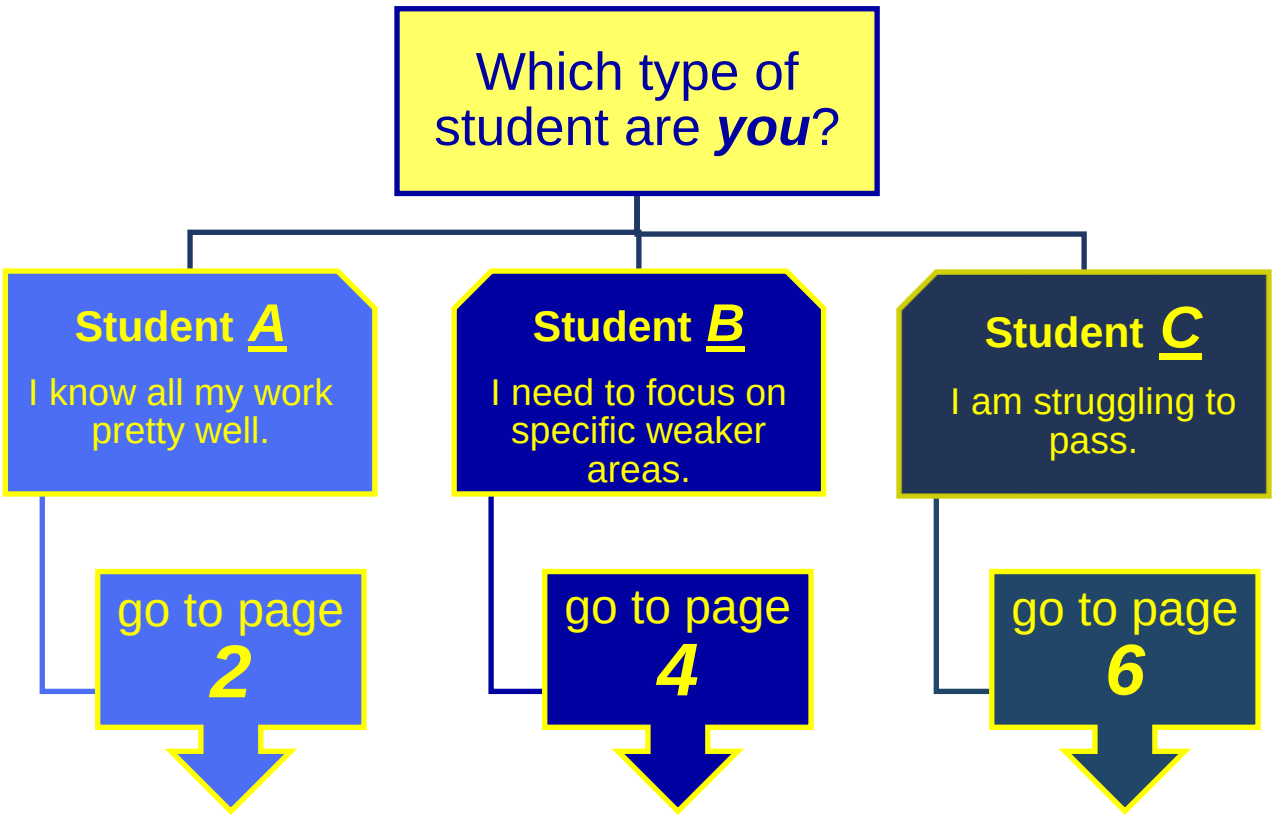
Mr Long Exam Guide 2021

Need help preparing for your IT practical exam?

Which type of student are you?



Student A – *I know all my work pretty well. What now?*

- ✓ Focus on examination preparation.
- ✓ Go through previous exam papers (Use table below).
- ✓ Attempt the exam paper first, using exam conditions:
  - 🛠 Don't use any resources like notes and textbooks.
  - 🛠 Do the WHOLE exam paper in a 3 hour period.
    (Once the 3 hour time limit is over, take note of how far you got, if not finished complete the exam)
  - 🛠 Use this to practice your time management of a 3 hour exam.
  - 🛠 Once you have completed the exam (or at a later time) identify areas that need improvement (Questions that need to be done faster, sections of work that need revision, etc).
  - 🛠 Use the memo and / or videos of the exam paper to mark your own work.
  - 🛠 Identity where you made mistakes, lost marks or misunderstood a question.

Previous Exam Papers

Paper	Description	Strategy and Link
2020 Final November *download here*	Question 1 - General Programming Skills 🛠 1.1 – 4 marks 🛠 1.2 – 8 marks 🛠 1.3 – 11 marks 🛠 1.4 – 17 marks ***TOTAL: 40 marks*** Question 2 – SQL and Database 🛠 2.1.1 to 2.1.5 – 22 marks 🛠 2.2.1 to 2.2.3 – 18 marks ***TOTAL: 40 marks*** Question 3 – Object Oriented Programming 🛠 3.1.1 to 3.1.5 – 23 marks 🛠 3.2.1 to 3.2.3 – 17 marks ***TOTAL: 40 marks*** Question 4 – Problem-Solving 🛠 4.1 – 12 marks 🛠 4.2 – 18 marks ***TOTAL: 30 marks***	✓ Questions 1, 2 & 3 = 48 minutes each ✓ Question 4 = 36 minutes Video Memo 👁 Question 1 = 3 videos 👁 Question 2 = 2 videos 👁 Question 3 = 2 videos 👁 Question 4 = 1 videos 👁 *Link to playlist* OR Scan QR code below

More exam papers lists below

PLEASE NOTE:
If you can't access the links (because document is printed) or if any links do not work then
- *Go to YouTube and search for **Mr Long IT and CAT***
- *Then go to playlists and search for relevant exam paper*
- *All videos have links to all videos and to data files in the Description*

Paper	Description	Strategy and Link
2019 Final November *download here*	Question 1 - General Programming Skills 🛠 1.1 – 8 marks 🛠 1.2 – 10 marks 🛠 1.3 – 9 marks 🛠 1.4 – 13 marks ***TOTAL: 40 marks*** Question 2 – SQL and Database 🛠 2.1.1 to 2.1.5 – 19 marks 🛠 2.2.1 to 2.2.2 – 21 marks ***TOTAL: 40 marks*** Question 3 – Object Oriented Programming 🛠 3.1.1 to 3.1.6 – 20 marks 🛠 3.2.1 to 3.2.4 – 20 marks ***TOTAL: 40 marks*** Question 4 – Problem-Solving 🛠 4.1 – 11 marks 🛠 4.2 – 19 marks ***TOTAL: 30 marks***	✓ Questions 1, 2 & 3 = 48 minutes each ✓ Question 4 = 36 minutes Video Memo 👁 Question 1 = 2 videos 👁 Question 2 = 3 videos 👁 Question 3 = 2 videos 👁 Question 4 = 2 videos 👁 ***Link to playlist*** OR Scan QR code below
2018 Final November *download here*	Question 1 - General Programming Skills 🛠 1.1 – 4 marks 🛠 1.2 – 9 marks 🛠 1.3 – 13 marks 🛠 1.4 – 14 marks ***TOTAL: 40 marks*** Question 2 – SQL and Database 🛠 2.1.1 to 2.1.5 – 25 marks 🛠 2.2.1 to 2.2.3 – 15 marks ***TOTAL: 40 marks*** Question 3 – Object Oriented Programming 🛠 3.1.1 to 3.1.4 – 17 marks 🛠 3.2.1 to 3.2.3 – 21 marks ***TOTAL: 38 marks*** Question 4 – Problem-Solving 🛠 4.1 – 14 marks 🛠 4.2 – 18 marks ***TOTAL: 32 marks***	✓ Questions 1 & 2 = 48 minutes each ✓ Question 3 = 46 minutes ✓ Question 4 = 38 minutes Video Memo 👁 Question 1 = 3 videos 👁 Question 2 = 2 videos 👁 Question 3 = 2 videos 👁 Question 4 = 2 videos 👁 ***Link to playlist*** OR Scan QR code below
2018 Exemplar *download here*	Question 1 - General Programming Skills 🛠 1.1 – 6 marks 🛠 1.2 – 14 marks 🛠 1.3 – 8 marks 🛠 1.4 – 12 marks ***TOTAL: 40 marks*** Question 2 – SQL and Database 🛠 2.1.1 to 2.1.5 – 20 marks 🛠 2.2.1 to 2.2.3 – 20 marks ***TOTAL: 40 marks*** Question 3 – Object Oriented Programming 🛠 3.1.1 to 3.1.5 – 22 marks 🛠 3.2.1 to 3.2.3 – 18 marks ***TOTAL: 40 marks*** Question 4 – Problem-Solving 🛠 4.1 – 16 marks 🛠 4.2 – 14 marks ***TOTAL: 30 marks***	✓ Questions 1, 2 & 3 = 48 minutes each ✓ Question 4 = 36 minutes Video Memo 👁 Question 1 = 2 videos 👁 Question 2 = 3 videos 👁 Question 3 = 2 videos 👁 Question 4 = 2 videos + 1 extra 👁 ***Link to playlist*** OR Scan QR code below

Student B – *I need to focus on specific weaker areas. What now?*

- ✓ A practical exam is divided into FOUR questions. Identity which questions are your weaker areas and then use the table below to work through that area.
- ✓ For each area:
  - 🛠 Watch the lesson series on the potential topics.
  - 🛠 Review the summary note on the content.
  - 🛠 Practice on that area's exam type questions.
    If you get stuck on a question, then refer back to note for guidance.
  - 🛠 Identify common algorithms (or "recipe code") that you can study in advance to help with your time management of a practical exam.

Practical Exam Sections

Section	Action	Resources Links
Question 1: General Programming Skills • Basic Grade 10 & 11 work • Basic calculations • Selection and Looping • String Manipulation	**Watch Lesson Series** 👁 Selection Programming 👁 For Loops 👁 Conditional Loops 👁 String Handling	Selection Programming Series For Loops Series Part 1 - 4 Conditional Loops Series Part 1 - 4 String Handling Series
	Review Summary Notes Selection Programming Summary For Loops Summary Conditional Loops Summary String Handling Summary	Selection Programming Summary For Loops Summary Conditional Loops Summary String Handling Summary
	Practice Question 1 🛠 November Final 2020 🛠 November Final 2019 🛠 November Final 2018 🛠 Exemplar 2018	November Final 2020 Data Files Q1.1-1.2 Q1.3 Q1.4 November Final 2019 Data Files Q1.1-1.2 Q1.3-1.4 November Final 2018 Data Files Q1.1-1.2 Q1.3 Q1.4 Exemplar 2018 Data Files Q1.1-1.2 Q1.3-1.4

Question 2: SQL and Database
• CRUD (Create, Read, Update, Delete)
• Manipulate data in DB
• Perform queries
• SQL
  ○ Select, Distinct,
  ○ Where, Order by
  ○ Group by, Having
  ○ Like, Between, Null
  ○ Aggregate functions
  ○ Date functions
  ○ String functions
  ○ Dynamic queries
  ○ Insert, Delete, Update
	Watch Lesson Series
	👁 Databases in Delphi
	👁 SQL module
		Databases in Delphi Series Videos 3 - 9
		SQL Series

	Review Summary Notes Databases in Delphi Summary SQL Summary	Databases in Delphi Summary SQL Summary
	Practice Question 1 🛠 November Final 2020 🛠 November Final 2019 🛠 November Final 2018 🛠 Exemplar 2018	November Final 2020 Data Files Q2.1 (SQL) Q2.2 November Final 2019 Data Files Q2.1 (SQL) Q2.2.1 Q2.2.2 November Final 2018 Data Files Q2.1 (SQL) Q2.2 Exemplar 2018 Data Files Q2.1 (SQL) Q2.2.1 Q2.2.2

Section	Action	Resources Links
Question 3: Object Oriented Programming • Private & public attributes and methods • Constructors • Accessor, Mutator, Auxiliary, toString methods • Instantiation of objects • Use of methods	**Watch Lesson Series** OOP Series	Object Oriented Programming Series Videos 1-4
	Review Textbook DBE Practical Textbook	DBE Practical Textbook (Chapter 2)
	Practice Question 1 November Final 2020 November Final 2019 November Final 2018 Exemplar 2018	November Final 2020 Data Files Q3.1 Q3.2 November Final 2019 Data Files Q3.1 Q3.2 November Final 2018 Data Files Q3.1 Q3.2 Exemplar 2018 Data Files Q3.1 Q3.2
Question 4: Problem Solving • Apply all programming constructs, techniques, algorithms • Recognise patterns • Customise algorithms • Computational thinking • Examples include: ○ Text files ○ Arrays	**Watch Lesson Series** Text Files Arrays Advanced Arrays 2D Arrays	Text File Series Videos 1 - 3 Arrays Module Video 1 – 10 (Not 4 & 7) Advanced Arrays Videos 1 – 5 Advanced Arrays Videos 6 - 8
	Review Summary Notes Text Files Summary Arrays Summary Advanced Arrays Summary	Text Files Summary Arrays Summary Advanced Arrays Summary
	Practice Question 1 November Final 2020 November Final 2019 November Final 2018 Exemplar 2018	November Final 2020 Data Files Q4 November Final 2019 Data Files Q4.1 Q4.2 November Final 2018 Data Files Q4.1 Q4.2 Exemplar 2018 Data Files Q4.1 Extra Q4.2

PLEASE NOTE:

If you can't access the links (because document is printed) or if any links do not work then

- *Go to YouTube and search for **Mr Long IT and CAT***
- *Then go to playlists and search for relevant exam paper*
- *All videos have links to all videos and to data files in the Description*

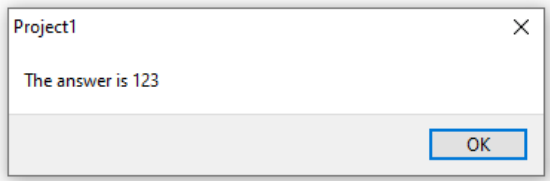
For more exercises on specific content use the following links:

- Selection Programming - https://tinyurl.com/MLE-G10IT-SelectionProgramming
- For Loops - https://tinyurl.com/MLE-G10IT-ForLoops
- Conditional Loops - https://tinyurl.com/MLE-G10IT-ConditionalLoops
- String Handling - https://tinyurl.com/MLE-G10IT-StringHandling
- Databases in Delphi - https://tinyurl.com/MLE-G11IT-DatabasesInDelphi
- SQL - https://tinyurl.com/MLE-G12IT-SQL
- Text Files - https://tinyurl.com/MLE-G11IT-TextFiles
- Arrays - https://tinyurl.com/MLE-G11IT-Arrays
- Advanced Arrays - https://tinyurl.com/MLE-G12IT-AdvancedArrays

Student C – *I am struggling to pass. What now?*

- ✓ If you are struggling to pass and there is limited time left until your exam, focus on specific key algorithms and examples that are commonly asked in exams.
- ✓ Identify key algorithms that you can study and know "off by heart".
- ✓ Remember to change variable names to ones used in your practical exam.
- ✓ Identify all the inputs (values coming from edit boxes, spin edits, InputBox, etc) and place them into variables.
- ✓ Identify all the outputs (anything displayed in edit boxes, rich edit controls or using ShowMessage) and display the answer as shown in the example in the exam paper
  - 🛠 Take note of formats like currency etc.
  - 🛠 If you require to display an answer from a variable, create that variable, assign it a value manually and then do the output.

Practical Exam Tips

Section	Scenario in Exam	Example or Resources Links
General Programming Tips	Any input from edit control or spin edit: 🛠 Make variable 🛠 Place value from component into variable 🛠 Check if conversation must occur (StrToInt)	sName := edtName.Text ; iNum := spnGrade.Value ; rNum := StrToFloat(edtTotal.Text) ;
	Any reference to word ***prompt*** the user and there isn't a component: 🛠 Use an **InputBox** function (3 string parameters) 🛠 Place value into variable 🛠 Check if conversation must occur (StrToInt)	sName := ***InputBox***('Name', '' , '') ; iNum :=***StrToInt***(InputBox('Num', '', ''));
	Any reference to a ***sum*** of values or ***count***ing values 🛠 Make Sum or Count variable 🛠 Initialise variable (set to 0) 🛠 Add value onto Sum OR increase Count 🛠 Display Sum or Count variable	rSum := 0 ; iCount := 0 ; ***rSum*** := ***rSum*** + **VALUE** ; ***Inc***(iCount) ; Showmessage(FloatToStr(rSum)) ;
	Any reference to an ***average*** of values 🛠 Do everything (both sum and count) as described in scenario above 🛠 Calculate average into real variable (Sum / count) 🛠 Display average variable	rAverage := rSum / iCount ; redDisplay.Lines.Add(FloatToStr(rAverage)) ;
	Any display of a variable in a box like below:  🛠 Use ***ShowMessage*** procedure 🛠 Convert number variables to string	***ShowMessage***('The answer is ' + IntToStr(iAnswer)) ;

Section	Scenario in Exam	Example or Resources Links
General Programming Tips (cont)	Displaying real (decimal) numbers to specific decimal places 🛠 Use ***FloatToStrF*** (note F at end) 🛠 First parameter is real variable to display 🛠 Second variable is ***ffCurrency*** if dealing with money values, all others use ***ffFixed***. 🛠 Third parameter use 8. 🛠 Forth parameter is number of decimal places to display (Example 2 for 2 decimal places)	redDisplay.Lines.Add(***FloatToStrF***(rAverage , *ffFixed* , 8, *1*)) ; ShowMessage(***FloatToStrF***(rPrice , *ffCurrency* , 8, *2*)) ;
Common Functions used	Finding if X is a factor of Y (Use ***MOD***)	if X ***MOD*** Y ***= 0*** then
	Remove decimal values (Use ***ROUND*** / ***TRUNC***)	iNum := ***TRUNC***(rSum / iCount) ;
	Remove whole number and keep decimal (Use ***FRAC***)	rValue := ***FRAC***(rSum) ;
	Generate a random number from X to Y 🛠 Add Math under uses 🛠 Use ***RandomRange*** with low and high values separated by comma. 🛠 Add one to high value.	iNum := ***RandomRange***(X , Y + 1) ; *//Dice roll* iThrow := ***RandomRange***(1, 7) ;
	"Round up" to nearest integer (Use ***CEIL***) "Round down" to nearest integer (Use ***FLOOR***)	iNum := ***CEIL***(rAnswer) ; iNum := ***FLOOR***(rAnswer) ;
	Use POWER from X to power of Y Use SQR to square and SQRT to squareroot	rAnswer := ***POWER***(X, Y) ; rAns := ***SQRT***(iNum) ;

Text Files	Any reference to ***READING*** from a text file:
	🛠 Declare
	var F : TextFile ;
	sLine : string ;
	🛠 If ***FileExists***('name of textfile') =FALSE then
	begin
	ShowMessage('File not found') ;
	Exit ;
	end ;
	🛠 ***AssignFile***(F, 'name of textfile') ;
	🛠 ***Reset***(F) ;
	🛠 While NOT ***eof***(F) do
	begin
	readln(F, sLine) ;
	//work with sLine here
	end ;
	🛠 ***CloseFile***(F) ;
		LEARN THIS ALGORITHM:
		Watch video for explanation
		https://youtu.be/wheQ-Vjmqk4
		
		Another example
		https://youtu.be/7MZ1dQuOc2c

Section	Scenario in Exam	Example or Resources Links

Text Files (cont)	Any reference to ***WRITING*** to a text file:
	🛠 Declare
	var F : TextFile ;
	sLine : string ;
	🛠 ***AssignFile***(F, 'name of textfile') ;
	🛠 ***Rewrite***(F) ;
	//Use rewrite ONLY if creating a new text file
	🛠 ***Append***(F) ;
	//Use append ONLY if added onto an exisiting text file. (Use Rewrite OR Append)
	🛠 sLine := //what you want to put into textfile ;
	🛠 ***WriteIn***(F, sLine) ;
	🛠 ***CloseFile***(F) ;
		LEARN THIS ALGORITHM:
		Watch video for explanation
		https://youtu.be/FecvQ_q8aqA

Strings separated by a symbol
NOTE: often used with reading from a text file (use the ***sLine*** variable mentioned on previous page)
	When you see something like ***data#data#data***
	(Example: Smith#John#45)
	sLine := 'Smith#John#45' ;
	🛠 iHash := pos('#' , sLine) ;
	🛠 sSurname := Copy(sLine, 1, iHash – 1) ;
	🛠 Delete(sLine, 1, iHash) ;
	//repeat the process for Name
	🛠 iHash := pos('#' , sLine) ;
	🛠 sName := Copy(sLine, 1, iHash – 1) ;
	🛠 Delete(sLine, 1, iHash) ;
	//last section is what is left in sLine
	🛠 iCode := StrToInt(sLine) ;
		LEARN THIS ALGORITHM:
		Watch video for explanation
		https://youtu.be/_okjwXswqKY
		
		Another example:
		https://youtu.be/TN7bbEDRlnY

Databases (NOT SQL)	Any reference to ADDING or INSERTING a new record: *NOTE: change **tblTableName** to questions table name* 🛠 tblTableName.***INSERT*** ; 🛠 tblTableName['FieldName'] := value ; //FieldName in quotes must be exactly as is in database. Value must be value you inserting for that field. Repeat this step for all OTHER values. 🛠 tblTableName.***POST*** ;	LEARN THIS ALGORITHM: Watch video for explanation https://youtu.be/L3GQPGfnuW8
	Any reference to CHANGING or EDITING a record: *NOTE: change **tblTableName** to questions table name* 🛠 tblTableName.***EDIT*** ; 🛠 tblTableName['FieldName'] := value ; //FieldName in quotes must be exactly as is in database. Value must be value OR formula you changing field to. Repeat this step if any other fields must be changed. 🛠 tblTableName.***POST*** ;	LEARN THIS ALGORITHM: Watch video for explanation https://youtu.be/93QZ9gLWQ_Q
	Any reference to REMOVING or DELETING a record: *NOTE: may need to use search algorithm, to find record* 🛠 tblTableName.***DELETE*** ;	LEARN THIS ALGORITHM: Watch video for explanation https://youtu.be/8UzyzueGm5Y

Section	Scenario in Exam	Example or Resources Links

Databases (NOT SQL) (cont)	Any other example that requires going through all the values in a table:
	🛠 tblTableName.***FIRST*** ;
	🛠 while NOT tblTableName.***EOF*** do
	Begin
	//work with fields for each record here
	🛠 tblTableName.***NEXT*** ;
	End ;
	
	Any reference to a field use:
	tblTableName['*FieldName*']
	where *FieldName* is exactly as name is in table
		LEARN THIS ALGORITHM:
		Watch video for explanation
		https://youtu.be/yD-PfGmToRo
		
		Another example:
		https://youtu.be/pw9NV7KutKI

Databases (SQL ONLY)	Each question will contain a ADOQuery component with either an empty string or the text 'Add SQL text here' REPLACE that with your SQL statement.	qryTemp.Text := ***' Add SQL text here '*** ;
	Any SQL statement that queries or displays a set of results: *SELECT* ***<column names>*** *FROM* ***<table name>*** *WHERE* ***<criteria>*** **Column names** must be exactly as in database table. **Table name** is name of table with column names. **Criteria** is Field operator Value Example: Grade >= 10	Watch these videos: Basics Part 1 https://youtu.be/ZgNdtFp4yME Basics Part 2 – https://youtu.be/ujZDLuuqya8 Basics Part 3 – https://youtu.be/9ESzJXPJVaA Basics Part 4 – https://youtu.be/OtytUq8Z-mM
	Any SQL statement that CHANGES or UPDATES a record: *UPDATE* ***<table name>*** *SET* ***<field name> = <value>*** *WHERE* ***<criteria>***	LEARN THIS ALGORITHM: Watch video for explanation https://youtu.be/EfwPzx25JPQ
	Any SQL statement that REMOVES or REMOVES a record: *DELETE* ***<table name>*** *WHERE* ***<criteria>***	LEARN THIS ALGORITHM: Watch video for explanation https://youtu.be/Tt4fQ6k0VoA
	Any SQL statement that ADDS or INSERTS a record: *INSERT INTO* ***<table name>*** ***(<field name> , <field name> ,…)*** *VALUES (* ***<value> , <value> , …)***	LEARN THIS ALGORITHM: Watch video for explanation https://youtu.be/wlhSsIDXh6A

<table>
<tr><th>Section</th><th>Scenario in Exam</th><th>Example or Resources Links</th></tr>
<tr><td rowspan="4">Object Oriented Programming</td><td>Any reference to a constructor:
Write a parameterised constructor accepting a parameter for the team's name and wins. All the other (integer) fields must be set to 0 by default.

constructor Create(sTeam: string ; iWins: integer);
begin
fTeam := sTeam ;
fWins := iWins ;
fLosses := 0 ;
fPoints := 0 ;
end;</td><td rowspan="3">LEARN THIS ALGORITHM:
Watch video for explanation
https://youtu.be/OuMu4dgJZ8M

Another example:
https://youtu.be/VBGu-stH6xk</td></tr>
<tr><td>Any reference to a accessor:
Write an accessor method for the name of the team (fTeam).

function GetTeam: string;
begin
result := fTeam ;
end;</td></tr>
<tr><td>Any reference to a mutator:
Write a mutator method that takes in a team name as a parameter for the name of the team (fTeam).

procedure SetTeam (sName : string);
begin
fTeam := sName ;
end;</td></tr>
<tr><td>Any reference to a toString:
function toString: string;
🛠 result := ' Text ' + private field + ….
// tabs = #9
// new lines = #13
//convert any numbers using IntToStr / FloatTostr
//Add Uses SysUtils ; at top of object under name</td><td>Example of toString</td></tr>
<tr><td>Arrays
(Do this only once you are comfortable with everything covered above)</td><td>Know the following algorithms:
🛠 Displaying the contents of an array
🛠 Selection Sort
🛠 Aggregate Functions
🛠 2D Arrays</td><td>https://youtu.be/fIcDET8nMpI
https://youtu.be/gpu04MS8pJY
https://youtu.be/gBmL92SCxSU
https://youtu.be/lCepY3luREc</td></tr>
<tr><td colspan="3">If you are looking for more video content:
🔑 Go to YouTube and search for Mr Long IT and CAT
🔑 Then go to playlists and search for relevant exam paper or topics
🔑 All videos have links to all videos and to data files in the Description</td></tr>
</table>