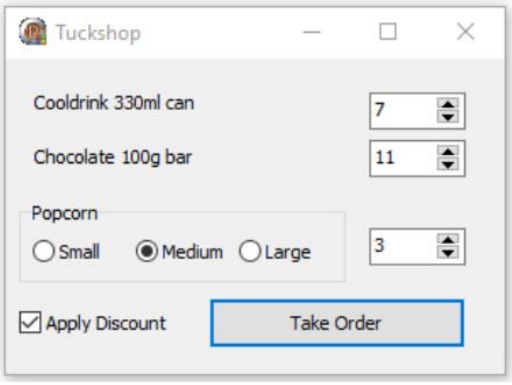
Examples of output:



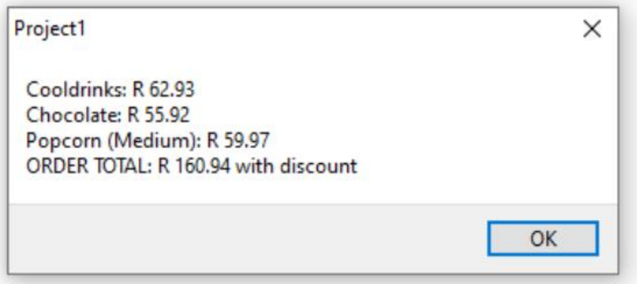
Cooldrink 330ml can: 3
Chocolate 100g bar: 5
Popcorn: ○ Small ○ Medium ○ Large: 0
□ Apply Discount



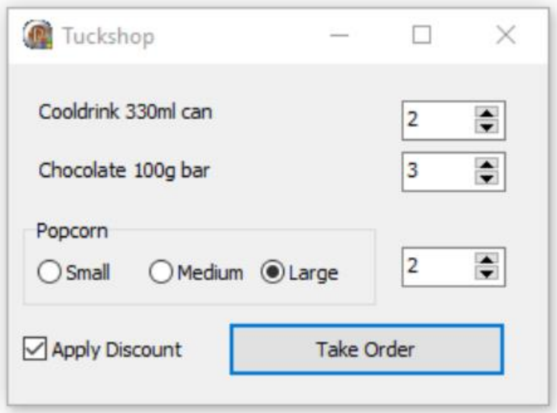
```
Cooldrinks: R 30.00
Chocolate: R 27.96
Popcorn (None): R 0.00
ORDER TOTAL: R 57.96
```



Cooldrink 330ml can: 7
Chocolate 100g bar: 11
Popcorn: ○ Small ● Medium ○ Large: 3
☑ Apply Discount



```
Cooldrinks: R 62.93
Chocolate: R 55.92
Popcorn (Medium): R 59.97
ORDER TOTAL: R 160.94 with discount
```

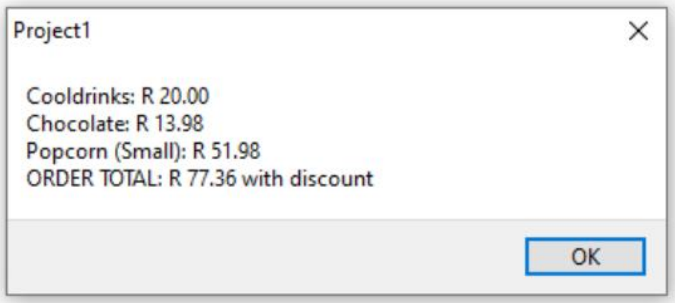


Cooldrink 330ml can 2

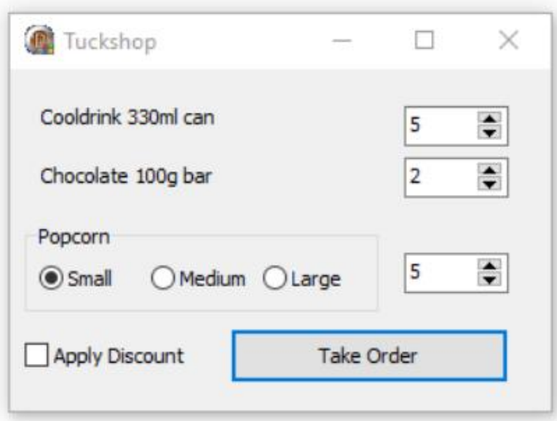
Chocolate 100g bar 3

Popcorn ○ Small ○ Medium ● Large 2

☑ Apply Discount



Cooldrinks: R 20.00
Chocolate: R 13.98
Popcorn (Small): R 51.98
ORDER TOTAL: R 77.36 with discount

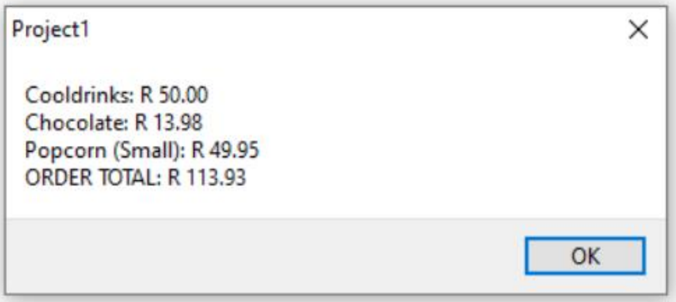


Cooldrink 330ml can 5

Chocolate 100g bar 2

Popcorn ● Small ○ Medium ○ Large 5

□ Apply Discount



Cooldrinks: R 50.00
Chocolate: R 13.98
Popcorn (Small): R 49.95
ORDER TOTAL: R 113.93

WEEK 5

HARDWARE

Lesson 1

- Complete Activity 3.1 on page 68 from the electronic theory learner textbook.
- Complete Activity 3.2 on page 74 from the electronic theory learner textbook.

Lesson 2

- Complete Activity 3.3 on page 78 from the electronic theory learner textbook.
- Complete Activity 3.4 on page 81 from the electronic theory learner textbook.

Lesson 3

- Complete Activity 4.2 on page 92 from the electronic theory learner textbook.
- Complete Activity 4.3 on page 95 from the electronic theory learner textbook.

Lesson 4

- Complete Consolidation Activity on page 96 from the electronic theory learner textbook.

WEEK 6

STRINGS

Resources: Mr Long videos –String Handling

https://www.youtube.com/watch?v=YNCR7RT4t2g&list=PLxAS51iVMjv-0sMKwn0DzTEM6L3HcFL5y

Lesson 1

Activity: Length Function

Given the following code segment, complete the table:

```
Var
        sName, sSurname : string;
                cLetter : char;
Begin
                sName := ‘Christopher’;
                sSurname := ‘Zwane’;
        End;
```

	Statement	Output
1	lblOutput.Caption := IntToStr(length(sName));	
2	lblOutput.Caption := IntToStr (length(sName+ sSurname));	
3	lblOutput.Caption := IntToStr (length(sName+ #9 + sSurname));	
4	lblOutput.Caption := IntToStr (length(sName[1]) + Length(sSurname));	

Lesson 2

Activity: Copy Function

1. Given the following code segment, complete the table:

```
Var
        sName, sSurname, sID : string;
        cLetter : char;
Begin
        sName := ‘Itumeleng’;
        sSurname := ‘Khune’
        sID := ‘9703225212085’
End;
```

	Statement	Output
1	lblOutput.Caption := Copy(sName,5,1)	
2	lblOutput.Caption := Copy(sName,1,0)	
3	lblOutput.Caption := Copy(sName,20,2)	
4	lblOutput.Caption := Copy(sName,6,4)	
5	lblOutput.Caption := Copy(sName,1,1) + ‘ ‘ + Copy(sSurname,1,5)	
6	lblOutput.Caption := Copy(sSurname,1,Length(sSurname))	
7	lblOutput.Caption := ‘Day: ‘ + Copy(sID,5,2)	
8	lblOutput.Caption := ‘Year: 19‘ + Copy(sID,1,2)	
9	lblOutput.Caption := IntToStr(length(Copy(sName,3,5)));	
10	lblOutput.Caption := IntToStr(length(Copy(sName,1,8)));	

2. Given the following code segment, complete the table:

```
Var
        sName, sSurname: string;
        iPos, iAge : integer;
Begin
        sPhrase := ‘Beyoncé,Knowles,30’;
End;
```

	Statement	Output

1	iPos := 8
	sName := Copy(sPhrase, 1, iPos-**1**);
	lblOutput.Caption := sName;	

2	sSurname := Copy(sPhrase, 1, iPos);
	lblOutput.Caption := sSurname;	

3	sSurname := Copy(sPhrase, 5);
	lblOutput.Caption := sSurname;	

Lesson 3

Activity: Pos Function

Given the following code segment, complete the table:

```
var
        sName, sSub: string;
                cLetter : char;
Begin
                sName := ‘Nelson Mandela’;
                sSub := ‘Man’;
                cLetter := ‘N’;
End;
```

	Statement	output
1	lblOutput.Caption := IntToStr(pos(‘n’, sName));	
2	lblOutput.Caption := IntToStr(pos(cLetter, sName));	
3	lblOutput.Caption := IntToStr(pos(sSub, sName));	
4	lblOutput.Caption := IntToStr(pos(‘man’, sName));	
5	lblOutput.Caption := IntToStr(pos(sName, sSub));	

Lesson 4

Activity 1: RightStr Function

Given the following code segment, complete the table:

```
        Var
        sName, sSurname: string;
        iPos, iAge : integer;
Begin
        sPhrase := ‘Beyoncé,Knowles,30’;
End;
```

	Statement	Output

1	sName := Rightstr(sPhrase, 5);
	lblOutput.Caption := sName;	

2	sSurname := Rightstr(sPhrase, 1);
	lblOutput.Caption := sSurname;	

3	sSurname := Rightstr(sPhrase, 0);
	lblOutput.Caption := sSurname;	

Activity 2: LeftStr Function

Given the following code segment, complete the table:

```
var
        sName, sSurname: string;
        iPos, iAge : integer;
Begin
        sPhrase := ‘Beyoncé,Knowles,30’;
End;
```

	Statement	Output

1	sName := leftstr(sPhrase, 5);
	lblOutput.Caption := sName;	

2	sSurname := lefttstr(sPhrase, 1);
	lblOutput.Caption := sSurname;	

3	sSurname := leftstr(sPhrase, 0);
	lblOutput.Caption := sSurname;	

WEEK 7

STRINGS

Lesson 1

Activity: String handling

1. Given the following code segment, complete the table:

```
Var
        sName, sSurname : string;
                cLetter : char;
Begin
                sName := ‘Jean’;
                sSurname := ‘de Viliers’
End;
```

	Statement	Result stored in cLetter
1	cLetter :=sName[1];	
2	cLetter:= sName[5];	
3	cLetter:= sSurname[3];	

2. Given the following code segment, complete the table:

```
Var
        sName, sSurname, sJoin : string;
Begin
                sName := ‘Jean’;
                sSurname := ‘de Viliers’
End;
```

	Statement	Result stored in sJoin
1	sJoin:=sName[1] + ‘ ‘ + sSurname;	
2	sJoin:= sName[1] + sName[3] + sSurname;	
3	sJoin:= sName[1] + sSurname[1] + sSurname[4];	

3. Given the following code segment, complete the table:

```
Var
        sWord1, sWord2, sWord3, sWord4, sWord5: string;
Begin
                sWord1 := ‘String’;
                sWord2 := ‘Manipulation’;
                sWord3 := ‘string’;
                sWord4 := ‘Manipulate’;
                sWord5 := ‘String Manipulation ’;
End;
```

	Statement	True or False
1	sWord1 = sWord3	
2	sWord1 > sWord3	
3	sWord2 > sWord4	
4	Uppercase(sWord1) > sWord1	
5	Uppercase(sWord1) <> Uppercase(sWord3)	
6	Upcase(sWord1[3]) = Upcase(sWord2[4])	
7	sWord5 = sWord1 + ‘ ‘ + sWord2	

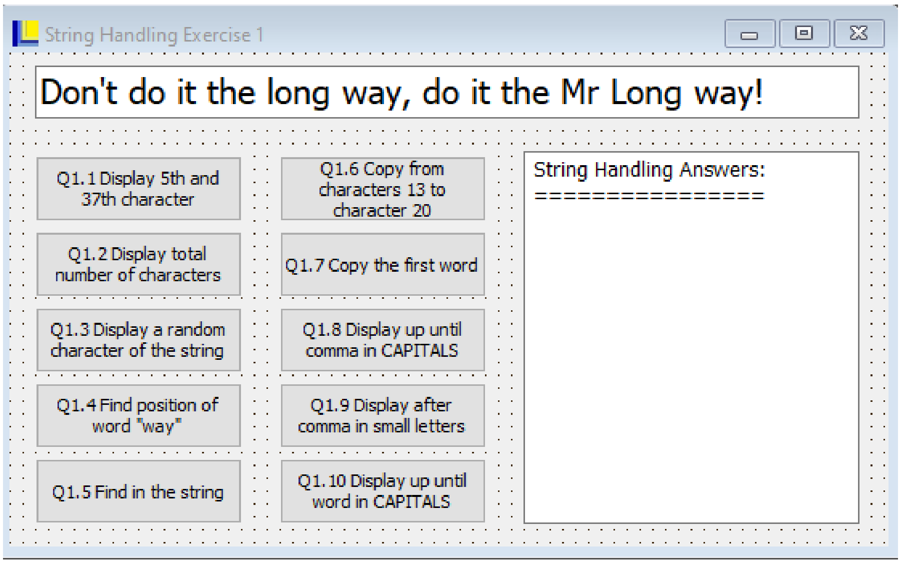
Lesson 2

- Complete the guided activity on page 230 from the electronic practical learner textbook.
- Complete Activity 8.9 on page 235-236 from the electronic practical learner textbook. The data files for this activity is available in the learner resource folder for Term 2 Week 7.

Lesson 3

- Complete String Handling Exercise 1 found in the Term 2 Week 7 learner resource folder. The data files for String Handling Exercise 1 are found within the same resource folder.

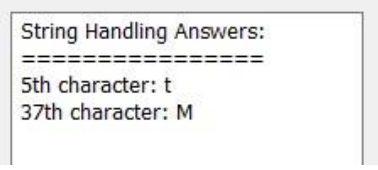
GUI: Exercise 1


String Handling Exercise 1
Q1.1 Display 5th and 37th character
Q1.2 Display total number of characters
Q1.3 Display a random character of the string
Q1.4 Find position of word "way"
Q1.5 Find in the string
Q1.6 Copy from characters 13 to character 20
Q1.7 Copy the first word
Q1.8 Display up until comma in CAPITALS
Q1.9 Display after comma in small letters
Q1.10 Display up until word in CAPITALS

Don't do it the long way, do it the Mr Long way!

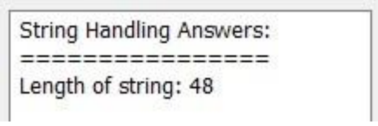
```
String Handling Answers:
================
```

1.1 Complete the code for the ***btnQ11*** button that will display the 5th and 37th character, of the sentence in the ***edtData*** edit control, in the ***memDisplay*** memo control.



```
String Handling Answers:
================
5th character: t
37th character: M
```

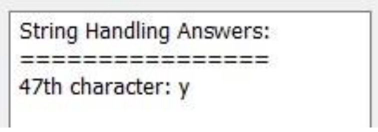
1.2 Complete the code for the ***btnQ12*** button that display the number of characters in the sentence in the ***edtData*** edit control, in the ***memDisplay*** memo control.



```
String Handling Answers:
================
Length of string: 48
```

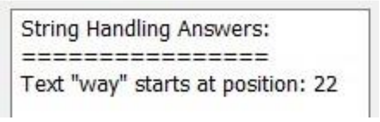
1.3 Complete the code for the ***btnQ13*** button that will display a random character, of the sentence in the ***edtData*** edit control, in the ***memDisplay*** memo control.

NOTE: Your results may be different to example because of the random character:



```
String Handling Answers:
================
47th character: y
```

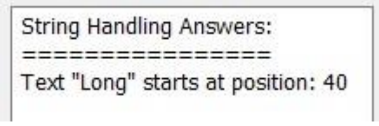
1.4 Complete the code for the ***btnQ14*** button that will display the first occurrence of the word "*way*" in the sentence from the ***edtData*** edit control, in the ***memDisplay*** memo control.



```
String Handling Answers:
================
Text "way" starts at position: 22
```

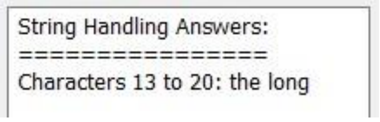
1.5 Complete the code for the ***btnQ15*** button that will prompt the user for a word and then display the first occurrence of the inputted word in the sentence from the ***edtData*** edit control, in the ***memDisplay*** memo control.

Example if the word "Long" is given as input:



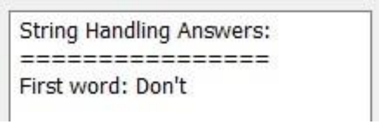
```
String Handling Answers:
================
Text "Long" starts at position: 40
```

1.6 Complete the code for the ***btnQ16*** button that will display characters 13 to 20, of the sentence in the ***edtData*** edit control, in the ***memDisplay*** memo control.



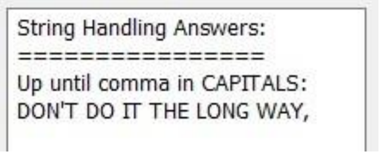
```
String Handling Answers:
================
Characters 13 to 20: the long
```

1.7 Complete the code for the ***btnQ17*** button that display the first word of the sentence in the ***edtData*** edit control, in the ***memDisplay*** memo control. This code must work for any sentence.



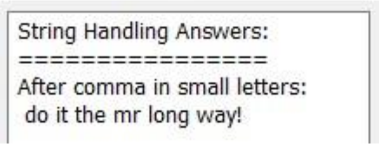
```
String Handling Answers:
================
First word: Don't
```

1.8 Complete the code for the ***btnQ18*** button that will display all the characters from the first character until the first occurrence of the comma symbol in the sentence in the ***edtData*** edit control in capital letters, in the ***memDisplay*** memo control.



```
String Handling Answers:
================
Up until comma in CAPITALS:
DON'T DO IT THE LONG WAY,
```

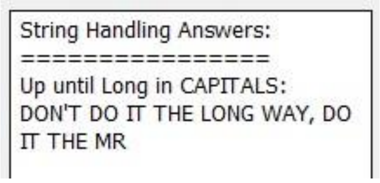
1.9 Complete the code for the ***btnQ19*** button that will display all the characters from after the first occurrence of the comma symbol until the end of the sentence of the text in the ***edtData*** edit control in small letters, in the ***memDisplay*** memo control.



```
String Handling Answers:
================
After comma in small letters:
 do it the mr long way!
```

1.10 Complete the code for the ***btnQ110*** button that will prompt the user for a word and then display all the characters from the first character until the first occurrence of the inputted word in the sentence from the ***edtData*** edit control, in the ***memDisplay*** memo control. Display the characters in capital letters

Example if the word "Long" is given as input:



```
String Handling Answers:
================
Up until Long in CAPITALS:
DON'T DO IT THE LONG WAY, DO
IT THE MR
```

Lesson 4

- Complete Activity 8.10 on page 235-237 of the electronic practical learner textbook. The data files for this activity is available in the learner resource folder for Term 2 Week 7.

WEEK 8

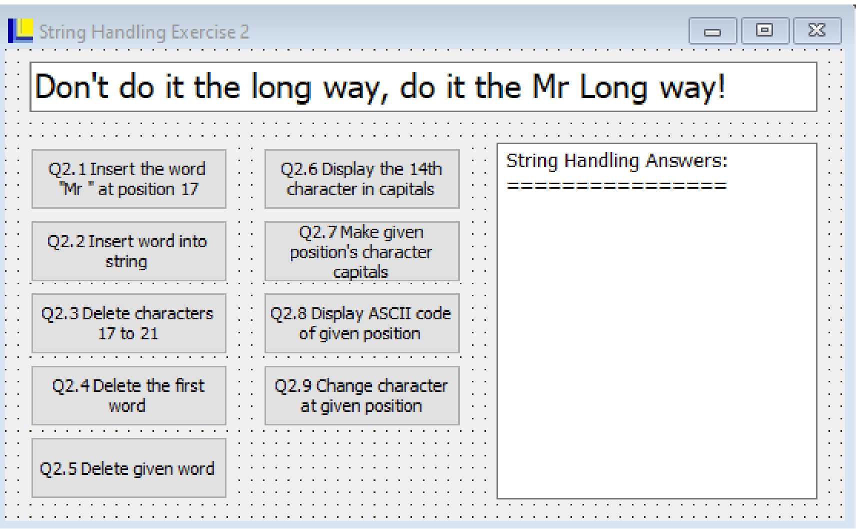
STRINGS

Learners are to make use of the Summary Notes located within learner resource folder for Term 2 Week 8.

Lesson 1

- Complete Exercise 2 found in the Term 2 Week 8 learner resource folder. The data files for Exercise 2 is found within the folder named: *“Grade 10 IT – String Handling Content”*.

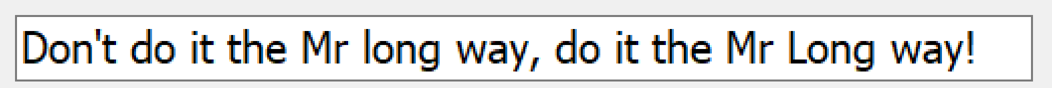
GUI: Exercise 2


String Handling Exercise 2
Q2.1 Insert the word "Mr " at position 17
Q2.2 Insert word into string
Q2.3 Delete characters 17 to 21
Q2.4 Delete the first word
Q2.5 Delete given word
Q2.6 Display the 14th character in capitals
Q2.7 Make given position's character capitals
Q2.8 Display ASCII code of given position
Q2.9 Change character at given position

Don't do it the long way, do it the Mr Long way!

String Handling Answers:
================

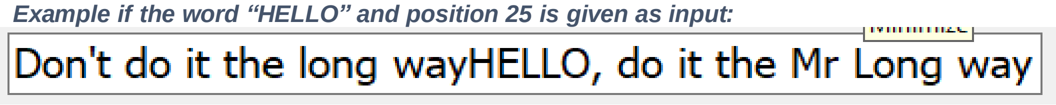
2.1. Complete the code for the ***btnQ21*** button that will insert the string “*Mr* “ into the sentence in the ***edtData*** edit control at position 17. The text in ***edtData*** must reflect the change.



Don't do it the Mr long way, do it the Mr Long way!

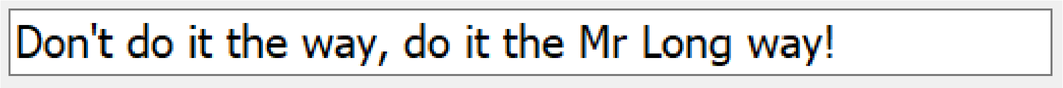
2.2. Complete the code for the ***btnQ22*** button that will prompt the user for a word and a position and then insert the inputted word into the sentence in the ***edtData*** edit control at the inputted position. The text in ***edtData*** must reflect the change.

Example if the word “HELLO” and position 25 is given as input:



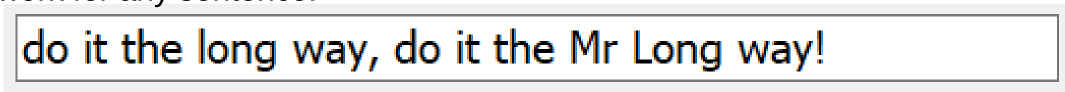
Don't do it the long wayHELLO, do it the Mr Long way

2.3. Complete the code for the ***btnQ23*** button that will delete characters 17 to 21 in the sentence in the ***edtData*** edit control. The text in ***edtData*** must reflect the change.



Don't do it the way, do it the Mr Long way!

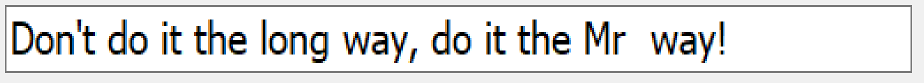
2.4. Complete the code for the ***btnQ24*** button that will delete the first word in the sentence in the ***edtData*** edit control. The text in ***edtData*** must reflect the change. This code must work for any sentence.



```
do it the long way, do it the Mr Long way!
```

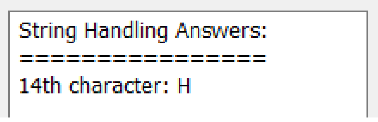
2.5. Complete the code for the ***btnQ25*** button that will prompt the user for a word and then will delete the first occurrence of that word in the sentence in the ***edtData*** edit control. The text in ***edtData*** must reflect the change.

Example if the word "Long" is given as input:



```
Don't do it the long way, do it the Mr  way!
```

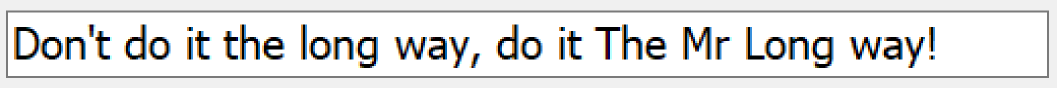
2.6. Complete the code for the ***btnQ26*** button that will display the 14th character of the sentence in the ***edtData*** edit control in uppercase, in the ***memDisplay*** memo control. Use the ***Upcase*** function.



```
String Handling Answers:
================
14th character: H
```

2.7. Complete the code for the ***btnQ27*** button that will prompt the user for a position of a character and will then change the character at that given position in the sentence in the ***edtData*** edit control to uppercase. The text in ***edtData*** must reflect the change. Use the ***Upcase*** function.

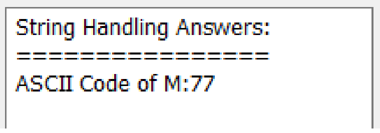
Example if the position 33 is given as input:



```
Don't do it the long way, do it The Mr Long way!
```

2.8. Complete the code for the ***btnQ28*** button that will prompt the user for a position of a character and will then display the ASCII code of the character in the sentence in the ***edtData*** edit control at the inputted position. Display the answer in the ***memDisplay*** memo control.

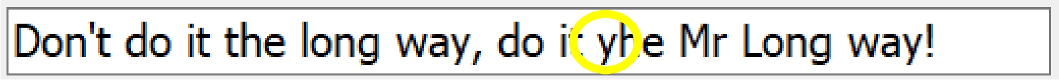
Example if the position 37 is given as input:



```
String Handling Answers:
================
ASCII Code of M:77
```

2.9. Complete the code for the ***btnQ29*** button that will prompt the user for a position of a character and will then change the character at that given position in the sentence in the ***edtData*** edit. It will change the character by changing it to an ASCII character 5 values up. Example: If the '*D*' character was selected, the '*D*' character's ASCII value is *68*. Increase the *68* by 5 to get *73*. The character for the ASCII value of *73* is an '*I*'. So the '*D*' character must be changed to an '*I*'. The text in ***edtData*** must reflect the change.

Example if the position 33 is given as input:

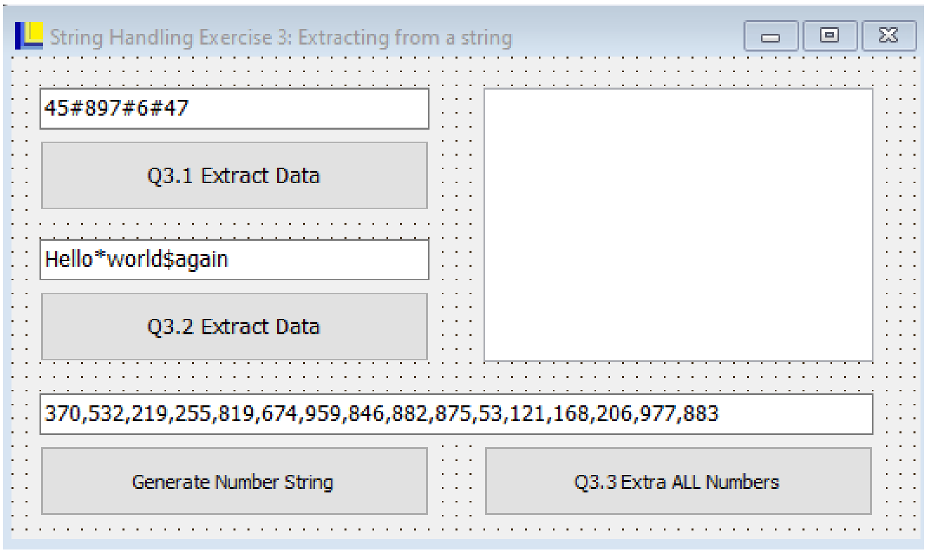


```
Don't do it the long way, do it yhe Mr Long way!
```

Lesson 2

- Complete Exercise 3 found in the Term 2 Week 8 learner resource folder. The data files for Exercise 3 is found within the folder named *“Grade 10 IT – String Handling Content”*.

GUI: Exercise 3



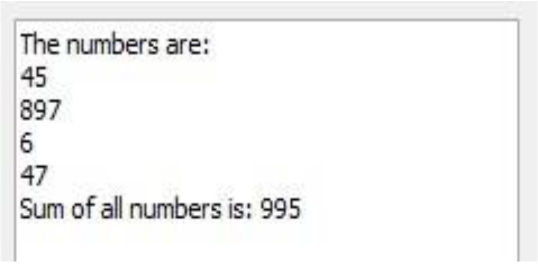
45#897#6#47

Hello*world$again

370,532,219,255,819,674,959,846,882,875,53,121,168,206,977,883

3.1. Complete the code for the ***btnQ31*** button’s onClick event handler that that receives an input string that consists of a set of FOUR numbers separated by a hash (#) symbol from the ***edtCode*** edit control (Example: *45#897#6#47*).

The program must extract the four numbers ***Example for the string: 45#897#6#47*** from the string and display them separately in the ***memDisplay*** memo control. Assume that the number of digits for each number is unknown and the program must work for any set of four numbers of any length. The sum of the four numbers must also be displayed in the **memDisplay** memo control as shown in the diagram.

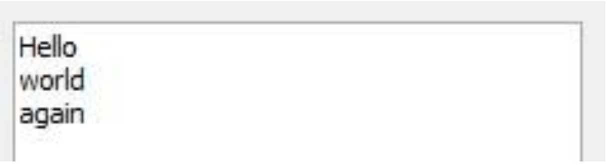


```
The numbers are:
45
897
6
47
Sum of all numbers is: 995
```

3.2. Complete the code for the ***btnQ32*** button’s onClick event handler that that receives as input a string that consists of a set of THREE words, the first word ends with the star (*) symbol and the second word ends with a dollar ($) symbol from the ***edtWords*** edit control (Example: *Hello*world$again*).

The program must extract the three words ***Example for the string: Hello*world$again*** from the string and display them separately in the ***memDisplay*** memo control.

Assume that the number of letters for each word is unknown and the program must work for any set of three words of any length.



```
Hello
world
again
```

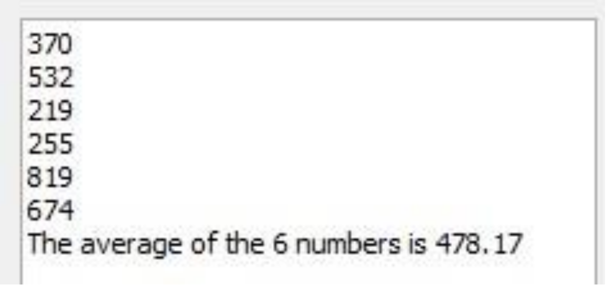
3.3. When the button, ***btnGenerate***, is clicked, a string with a random set of numbers separated by commas is displayed in the ***edtLongSet*** edit control. The amount of numbers generated is unknown and the number of digits in each number is random.

(Example: *370,532,219,255,819,674*)

Complete the code for the ***btnQ33*** button's onClick event handler that that receives as input the generated string from the ***edtLongSet*** edit control and extracts ALL the individual numbers and displays them individually in the ***memDisplay*** memo control.

The average of all the numbers in the string must also be displayed (to TWO decimal places) in the **memDisplay** memo control as shown in the diagram.

Example for the string: 370,532,219,255,819,674

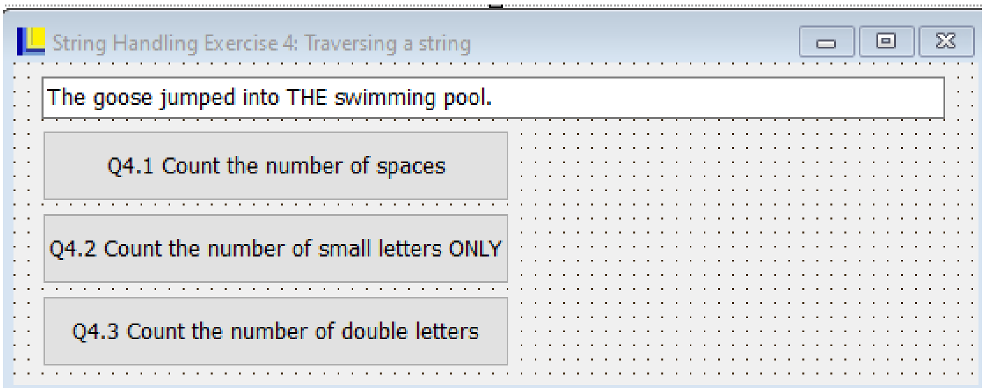


```
370
532
219
255
819
674
The average of the 6 numbers is 478.17
```

Lesson 3

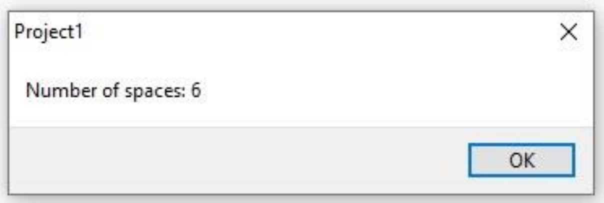
- Complete Exercise 4 found in the Term 2 Week 8 learner resource folder. The data files for Exercise 4 is found within the folder named *“Grade 10 IT – String Handling Content”*.

GUI: Exercise 4


String Handling Exercise 4: Traversing a string
Q4.1 Count the number of spaces
Q4.2 Count the number of small letters ONLY
Q4.3 Count the number of double letters

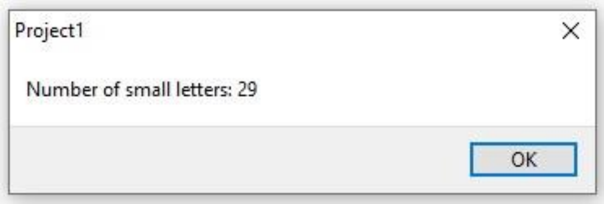
The goose jumped into THE swimming pool.

4.1. Complete the code for the ***btnQ41*** button’s onClick event handler that receives, as input, a sentence from the ***edtSentence*** edit control and counts the number of spaces that appear in the sentence. Show your answer using a ***showmessage*** procedure.



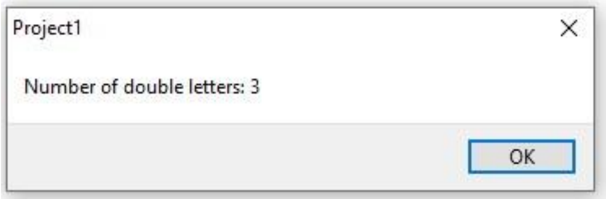
Number of spaces: 6

4.2. Complete the code for the ***btnQ42*** button’s onClick event handler that that receives as input a sentence from the ***edtSentence*** edit control and counts the number of small letters (letters in lowercase) that appear in the sentence. Show your answer using a ***showmessage*** procedure.



Number of small letters: 29

4.3. Complete the code for the ***btnQ43*** button’s onClick event handler that that receives, as input, a sentence from the ***edtSentence*** edit control and counts the number of double letters (when two consecutive letters are the same, for example book, will, three) that appear in the sentence. Show your answer using a ***showmessage*** procedure.

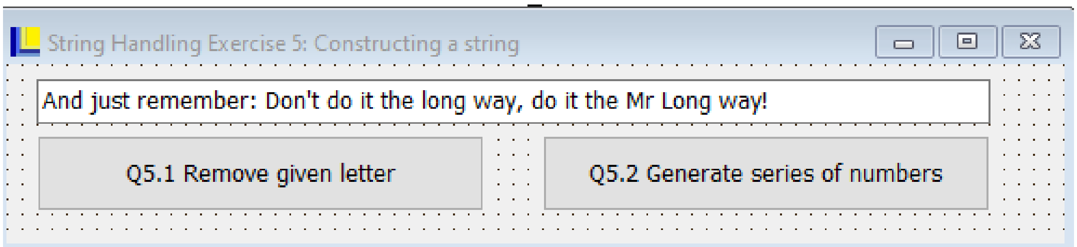


Number of double letters: 3

Lesson 4

- Complete Exercise 5 and String Handling Final Challenge found in the Term 2 Week 8 learner resource folder. The data files for Exercise 5 and String Handling Final Challenge are found within the folder named *"Grade 10 IT – String Handling Content"*.

GUI: Exercise 5

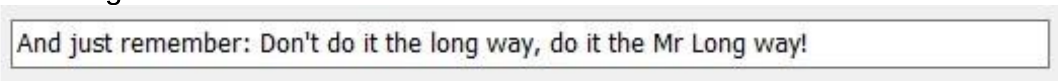


String Handling Exercise 5: Constructing a string

And just remember: Don't do it the long way, do it the Mr Long way!

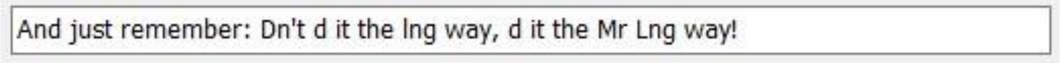
5.1. Complete the code for the ***btnQ51*** button's onClick event handler that that receives as input sentence from the ***edtSentence*** edit control and prompts the user for character. All occurrences of that character in the inputted sentence must be removed and the new sentence must be displayed back in the ***edtSentence*** edit control (new sentence with all occurrences of inputted character removed. If the inputted character is a letter, then BOTH uppercase and lowercase versions of that letter must be removed.

Original sentence:



And just remember: Don't do it the long way, do it the Mr Long way!

Sentence if the letter 'o' is given when prompted:



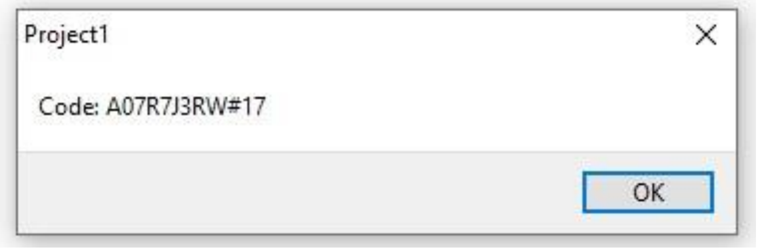
And just remember: Dn't d it the lng way, d it the Mr Lng way!

5.2. Complete the code for the ***btnQ52*** button's onClick event handler that that prompts the user for a number representing the number of characters in a code and then generates a code consisting of random digits (0-9) and capital letters (A-Z) of the length given as input. character. Display the code using a ***showmessage*** procedure.

At the end of the code, add a hash (#) symbol following by the SUM of all the digits (0-9) in the code.

Example (if the number 9 is given when prompted:

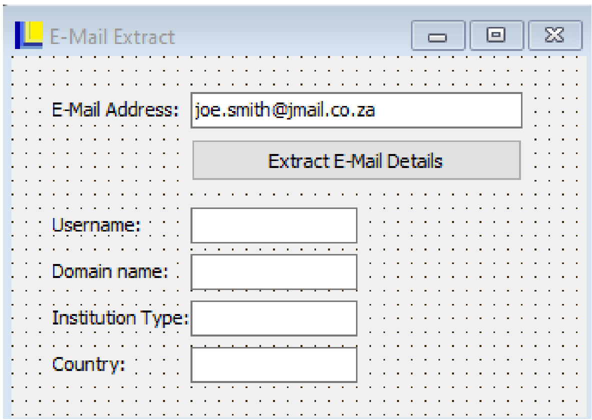
- 9 character code: *A07R7J3RW*
- Sum of the digits: *A**07**R**7**J**3**RW* therefore 0 + 7 + 7 +3 = 17
- Final code is: ***A07R7J3RW#17***



Project1

Code: A07R7J3RW#17

GUI: Final Challenge Q1



E-Mail Address: joe.smith@jmail.co.za

Complete the code for the ***OnClick*** event handler of the ***btnQ1*** button that receives as input an e-mail address via the ***edtEMail*** edit control. The program must then extract the following information from an e-mail address such as joe.smith@jmail.co.za as follows:

- The username which is all the text that occurs before the '@' symbol.
- The domain which is the text that occurs after the '@' symbol up to the first full stop.
- The website type which is the text that occurs after the first full stop.
- The country code which is the text that occurs after the second full stop. Some e-mail addresses do not reference the country code. If the e-mail address has no country code (only one fullstop after the '@' symbol), then a '***NA***' must be displayed for the country code. The extract information must be displayed in the appropriate edit controls provided.

  For example:

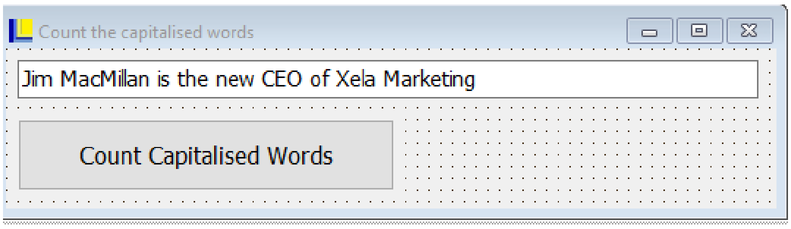
If the e-mail ***joe.smith@jmail.co.za*** is given as input then

Username	Domain	Website Type	Country Code
joe.smith	Jmail	co	za

If the e-mail ***brucebanner23@avengers.com*** is given as input then

Username	Domain	Website Type	Country Code
brucebanner23	avengers	com	NA

GUI: Final Challenge Q2



Jim MacMillan is the new CEO of Xela Marketing

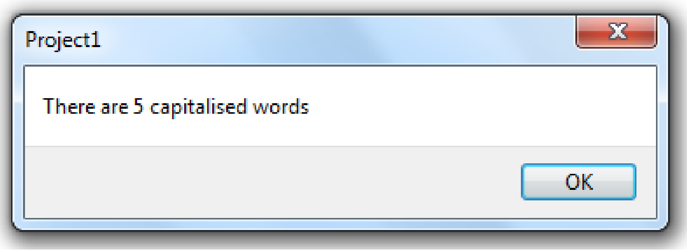
Complete the code for the ***OnClick*** event handler of the ***btnQ2*** button that receives as input a sentence via the ***edtSen*** edit control and must count the number of words that START with a CAPITAL LETTER.

HINT: *Every word in the sentence that starts with a capital letter has a space INFRONT of the capital letter EXCEPT the first word.*

For example: If the sentence is

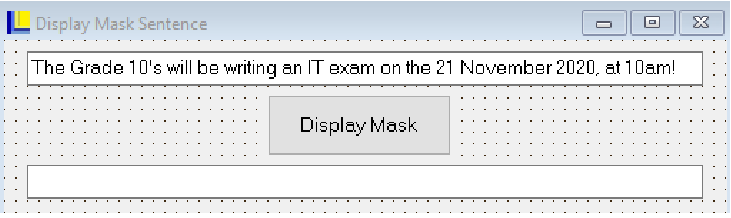
“***Jim MacMillan is the new CEO of Xela Marketing.***”

Then there are 5 words that start with a capital letter: Jim, MacMillan, CEO, Xela, Marketing Display your answer as shown in the diagram below.



There are 5 capitalised words

GUI: Final Challenge Q3

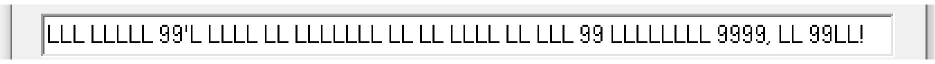


The Grade 10's will be writing an IT exam on the 21 November 2020, at 10am!

Complete the code for the ***OnClick*** event handler of the ***btnQ3*** button that receives as input a sentence via the ***edtSen*** edit control and must construct a new string with the following criteria:

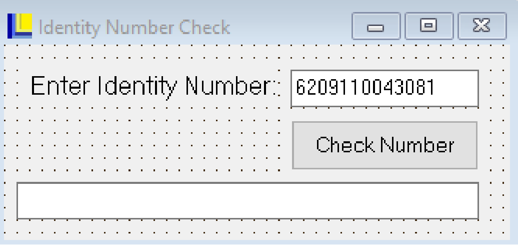
- All letters (upper or lowercase) must be replaced with an L
- All numbers must be replaced with a 9
- Any character that is not a letter or number (example: space, comma, full stop, etc) must remain the same.

The new string must be displayed in the ***edtDisplay*** edit control.



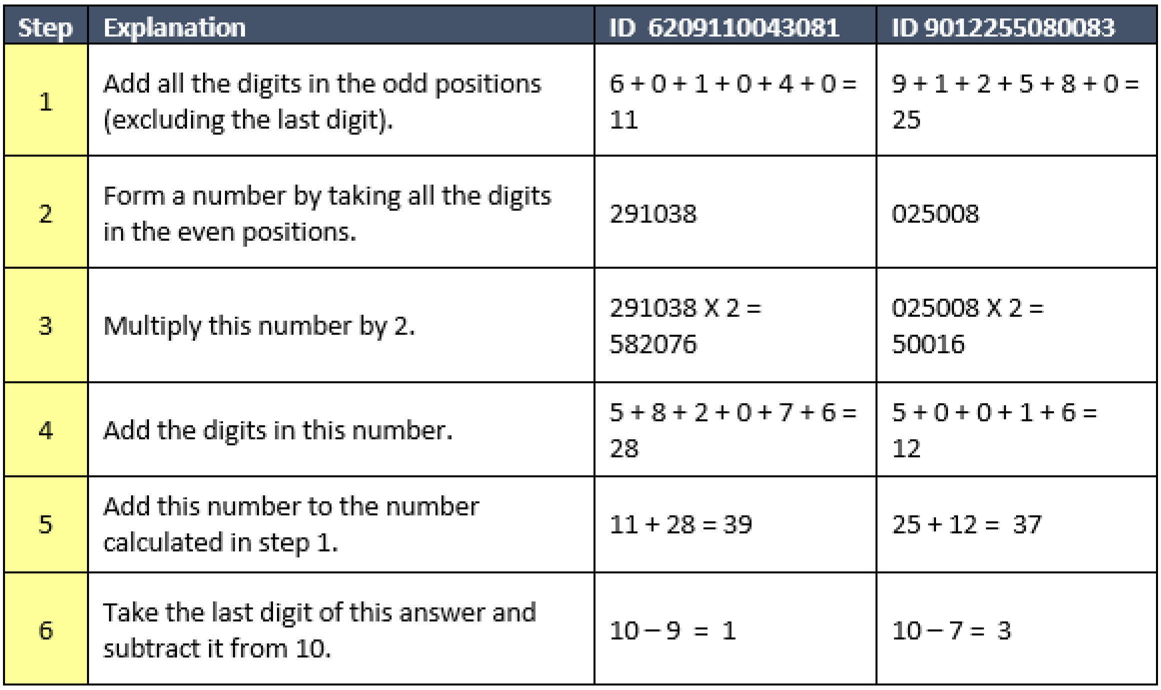
LLL LLLLL 99'L LLLL LL LLLLLLL LL LL LLLL LL LLL 99 LLLLLLLL 9999, LL 99LL!

GUI: Final Challenge Q4



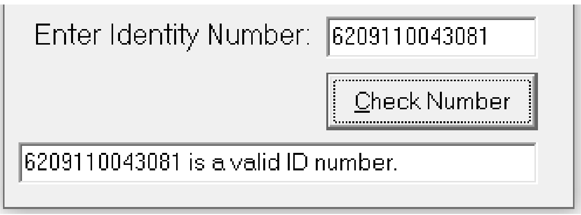
Enter Identity Number: 6209110043081

Complete the code for the ***OnClick*** event handler of the ***btnQ4*** button that receives as input a South African ID number via the ***edtID*** edit control and must check if the 13th digit of the ID number is valid. The 13th digit of all identity numbers is a check digit which is calculated as follows, using the ID numbers *6209110043081* and *9012255080083* as examples:



Step	Explanation	ID 6209110043081	ID 9012255080083
1	Add all the digits in the odd positions (excluding the last digit).	6 + 0 + 1 + 0 + 4 + 0 = 11	9 + 1 + 2 + 5 + 8 + 0 = 25
2	Form a number by taking all the digits in the even positions.	291038	025008
3	Multiply this number by 2.	291038 X 2 = 582076	025008 X 2 = 50016
4	Add the digits in this number.	5 + 8 + 2 + 0 + 7 + 6 = 28	5 + 0 + 0 + 1 + 6 = 12
5	Add this number to the number calculated in step 1.	11 + 28 = 39	25 + 12 = 37
6	Take the last digit of this answer and subtract it from 10.	10 – 9 = 1	10 – 7 = 3

Display your result in the ***edtAnswer*** edit control as shown in the diagram below:



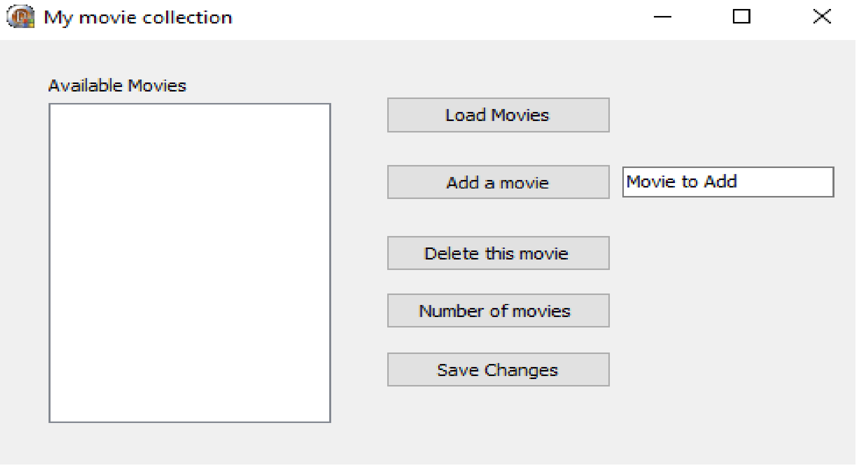
Enter Identity Number: 6209110043081

6209110043081 is a valid ID number.

Enrichment Activity: Using a listbox

Complete the enrichment activity "*Movies folder*" found in the Term 2 Week 8 learner resource folder.

You have some movies on an external hard drive. You want an application that will help you to manage your movies.



- Write code for *btnLoadMovies* to load the movies from the file Movies.txt in your folder into the *ListBox*.
- Write code for *btnAdd* to add a new movie to the list. The name of the movie should be entered in *edtAdd*.
- Add code to the *btnDelete* event to delete a movie that a user clicked on in the *ListBox*.
- Write the code for *btnNumMovies* to display the number of movies in the *ListBox* in a *ShowMessage* dialog box.
- Write code for *btnSaveChanges* to save the content of the *ListBox* as it is displayed in *lstMovies*.
- Save and run your app.

WEEK 9 - 10

NETWORKS + E-COMMUNICATIONS + SOCIAL IMPLICATIONS

Lesson 1

- Complete Activity 5.1 on page 102 of the electronic theory learner textbook.

Lesson 2

Activity: Networks

1. Define a network.

____
____
____

2. Why would you want to set up a network at your home?

____
____
____

3. Name two methods of communication through a network.

____
____

4. What factors will make it difficult for you to set up a network at home?

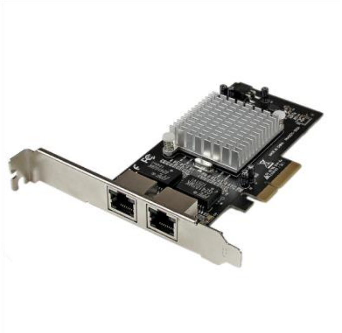
____
____
____

5. What does the term "node" refer to?

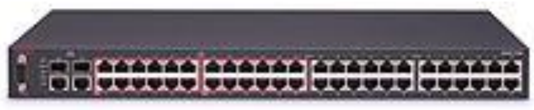
____

6. Identify and name the following pictures:

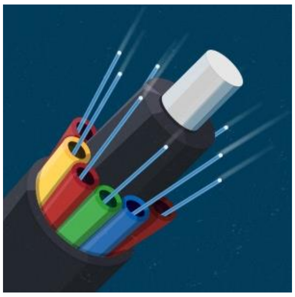
A. ________________________



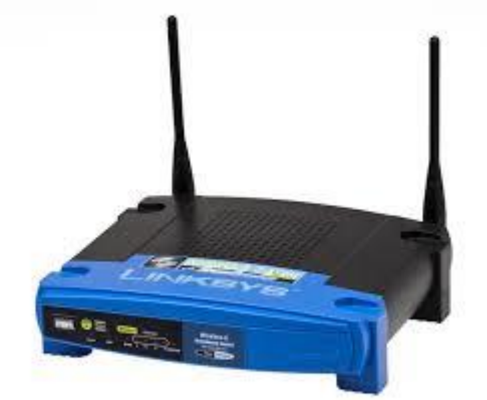
B. ________________________



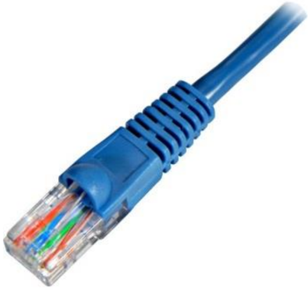
C. ________________________



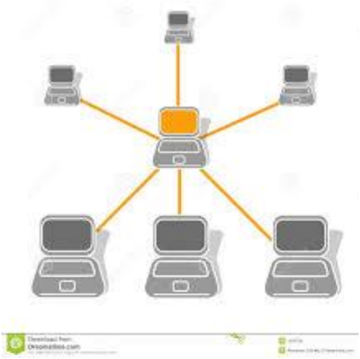
D. ________________________



E. ________________________



F. ________________________



7. What to communication media do we usually use in a network?
________________________ or ________________________

8. Why would you rather set up a WLAN at home instead of a normal LAN?
____
____
____

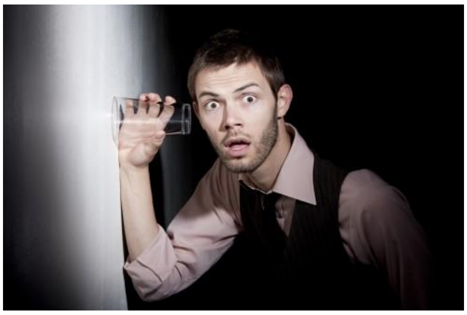
9. A WAN is usually connected by ________________________ or
________________________ or
____

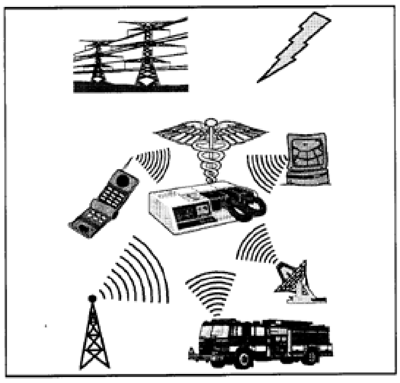
10. The internet is also known as _____________

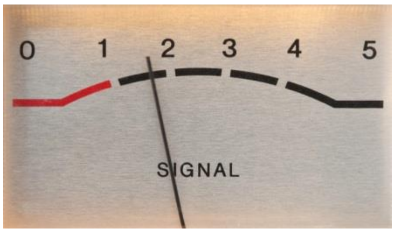
11. Fill in the table to name 4 differences between a peer-to-peer and client-server network.

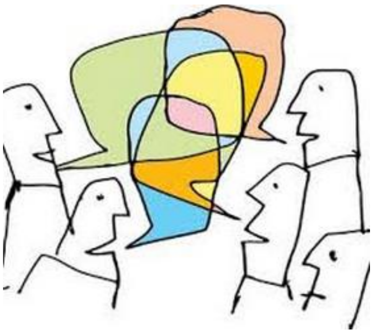
Peer-to-peer	**Client-server**

12. Identify the terms that the pictures represent in failures in communication media:

A. 

B. 

C. 

D. 

Lesson 3 and 4

- Complete Activity 6.1 on page 117-118 of the electronic theory learner textbook.
- Complete Activity 6.2 on page 124 of the electronic theory learner textbook.

Lesson 5

Activity: E-Communications

1.1. I want to use my computer to attend meetings in other countries.

1.1.1 What communication protocol will I make use of to set this up? ______________

1.1.2 Name one software package that I need to install to ensure that I can participate in a video conference? ____

1.2. Give one word for the following descriptions :

1.2.1 An application that gives two or more people the ability to have a conversation by typing on a computer or cellphone. ___________________________________

1.2.2 This can be used in a school situation to allow teachers to communicate with parents. ____

1.2.3 An application that allows two or more people to hold on-line conferences from different locations using a computer network to transmit audio and video data as if they were all in the same room. _______________________________________

1.3. Define the term RSS and give one advantage of an RSS feed.

____

____

____

1.4. You need to set up the following email address for your uncle:

pietermathews@mweb.co.za
1 2 3 4

Explain the format (layout) by referring to each part numbered.

1. ______________________________________

2. ______________________________________

3. ______________________________________

4. ______________________________________

1.5. The sending/receiving of emails is a daily happening for many users. Name the code of conduct that guides users on the proper use of emails. List four such guidelines.

____

____

____

____

1.6. What are blogs? ____

1.7. The secretary sent electronic brochures to a number of clients via e-mail.
1.7.1. The secretary complains that she is suddenly receiving lots of spam. What is spam?

____

1.7.2. Give TWO of the main advantages of e-communication.

____

____

1.7.3. Give TWO disadvantages of e-communication with reference to the human aspect of communication.

____

____

____

1.7.4. She received feedback from some clients that mentioned that the e-mail took quite a long time to download. Name two reasons why the e-mails took a long time to download.

____

____

____

2. Fill in the table to name 4 differences between an ISP e-mail and a web-based e-mail.

ISP	Web-based

3. When will you enter a recipients e-mail address in the TO, CC or BCC fields?

TO: ____

CC: ____

BCC: ____

Lesson 6

Activity: Social Implications (E-mail threats and issues)

1. Define the following terms:
   - firewall
   - virus
   - anti-virus
   - spyware
   - anti-spyware

2. Explain the difference between a worm and a Trojan horse.

3. How will you know when your computer has a virus? List at least THREE common symptoms.

4. What makes e-mail the ideal method for spreading malware?

5. What are virus definitions and why do we need to keep our anti-virus program's definitions updated?

6. Briefly discuss the importance of data security or data protection.

WEEK 11

EVENTS + VALIDATION

Lesson 1

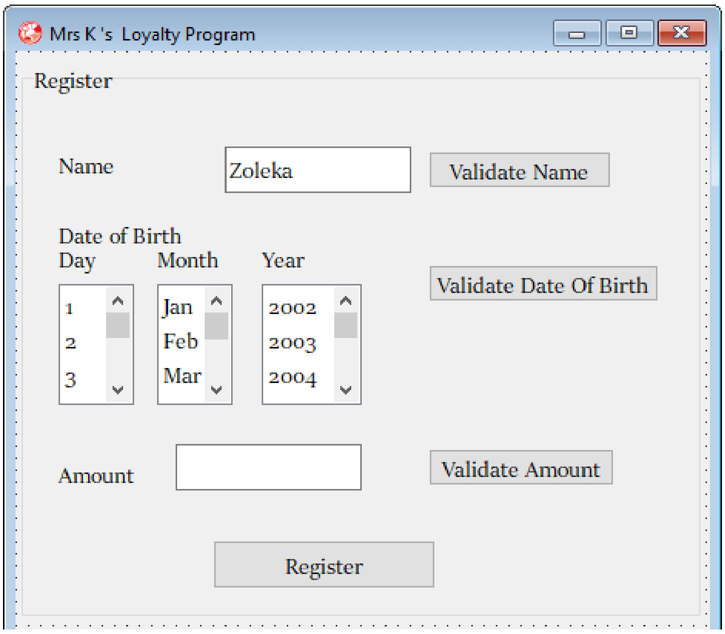
- Complete Activity 2.4 on page 30 of the electronic practical learner textbook.
- Complete Activity 6.2 on page 142 of the electronic practical learner textbook.

Lesson 2

Activity: Internet Café

Complete the program in the folder '*Internet Café'* found in the Term 2 Week 11 learner resource folder.

Mrs K runs an internet café. She has a loyalty program where customers register to get points and discounts. She realised that her customers sometimes register with incorrect data. Complete the validation of her program.



Name: Zoleka

1.1 Button **Validate Name**

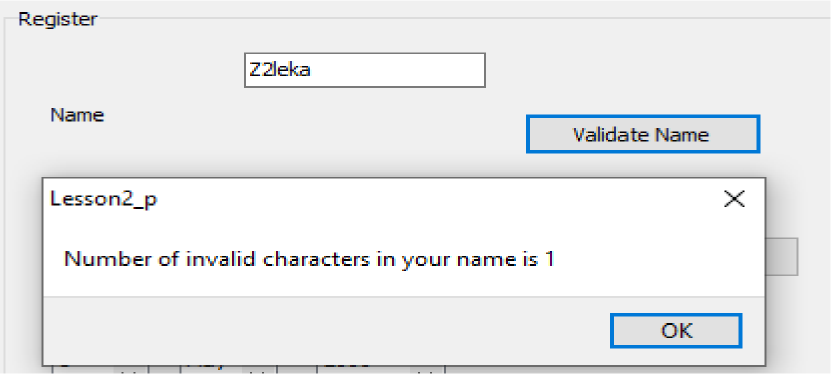
A valid name consist of the letters of the alphabet in the range [a..z,A..Z].

Write code to do the following:
Declare global variables for the user's input

- Initialize the counter to zero in the oncreate event.
- Check if each character is valid using the IN operator
- Count the number of invalid characters
- Call the validateName procedure.

If the name is valid, enable the button ValidateDateOfBirth.

SAMPLE OUTPUT : Invalid Name



Number of invalid characters in your name is 1

1.2 Button **ValidateDateofBirth**

Three listbox are provided , lstDay for the day of birth, lstMonth for the year of birth, lstYear for the year of birth.

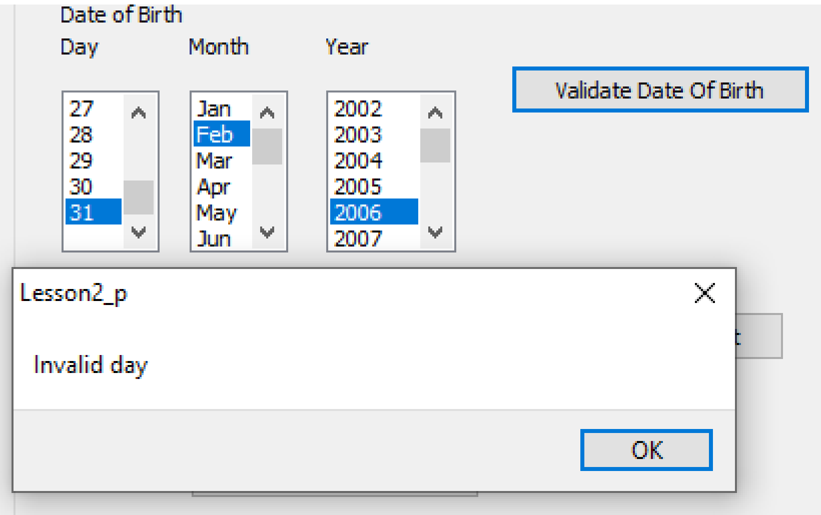
Write code to ensure the following:

- A person born in February cannot be born on a day greater than the 29th .
- A person born in April, June, September, November cannot be born on the 31st .

Display a suitable message if the date of birth is invalid.

➢ If the date of birth is valid enable the **ValidateAmount** button

SAMPLE OUTPUT: when 31 Feb is selected



Invalid day

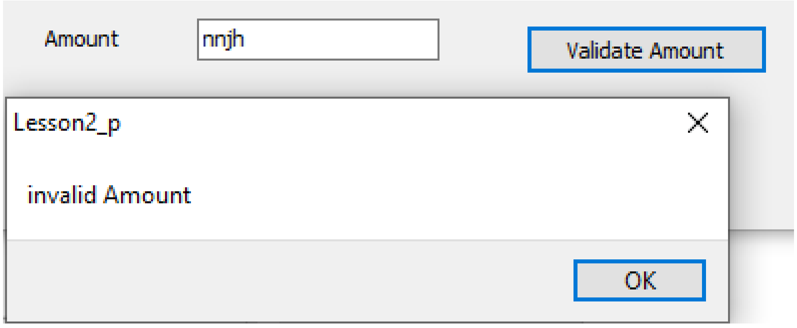
1.3 Button **Validate Amount**

It costs R150 to register for the loyalty program. A constant variable RegAmount is provide.

Write code to do the following.

- Use try and except to check if a number has been entered. Display a suitable message if the amount is invalid.
- Display the amount Owed by the user after paying.

SAMPLE OUTPUT: when invalid amount is entered



Amount nnjh

Lesson2_p

invalid Amount

If the amount is valid enable the **Register** button.

1.4 Button **Register**

Two listbox are provided, lstNames to save the users' names and lstDOB to save the users' date of birth.

- Add the name in the lstName listbox.
- Add the date of birth in the lstDOB listbox in the format yyyy/mm/dd.
- Display a suitable message that the customer has been added.

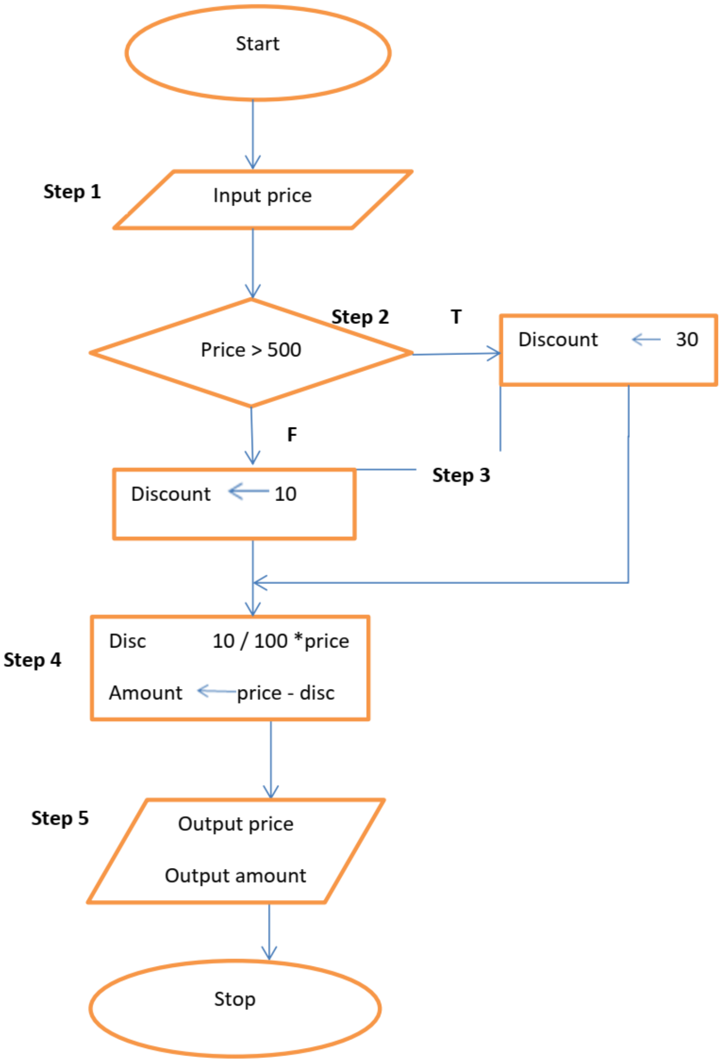
Activity: Debugging

1. The following flowchart contains an error. Trace through the flowchart and indicate at which step the error is located.

2. a) What type of error is it? Logical, Syntax or Runtime.
   b) Explain your choice.

3. Correct the error and thereafter trace through the correct flowchart with the below input values, using a trace table:

- **250**
- **630**



Lesson 3

- Complete Activity 6.2 on page 142 of the electronic practical learner textbook.
- Complete Activity 3.6 on page 56 of the electronic practical learner textbook.

CONCLUSION

The STEP AHEAD programme is designed to cater for learning losses in the 2021 academic year due to the disruption of teaching and learning as a result of the Covid pandemic.

We place on record our sincere thanks and appreciation to all our Information Technology Lead Teachers and Subject Advisors that were involved in the compilation of these booklets.