



**higher education  
& training**

Department:  
Higher Education and Training  
**REPUBLIC OF SOUTH AFRICA**

**NATIONAL CERTIFICATE (VOCATIONAL)**

**COMPUTER PROGRAMMING**

(Second paper)

**NQF LEVEL 4**

(10041024)

**4 December 2020 (X-paper)**

**09:00–13:00**

**This question paper consists of 12 pages.**

291Q2N2004

<p><b>TIME: 4 HOURS</b> <b>MARKS: 100</b></p>
---

---

## **INSTRUCTIONS AND INFORMATION**

1. Answer all the questions.
2. Read all the questions carefully.
3. Number the answers according to the numbering system used in this question paper.

## **INSTRUCTIONS TO STUDENTS AND LECTURERS**

1. Students **MUST** be seated in the computer laboratory 30 minutes before the start of the examination to ensure that all computers have the appropriate programs.
2. These questions must be completed on a computer by making use of MS Office 2010 or a later version and VB.NET 2005 or VB.NET Express or VB.NET 2008 or a later version.
3. Students must make sure that they print their work immediately after completing a question. They must ensure that their examination number appears on each printout. No questions without a printed **EXAMINATION NUMBER** will be marked.
4. In the event of problems such as a power failure, computer breakdown or a printer breakdown, the invigilator will make the necessary arrangements to continue with the examination process once the problem has been resolved. Students will not be penalised with time lost under these circumstances.
5. All work on the computer should be saved at regular intervals to prevent loss of work. **NO** additional time may be allowed for work lost due to lack of saving.
6. No student may print his/her work for another student or make a memory stick available to another student or access another student's work on the network. Any attempt to access any information from or transfer information to another student in whatever manner is a contravention of the examination rules and regulations and will be viewed in an extremely serious light.
7. All **PRINTOUTS** or screen prints done during the session **MUST** be handed in.

8. Students may not copy the source code from VB.NET to MS Word and then print it from the word processor.
  9. Steps to print a VB.NET form or any other screen:
    - 9.1 Run the solution.
    - 9.2 Hold the ALT key down and press the print screen button. This will copy the active screen.
    - 9.3 Open MS Word and paste the form/screen in a new document.
    - 9.4 Add the examination number to the top right-hand side of the page.
    - 9.5 Print the page.
  10. Students are allowed to make use of the help files in VB.NET.
  11. The following files will appear on the hard drive in the folders:
    - 11.1 QUESTION 1  
C:\CP L4 P2\ FuelCalculator  
FuelCalculator (Microsoft Visual Studio Solution)
    - 11.2 QUESTION 2  
C:\CP L4 P2\Databases\  
C:\ CP L4\ Leaveregister.mdb  
LeaveregisterSystem (Microsoft Visual Studio Solution)
    - 11.3 QUESTION 3  
C:\CP L4 P2\QUESTION 3  
House.jpg  
Style.css
  12. Students may use the following or any other program of their choice to create webpages:  
  
Notepad ++, Netbeans, Blue voda, Dreamweaver
-

**QUESTION 1**

The following questions refer to the VB.NET solution *FuelCalculator* (\CP P2 \QUESTION 1\FuelCalculator)



You are tasked to design an application for EnergyEfficiency customers to calculate the cost of fuel for their vehicles. Customers must enter the litres and the type of fuel the vehicle uses.

The price per litre for the different fuels are:

Diesel = R15,50

Unleaded 93 = R15,72

Unleaded 95 = R16,49

Below is the IPO chart for the problem.

Input	Processing	Output
Litre fueltype	Determine fuel Calculate fuel cost $\text{price} = \text{type} \times \text{litres}$ Determine vat $\text{totalcost} = \text{price} + (\text{vat} \times \text{price})$	Type Litre Price totalcost

FIGURE 1 is a printout of what the user interface looks like:

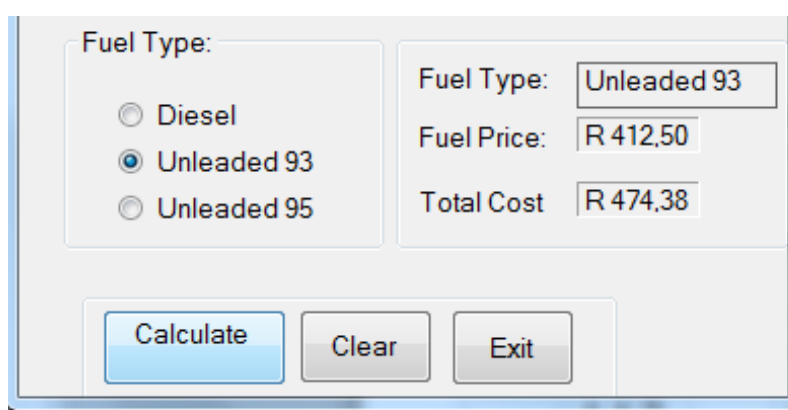


**FIGURE 1**

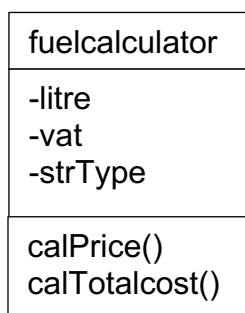
STEP 1 Run the program and enter litres of fuel for a vehicle.



STEP 2 Click on the Calculate button to calculate and display the fuel price and the total cost.

Study the class diagram below for the salary class and answer the questions.



The completed interface has been saved on the hard drive. Open the solution *FuelCalculator* and follow the instructions to complete the program.

1.1 Add the following program code to the fuelcalculator class.


1.1.1 Create the class with all the private fields.

1.1.2 Create the public property for all the fields.



(2 × 2) (4)


1.2 Create a public function calPrice() to calculate the price of the fuel per litre. (3)

- 1.3 Create a public function `calTotalcost()` to calculate the total cost including VAT of 15% for fuel. The `calTotalcost()` method must invoke the `calPrice()` method of the class. (4)
- 1.4 Add the following program code to the click event of the Calculate button on the *FuelCalculator* form.
- 1.4.1 Write the program code to instantiate an object from the *FuelCalculator* class.  (2)
- 1.4.2 Assign the values from the text boxes to the appropriate properties. (2)
- 1.4.3 Write the program code to display the fuel type, price and total cost as shown above. (3)
- 1.4.4 Invoke the `calPrice()` method of the *FuelCalculator* class to display the calculated value in the appropriate control. (3)
- 1.4.5 Invoke the `calTotalcost()` method of the salary class. The total cost should be calculated and displayed in the appropriate control. (2)

Add your EXAMINATION NUMBER to the program code of the *FuelCalculator* form.  
Make a printout of the program code of the main form.

- 1.5 Change the text in the title bar of the form to your EXAMINATION NUMBER.

Run the program and enter the test data below. Calculate the result and make a PRINTOUT of the form.



Fuel	Litres
Unleaded95	25
Unleaded93	35

(2)  
[25]

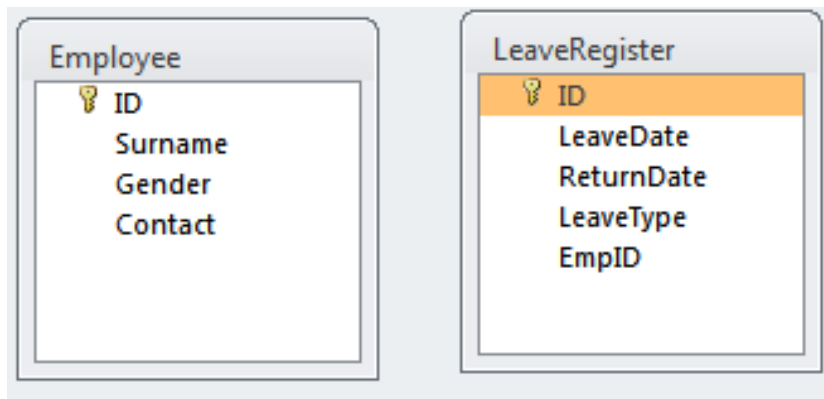
## QUESTION 2


The following question refers to the VB.NET solution Leave Register (\CP P2\QUESTION 2\LeaveRegisterSystem) that has been saved on the hard drive.

You must create an application for the HR Department which will enable the administrator to keep track of an employee's leave. A database called *LeaveRegister.mdb* with related data-tables has been created and populated with data. It is located in the folder called *C:\CP L4 P2\Learners\_Extra\_Classes*.

Below is the graphical representation of the two tables. 

Study the tables below and answer the questions.



- 2.1 Create a one to many relationships between the two tables, *Employee* and *LeaveRegister*. 

Make a printout of the relationship window.

(3)

The steps below explain how to use the program to insert records into the *LeaveRegister* database.

STEP 1: Run the program.

**LEAVE**

**HR Leave Register:**

EmployeeID

Leave Date: 10 August ▼

Return Date: 10 August ▼

**Type of Leave**

☐ Sick Leave

☐ Speacial Leave

☐ Annual Leave

**Process Leave** **Exit**



STEP 2: Click on the Process Leave button to insert the record into the *LeaveRegister* table.



The completed interface has been saved on the hard drive. Open the solution *LeaveRegisterSystem* (\QUESTION 2\ *LeaveRegisterSystem*) and follow the instructions to complete the program.

**NOTE:** You are required to write the program code and SQL statements to complete the question. You may NOT use the wizard to complete the question.



Add the following program code to the click event of the Process Leave button to insert employees' leave records into the database table *LeaveRegister* (STEP 2). Add *try ... catch* to catch exceptional errors.

2.2 Add your EXAMINATION NUMBER as comment to the program. Write the program code to establish a connection to the database to insert an employee's leave record. (3)

2.3 Write a SQL statement for the following:

2.3.1 Determine which leave type is selected and return the type of leave. (5)



2.3.2 To insert leave records for employees into the database table *LeaveRegister*. (12)



- 2.4 Run the program and enter the following test data. Make a PRINTOUT of the running program.

IdentityNumber	leavedate	returndate	type
7505010660082	2017/02/06	2017/02/14	sick
9501305977082	2017/03/20	2017/03/27	sick
8010205544083	2017/03/22	2017/03/29	special
8808125612082	2017/05/05	2017/06/05	annual



(2)  
[25]

### QUESTION 3

The image in FIGURE 2 is a screenshot of a webpage belonging to 'My House My Home'. It allows a user to send a request for affordable accommodation depending on whether the customer wants to buy or rent a house and any additional requirements they may have. You are required to reproduce the page as shown by answering the questions:



## My House My Home

[HOME](#) | [GALLERY](#) | [PROPERTIES](#) | [CONTACT US](#) | [ABOUT US](#)



### Affordable Accomodation

Email:

Number of Bedrooms

Do you want to Buy or Rent




☐ Buy ☐ Rent

Do you need additional requirements?

☐ Garden ☐ Swimming Pool ☐ Entertainment

Insert Examination Number

FIGURE 2

- 3.1 Open the external style sheet called style.css. Insert your EXAMINATION NUMBER as a comment in the style sheet.
- 3.1.1 Add a rule set for a link element. Set the text decoration to none. (2)
- 3.1.2 Create a rule set for the navigation division. Set the height to 30px, the text alignment to text centre, the font weight to bold and the background colour to #ccc.  (3)
- 3.2 Open a new text document in the editor of your choice and save it as index.html. Create all divisions inside the container.
- 3.2.1 Add a link to the external cascading style sheet called style.css. (1)
- 3.2.2 Create the header division. Insert the heading 'My House My Home'. Make use of heading 1 element. (2)
- 3.2.3 Create the navigation division. Insert the hyperlinks Home.html, Gallery.html, Properties.html and Contact Us.html.  (2)
- 3.2.4 Create the sidebar division. Insert the image *House.jpg*, set to a width of 140px and height of 300px saved in the folder called images. (2)
- 3.2.5 Inside the main division of the webpage, create a borderless table element with 5 rows and 2 columns. (2)
- 3.2.6 Create a new form element and name it "registration". (1)
- 3.2.7 Insert the text 'Affordable Accommodation' as a H2 heading element and add a horizontal rule. (3)
- 3.2.8 Add a text field for the email address.  (2)
- 3.2.9 Add a drop-down list with the select options as shown in the figure below:

Email:

Number of Bedrooms ▼

Number of Bedrooms

One

Two

Three

Four

(3)

- 3.2.10 Create a radio button group to display the options as in the figure below:



Do you want to Buy or Rent

☐ Buy ☐ Rent

(2)

- 3.2.11 Create a check box group with the options garden, swimming pool and entertainment, as in the figure below.

Do you need additional requirements?

☐ Garden ☐ Swimming Pool ☐ Entertainment

Request Clear Text

(2)

- 3.2.12 Add the “Request” button with an event handler function called *check()* to validate the email address content.



(2)

- 3.2.13 Add the button “Clear Text” to clear the fields for data entry.

(1)  
[30]

#### QUESTION 4

Add the necessary JavaScript code to the webpage in QUESTION 3 that will do the following:

The user enters information and then clicks the “Request” button which invokes the *check()* function to determine if the content in the email text field is a valid email address. An email address will be considered valid if it contains an “@” sign and at least one “.”.

- 4.1 Add a script element to the webpage and indicate that JavaScript will be used.



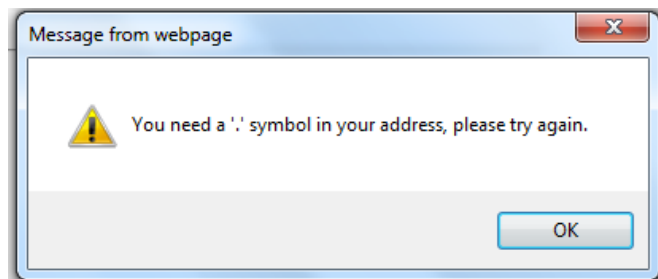
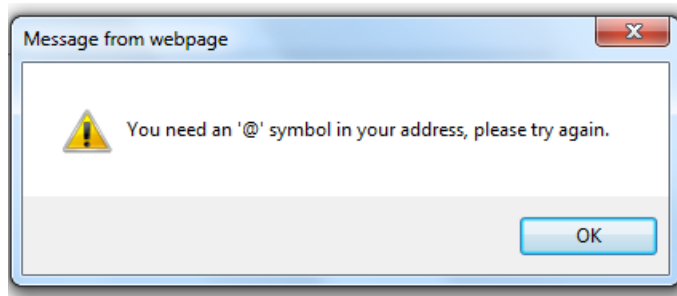
- 4.1.1 Create a JavaScript function called *check()* (2)

- 4.1.2 Declare a variable inside the *check()* function and assign it the value of the email text field. (4)

- 4.1.3 Make use of an *IF* statement to determine which character is present in the email address. Display appropriate messages as shown below.

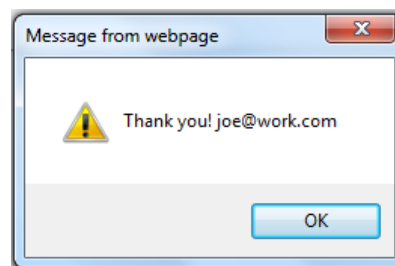


Validate the email content in the text field as valid or not by checking that the @ symbol and period(.) are contained in the email address.



(10)

- 4.2 Determine if the email contents are valid and display the message as shown.



(4)

Make a PRINTOUT of the source code.  
Make a PRINTOUT of the browsed webpage.



[20]

**TOTAL: 100**