basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

SENIORSERTIFIKAAT-EKSAMEN/
NASIONALE SENIORSERTIFIKAAT-EKSAMEN

INLIGTINGSTEGNOLOGIE V1

MEI/JUNIE 2026

PUNTE: 150

TYD: 3 uur

**Hierdie vraestel bestaan uit 26 bladsye, 2 bladsye met data,
2 bladsye vir beplanning en 'n aparte inligtingsblad.**

Kopiereg voorbehou

INSTRUKSIES EN INLIGTING

1. Hierdie vraestel is in VIER afdelings ingedeel. Kandidate moet AL die vrae in AL VIER afdelings beantwoord.

2. Twee leë bladsye wat vir beplanningsdoeleindes gebruik kan word, is aan die einde van die vraestel ingesluit.

3. 'n Inligtingsblad is voorsien wat jy aan die einde van die eksamensessie moet voltooi. Maak seker dat AL die inligting wat jy verskaf het, korrek is en lewer die inligtingsblad in voordat jy die eksamenlokaal verlaat.

4. Die duur van hierdie eksamen is drie uur. As gevolg van die aard van hierdie eksamen is dit belangrik om daarop te let dat jy nie toegelaat sal word om die eksamenlokaal voor die einde van die eksamensessie te verlaat nie.

5. Hierdie vraestel is opgestel met programmeringsterme wat kenmerkend vir Delphi as programmeringstaal is. Die Delphi-programmeringstaal moet gebruik word om die vrae te beantwoord.

6. Maak seker dat jy die vrae beantwoord volgens die spesifikasies wat by elke vraag gegee word. Punte sal volgens die voorgeskrewe vereistes toegeken word.

7. Beantwoord slegs wat by elke vraag gevra word. Byvoorbeeld, indien die vraag nie vir datavalidering vra nie, sal geen punte vir datavalidering toegeken word nie.

8. Jou programme moet op só 'n manier gekodeer word dat dit met enige data sal werk en nie net met die voorbeelddata wat voorsien is of enige data-uittreksels wat in die vraestel verskyn nie.

9. Roetines, soos soek, sorteer en seleksie, moet vanuit eerste beginsels ontwikkel word. Jy mag NIE die ingeboude funksies van die Delphi-programmeringstaal vir enige van hierdie roetines gebruik NIE.

10. Alle datastrukture moet deur jou, die programmeerder, verklaar word, tensy die datastrukture verskaf word.

11. Jy moet jou werk gereeld stoor op die disket/CD/DVD/geheuestokkie wat aan jou gegee is of op die skyfspasie wat vir hierdie eksamensessie aan jou toegeken is.

12. Maak seker dat jou eksamennommer as kommentaar verskyn in elke program wat jy kodeer, asook op elke gebeurtenis wat aangedui word.

13. Indien dit vereis word, druk die programmeringskode van al die programme/ klasse wat jy voltooi het. Jou eksamennommer moet op alle drukstukke voorkom. Jy sal ná die eksamensessie 'n halfuur tyd vir drukwerk gegee word.

14. Aan die einde van hierdie eksamensessie moet jy 'n disket/CD/DVD/ geheuestokkie inlewer met al jou werk daarop gestoor OF jy moet seker maak dat al jou werk op die skyfspasie gestoor is wat vir hierdie eksamensessie aan jou toegeken is. Maak seker dat al die lêers gelees kan word.

15. Die lêers wat jy benodig om hierdie vraestel te voltooi, is aan jou gegee op 'n disket/CD/DVD/geheuestokkie of op die skyfspasie wat aan jou toegeken is. Die lêers word in die vorm van wagwoordbeskermde uitvoerbare lêers verskaf.

    Doen die volgende:

    - Dubbelklik op die volgende uitvoerbare lêer wat met 'n wagwoord beskerm is: **IT P1 MayJune 2026 DATA.exe**
    - Klik op die 'Extract'-knoppie.
    - Sleutel die volgende wagwoord in: **J26@Sea#**

    Nadat dit onttrek ('extracted') is, sal die volgende lys lêers in die lêergids ('folder') **IT P1 MayJune 2026 DATA** beskikbaar wees:

Question 1:
Question1_P.dpr
Question1_P.dproj
Question1_P.res
Question1_U.dfm
Question1_U.pas

Question 2:
DBConnect_U.pas
MarineDB - Copy.mdb
MarineDB.mdb
Question2_P.dpr
Question2_P.dproj
Question2_P.res
Question2_U.dfm
Question2_U.pas

Question 3:
Populate_U.pas
Question3_P.dpr
Question3_P.dproj
Question3_P.res
Question3_U.dfm
Question3_U.pas
SeaTurtle_U.pas

Question 4:
Fishing.txt
Question4_P.dpr
Question4_P.dproj
Question4_P.res
Question4_U.dfm
Question4_U.pas

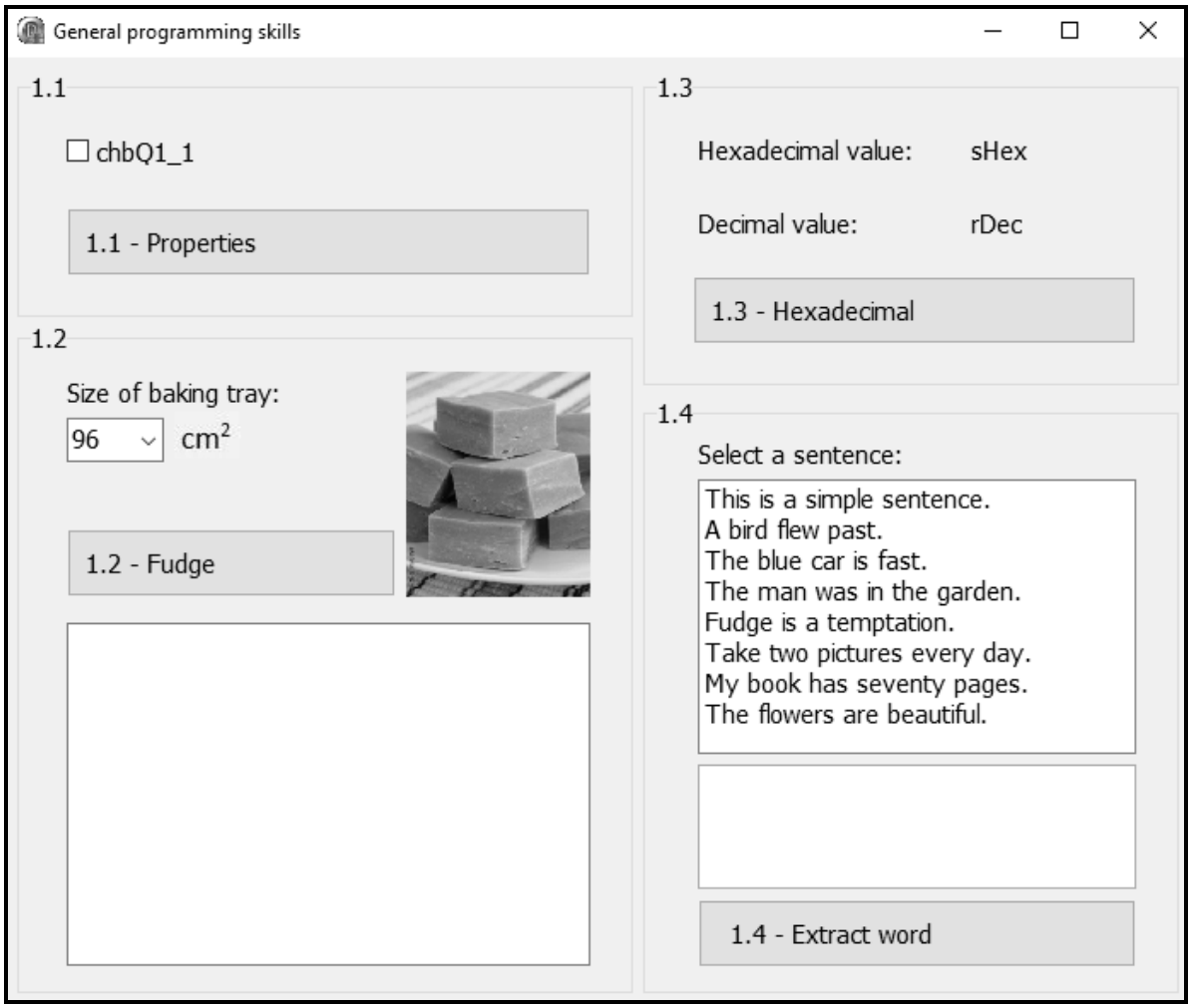
AFDELING A

VRAAG 1: ALGEMENE PROGRAMMERINGSVAARDIGHEDE

Doen die volgende:

- Maak die onvolledige program in die **Question 1**-lêergids oop.
- Sleutel jou eksamennommer as kommentaar in die eerste reël van die **Question1_U.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans geen funksionaliteit nie.

Voorbeeld van grafiese gebruikerskoppelvlak (GGK ('GUI')):



- Voltooi die kode vir elke afdeling van VRAAG 1, soos wat in VRAAG 1.1 tot VRAAG 1.4 op die volgende bladsy beskryf is.

1.1 **Knoppie [1.1 - Properties]**

Skryf kode om die volgende veranderinge aan die kontroleblokkie('check box')-komponent **chbQ1_1** aan te bring:

- Stel die opskrif ('caption') op 'Marine World'.
- Maak seker dat die kontroleblokkie gemerk is ('ticked)'.
- Stel die font op vetdruk ('bold').
- Maak die kontroleblokkie onaktief ('disable').

Voorbeeld van afvoer:



(4)

1.2 **Knoppie [1.2 - Fudge]**

Fudge, wat 'n soort lekkergoed is, moet vir 'n fondsinsamelingsgeleentheid berei/gemaak word. Die volgende voorwaardes is tydens die bereiding en verpakking van die fudge van toepassing:

- Daar is drie verskillende groottes bakplate beskikbaar, naamlik 96 cm^2, 144 cm^2 en 400 cm^2.
- Elke stukkie fudge moet 4 cm^2 groot wees.
- Elke pakkie moet ses stukkies fudge bevat.

Die gebruiker moet die grootte van die bakplaat wat gebruik gaan word, uit kombinasieblokkie ('combo box') **cmbQ1_2** selekteer.

Kode is voorsien om die **memQ1_2**-afvoerarea skoon te maak**.**

Die volgende vier heelgetalveranderlikes is voorsien en moet in jou oplossing gebruik word:

```
iArea: Integer;
iNumPieces: Integer;
iPackets: Integer;
iExtra: Integer;
```

Skryf kode om die volgende te doen:

- Onttrek ('Extract') die grootte van die bakplaat uit kombinasieblokkie **cmbQ1_2**.
- Bepaal die totale getal stukkies fudge wat uit die bakplaat gesny kan word.
- Bepaal die getal pakkies fudge wat uit die totale getal stukkies fudge voorberei kan word.

- Bepaal die getal oorblywende stukkies fudge wat nie genoeg sal wees om 'n pakkie vol te maak nie.
- Vertoon die totale getal stukkies fudge, die getal pakkies fudge en die getal oorblywende stukkies fudge in die memo-komponent **memQ1_2**, in die formaat wat hieronder getoon word:

```
Total pieces of fudge: <getal stukkies fudge>
Packets of fudge: <getal pakkies fudge>
Remaining pieces of fudge: <getal oorblywende stukkies fudge>
```

Voorbeeld van afvoer as die grootte bakplaat wat geselekteer is, 96 cm^2 is:

```
Total pieces of fudge: 24
Packets of fudge: 4
Remaining pieces of fudge: 0
```

Voorbeeld van afvoer as die grootte bakplaat wat geselekteer is, 400 cm^2 is:

```
Total pieces of fudge: 100
Packets of fudge: 16
Remaining pieces of fudge: 4
```

(9)

1.3 **Knoppie [1.3 - Hexadecimal]**

'n Heksadesimale waarde wat deur die gebruiker ingesleutel word, moet na 'n desimale waarde omgeskakel word. Validering van die toevoerwaarde sal NIE vereis word NIE.

Die volgende veranderlikes is voorsien:

```
sHex : String;
rDecimal : Real;
iExp : Integer;
```

Skryf kode om die volgende te doen:

- Gebruik 'n toevoerblokkie ('input box') om 'n heksadesimale waarde in te sleutel en stoor die waarde in die veranderlike **sHex**.
- Vertoon die waarde wat in **sHex** gestoor is in die etiket ('label') **lblQ1_3Hex**.
- Gebruik die algoritme op die volgende bladsy om die heksadesimale waarde wat in **sHex** gestoor is, na 'n desimale waarde om te skakel:

Inisialiseer:
- Stel veranderlike **rDecimal** op 0.
- Stel veranderlike **iExp** op 0.

Herhaling:
Vir elke karakter van die heksadesimale waarde, vanaf die laaste karakter tot die eerste karakter:
- Onttrek die karakter uit die heksadesimale waarde.
- Gebruik 'n CASE-stelling om die heelgetalwaarde van die karakter te bepaal deur die volgende voorwaardes te gebruik:
  - As die karakter '0'-'9' is, skakel die datatipe van die karakter na 'n heelgetalwaarde om en stoor die heelgetalwaarde.
  - As die karakter 'A'-'F' is, stoor die spesifieke heelgetalwaarde, op die volgende gebaseer: A=10, B=11, C=12, D=13, E=14, F=15
- Bereken 16, wat verhef is tot die mag **iExp**, 16^{iExp}, deur die tot-die-mag('power')-funksie te gebruik.
- Vermenigvuldig die karakter se heelgetalwaarde met die resultaat van 16^{iExp}.
- Voeg die resultaat van die vorige stap by **rDecimal**.
- Vermeerder die **iExp**-veranderlike met die waarde van 1.

Afvoer:
- Vertoon die waarde van **rDecimal** in etiket **lblQ1_3Dec.**

Voorbeeld van afvoer as die heksadesimale getal 1A ingesleutel is:

Hexadecimal value:	1A
Decimal value:	26

Voorbeeld van afvoer as die heksadesimale getal A2B ingesleutel is:

Hexadecimal value:	A2B
Decimal value:	2603

(14)

1.4 **Knoppie [1.4 - Extract word]**

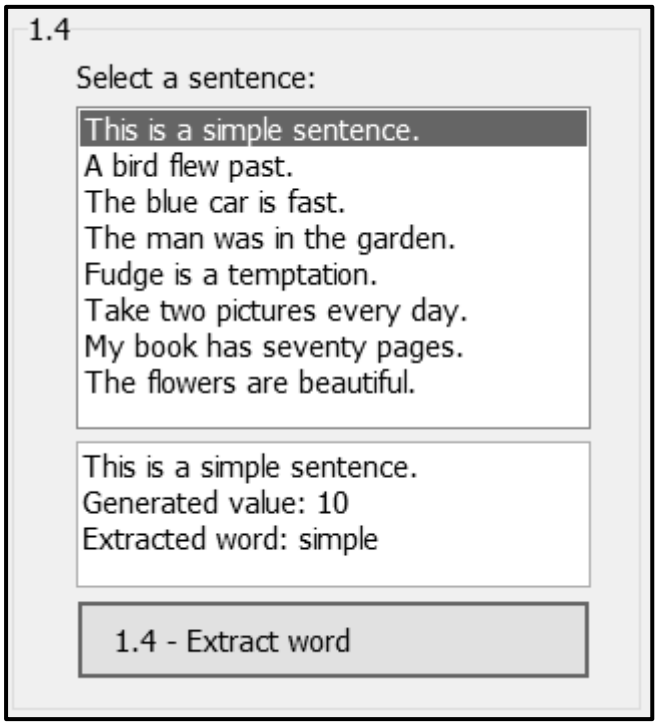
Lysblokkie ('List box') **lstQ1_4** is met verskeie sinne gevul. Die gebruiker moet 'n sin uit die lysblokkie selekteer. Om enige woord, behalwe die eerste woord, uit die sin wat geselekteer is, te onttrek, moet 'n ewekansige waarde herhaaldelik gegenereer word totdat die ewekansige getal die posisie/indeks van 'n spasie is. Die woord wat op die spasie wat geïdentifiseer is, volg, moet onttrek en vertoon word.

Kode is voorsien om komponent **redQ1_4** skoon te maak.

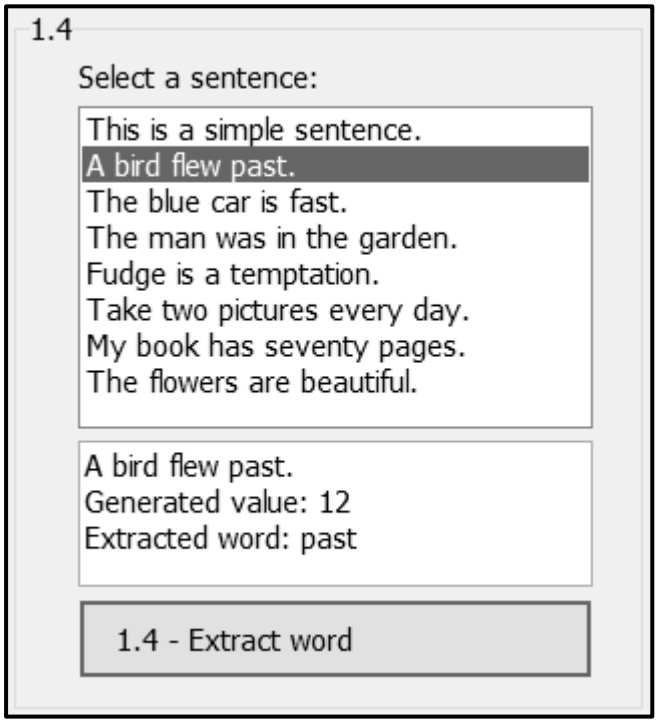
Skryf kode om die volgende te doen:

- Onttrek die geselekteerde sin uit lysblokkie **lstQ1_4**.
- Vertoon die sin in die 'rich edit'-komponent **redQ1_4**.
- Genereer herhaaldelik 'n ewekansige getal in die reeks van 1 tot die lengte van die sin totdat die karakter in die sin by die gegenereerde posisie/indeks 'n spasie is.
- Vertoon die ewekansige getal in die 'rich edit'-komponent **redQ1_4**.
- Vee al die karakters in die sin uit tot by die spasie wat geïdentifiseer is, die spasie ingesluit.
- Vind die posisie van die eerste spasie in die oorblywende deel van die sin.
  - Gebruik hierdie waarde om die eerste woord in die oorblywende deel van die sin te onttrek.
  - As die laaste woord in die sin onttrek word, maak seker dat die punt nie as deel van die woord onttrek word nie.
- Vertoon die woord wat onttrek is in die 'rich edit'-komponent **redQ1_4**.

Voorbeeld van afvoer as die eerste sin geselekteer is en as 10 die ewekansige waarde was wat gegenereer is:



Voorbeeld van afvoer as die tweede sin geselekteer is en 12 die ewekansige waarde was wat gegenereer is:



LET WEL: Jou afvoer mag anders wees as gevolg van die ewekansige waardes wat gegenereer is. (13)

- Sleutel jou eksamennommer as kommentaar in die eerste reël van die programlêer in.
- Stoor jou program.
- Druk die kode indien dit vereis word.

TOTAAL AFDELING A: 40

AFDELING B

VRAAG 2: DATABASISPROGRAMMERING

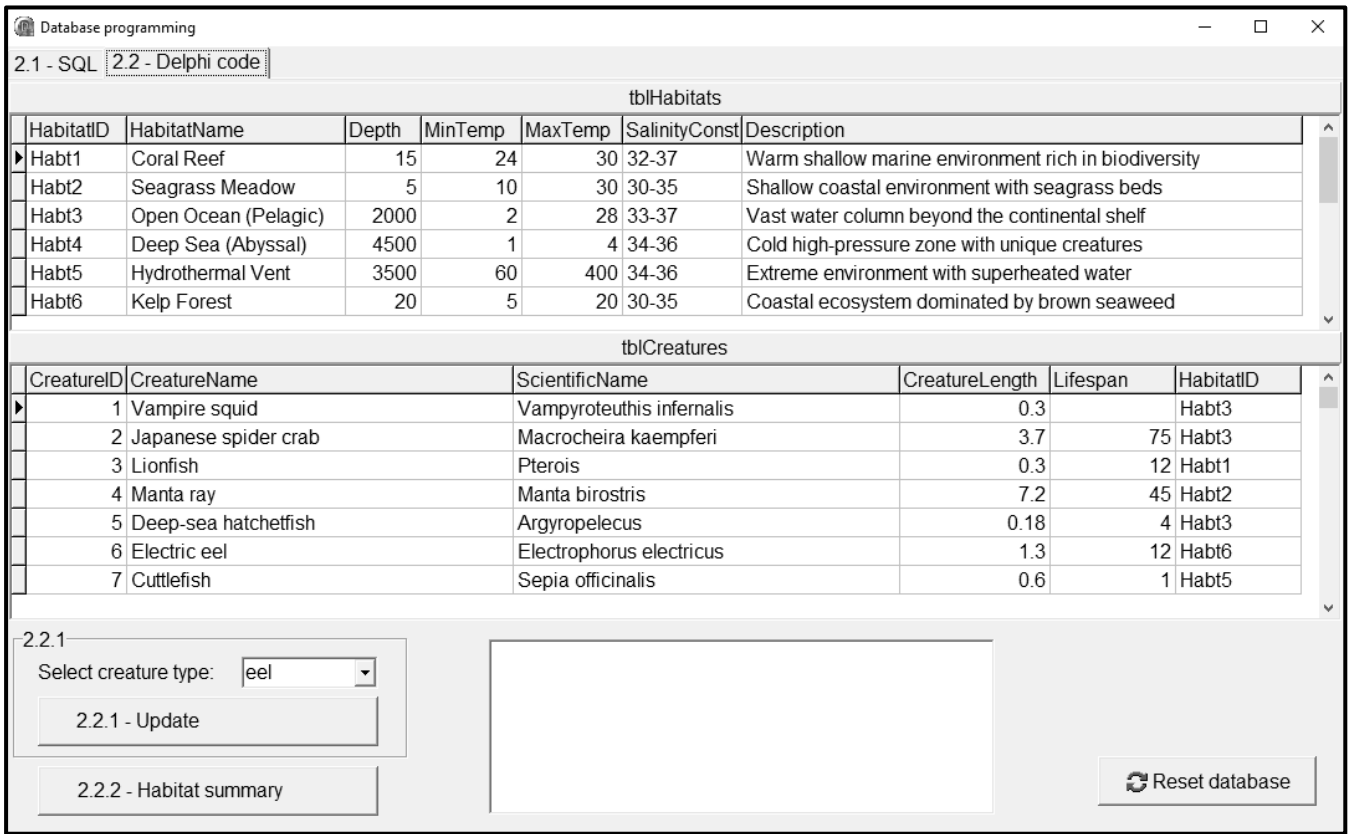
Marine World is besig om 'n reeks infografika ('infographics') vir sosiale media te skep om die aandag op mariene habitatte en seediere te vestig. Hulle het spesifieke inligting uit hulle databasis nodig om feite oor spesies, habitatte en die bewaringstatus van seelewe te vertoon.

'n Databasis met die naam **MarineDB.mdb** is ontwikkel, wat twee tabelle met die name **tblHabitats** en **tblCreatures** bevat.

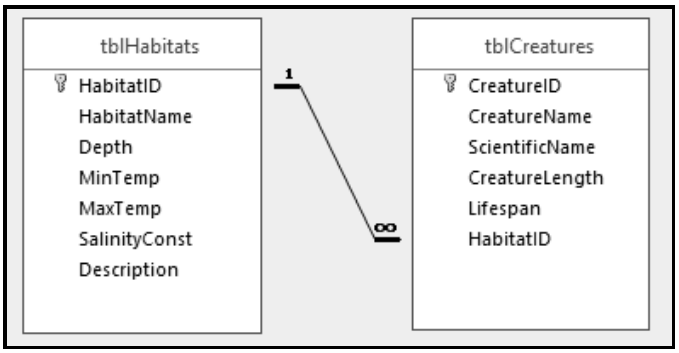
LET WEL: Die bladsye met data wat aan die einde van die vraestel aangeheg is, verskaf inligting oor die ontwerp en die inhoud van die **MarineDB**-databasis.

Doen die volgende:

- Maak die onvolledige projeklêer met die naam **Question2_P.dpr** in die **Question 2**-lêergids oop.
- Sleutel jou eksamennommer as kommentaar in die eerste reël van die **Question2_U.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans beperkte funksionaliteit. Die inhoud van die tabelle word vertoon, soos wat hieronder op die seleksie van die oortjieblad ('tab sheet') **2.2 - Delphi code** getoon word.



Die verwantskap tussen die twee tabelle **tblHabitats** en **tblCreatures** word hieronder getoon.



- Volg die instruksies hieronder om die kode vir elke afdeling wat in VRAAG 2.1 en VRAAG 2.2 beskryf word, te voltooi.
- Gebruik SQL-stellings om VRAAG 2.1 te beantwoord en Delphi-kode om VRAAG 2.2 te beantwoord.

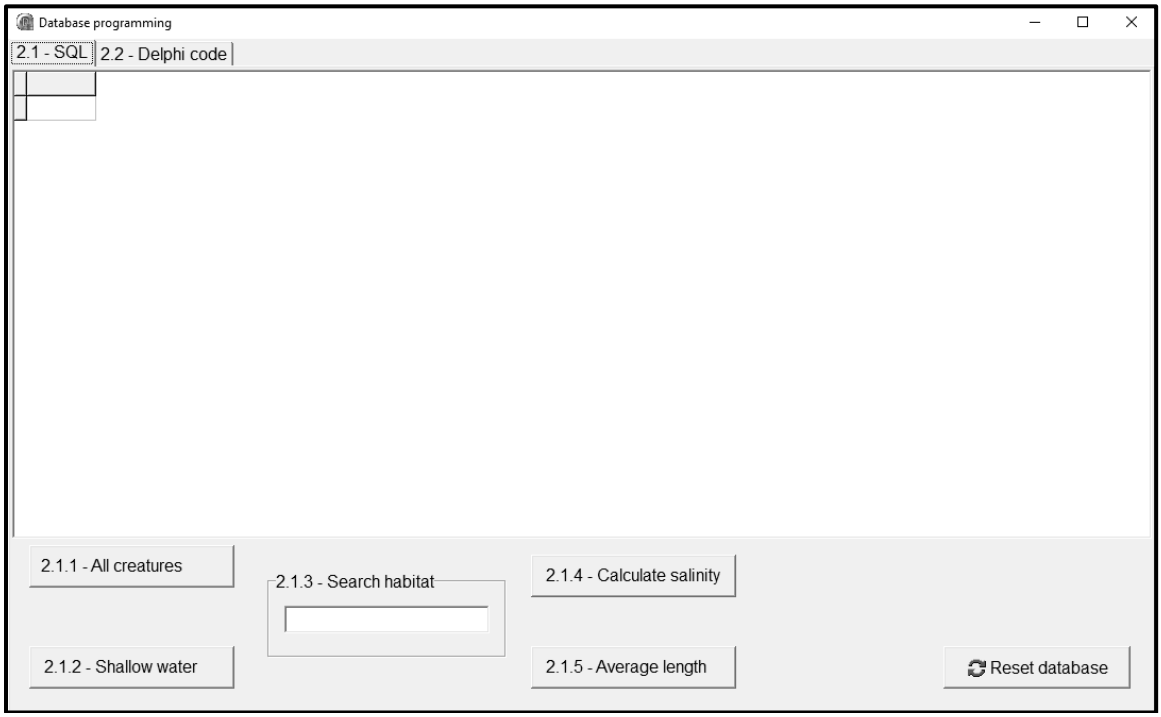
LET WEL:

- Die 'Reset database'-knoppie word voorsien om die data wat in die databasis voorkom, na die oorspronklike inhoud te herstel.
- Kode word voorsien om die GGK('GUI')-komponente met die databasis te koppel. MOENIE enige van die kode wat voorsien is, verander NIE.
- TWEE veranderlikes word as globale veranderlikes verklaar, soos in die tabel hieronder beskryf word.

Veranderlike	**Datatipe**	**Beskrywing**
`tblHabitats`	`TADOTable`	Verwys na die tabel **tblHabitats**
`tblCreatures`	`TADOTable`	Verwys na die tabel **tblCreatures**

2.1 Oortjieblad ('Tab sheet') [2.1 - SQL]

Voorbeeld van grafiese gebruikerskoppelvlak (GGK ('GUI')) vir VRAAG 2.1:



LET WEL:

- Gebruik SLEGS SQL-stellings om VRAAG 2.1.1 tot VRAAG 2.1.5 te beantwoord.
- Kode is voorsien om die SQL-stellings uit te voer en die resultate van die navrae te vertoon. Die SQL-stellings wat aan die globale veranderlike **sSQL** toegeken is, is onvolledig.

Voltooi die SQL-stellings om die take wat in VRAAG 2.1.1 tot VRAAG 2.1.5 hieronder beskryf word, uit te voer.

2.1.1 **Knoppie [2.1.1 - All creatures]**

Vertoon al die besonderhede van elke seedier in die **tblCreatures**-tabel, in alfabetiese volgorde volgens **CreatureName** gesorteer.

Voorbeeld van afvoer van die eerste agt rekords:

CreatureID	CreatureName	ScientificName	CreatureLength	Lifespan	HabitatID
30	Amerurus catfish	Nebulosus	0.25	12	Habt6
116	Amphipod	Gammarus pulex	0.9	3	Habt2
113	Anchor worm	Lernae cyprinacea	2	4	Habt1
112	Antlion	Myrmeleontidae	1.2	3	Habt2
111	Aphid	Aphidoidea Delta	0.8	1	Habt2
114	Apple fly	Rhagoletis pomonella	0.4	1	Habt6
115	Aquatic snail	Pysella acuta	1.5	5	Habt7
93	Axolotl	Ambystoma mexicanum	0.1	12	Habt6

(3)

2.1.2 **Knoppie [2.1.2 - Shallow water]**

Vertoon die **HabitatID**, **Depth** en **MinTemp**-velde waar die diepte van die water minder as 25 m is en die minimum temperatuur 10 °C of meer is.

Voorbeeld van afvoer:

HabitatID	Depth	MinTemp
Habt1	15	24
Habt7	2.5	20
Habt2	5	10

(3)

2.1.3 **Redigeerblokkie ('Edit box') [2.1.3 - Search habitat]**

Die gebruiker moet 'n gedeelte van 'n beskrywing ('**Description**') in redigeerblokkie ('edit box') **edtQ2_1_3** insleutel. Kode is voorsien om die beskrywing te onttrek en in veranderlike **sFind** te stoor.

Vertoon die **HabitatName** en **Description** van al die rekords waar die beskrywing-veld die gedeelte van die beskrywing wat in die redigeerblokkie ingesleutel is, bevat.

Voorbeeld van afvoer as 'WA' ingesleutel is:

HabitatName	Description
Coral Reef	Warm shallow marine environment rich in biodiversity
Open Ocean (Pelagic)	Vast water column beyond the continental shelf
Hydrothermal Vent	Extreme environment with superheated water
Arctic Ocean	Cold waters with seasonal ice cover

Voorbeeld van afvoer as 'Unique' ingesleutel is:

HabitatName	Description
Deep Sea (Abyssal)	Cold high-pressure zone with unique creatures

(4)

2.1.4 **Knoppie [2.1.4 - Calculate salinity]**

SalinityConst is 'n veld wat 'n stringwaarde in die formaat XX-YY stoor, waar XX die minimum soutgehalte ('salinity') en YY die maksimum soutgehalte is. Jy mag aanneem dat XX en YY altyd 'n tweesyfer-getal sal wees.

Vertoon die **HabitatID**, **SalinityConst** en 'n berekende veld met die naam **Salinity**. Die soutgehalte van die habitat word bereken deur die minimum soutgehalte van die maksimum soutgehalte in die **SalinityConst**-veld af te trek.

Voorbeeld van afvoer nadat die soutgehalte bereken is:

HabitatID	SalinityConst	Salinity
Habt1	32-37	5
Habt3	33-37	4
Habt4	34-36	2
Habt5	34-36	2
Habt6	30-35	5
Habt7	10-35	25
Habt8	32-35	3
Habt9	33-35	2
Habt2	30-35	5

(4)

2.1.5 **Knoppie [2.1.5 - Average length]**

Vertoon die **HabitatName** en die gemiddelde lengte van die seedier ('average creature length'), afgerond ('rounded') tot TWEE desimale plekke, slegs as die gemiddelde lengte van die seedier meer as 1 meter is.

Stoor die gemiddelde lengte van die seedier in 'n nuwe veld met die naam **AvgLength.**

Voorbeeld van afvoer:

HabitatName	AvgLength
Hydrothermal Vent	3.12
Mangrove Swamp	1.5
Open Ocean (Pelagic)	1.07
Seagrass Meadow	6.45

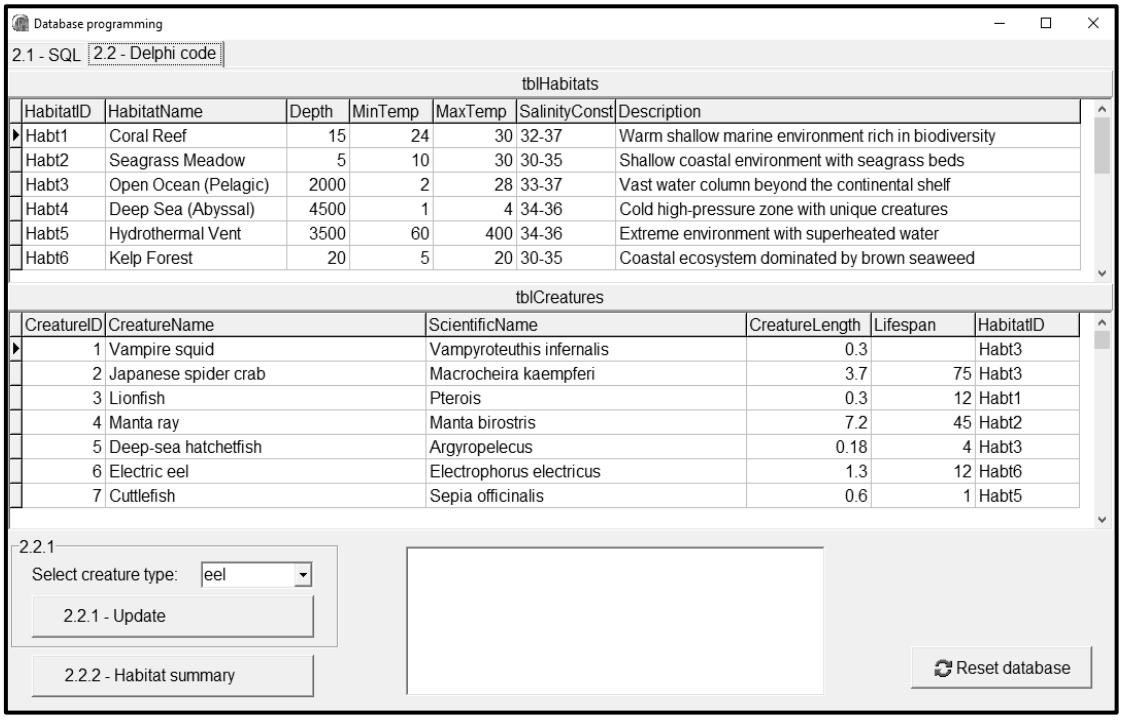
(7)

2.2 Oortjieblad ('Tab sheet') [2.2 - Delphi code]

LET WEL:

- Gebruik SLEGS Delphi-programmeringskode om VRAAG 2.2.1 en VRAAG 2.2.2 te beantwoord.
- GEEN punte sal vir SQL-stellings in VRAAG 2.2 toegeken word nie.

Voorbeeld van grafiese gebruikerskoppelvlak (GGK ('GUI')) vir VRAAG 2.2:



2.2.1 Knoppie [2.2.1 - Update]

Seediere van 'n spesifieke tipe sal soms na ander habitatte verskuif moet word, afhangende van verskeie faktore. Die tipe seedier word altyd as deel van die **CreatureName**-veld in die **tblCreatures**-tabel ingesluit.

Die gebruiker moet 'n tipe seedier uit kombinasieblokkie ('combo box') **cmbQ2_2_1** selekteer.

Kode is voorsien om:

- Die tipe seedier te onttrek en in veranderlike **sCreature** te stoor
- Die 'rich edit'-komponent **redQ2_2** skoon te maak en 'n opskrif daarin te vertoon

Skryf kode om te toets of die seedier se naam die tipe seedier insluit wat in die kombinasieblokkie geselekteer is, en dateer die seedier se naam op deur 'n asterisk-karakter (**'*'**) aan die einde in te sluit. Vertoon die **CreatureName**- en die **HabitatID**-velde in die 'rich edit'-komponent **redQ2_2**.

Voorbeeld van die eerste drie reëls met afvoer as 'eel' geselekteer is:

Creature name	Habitat ID
Electric eel*	Habt6
Gulper eel*	Habt3
Sphinxeel*	Habt3

(9)

2.2.2 **Knoppie [2.2.2 - Habitat summary]**

'n Opsomming van die habitatte en die getal seediere in elke habitat moet saamgestel word.

Kode is voorsien om die 'rich edit'-komponent **redQ2_2** skoon te maak en 'n opskrif daarin te vertoon.

Skryf kode om die getal seediere in elke habitat te tel. Gebruik die 'rich edit'-komponent **redQ2_2** om elke habitat se naam en die getal seediere in elke habitat te vertoon, slegs as daar ten minste een seedier in die habitat is.

Voorbeeld van afvoer van die eerste vyf name van habitatte en die getal seediere:

Habitat name	Number of creatures
Coral Reef	7
Seagrass Meadow	36
Open Ocean (Pelagic)	37
Hydrothermal Vent	8
Kelp Forest	15

(10)

- Sleutel jou eksamennommer as kommentaar in die eerste reël van die programlêer in.
- Stoor jou program.
- Druk die kode indien dit vereis word.

TOTAAL AFDELING B: 40

AFDELING C

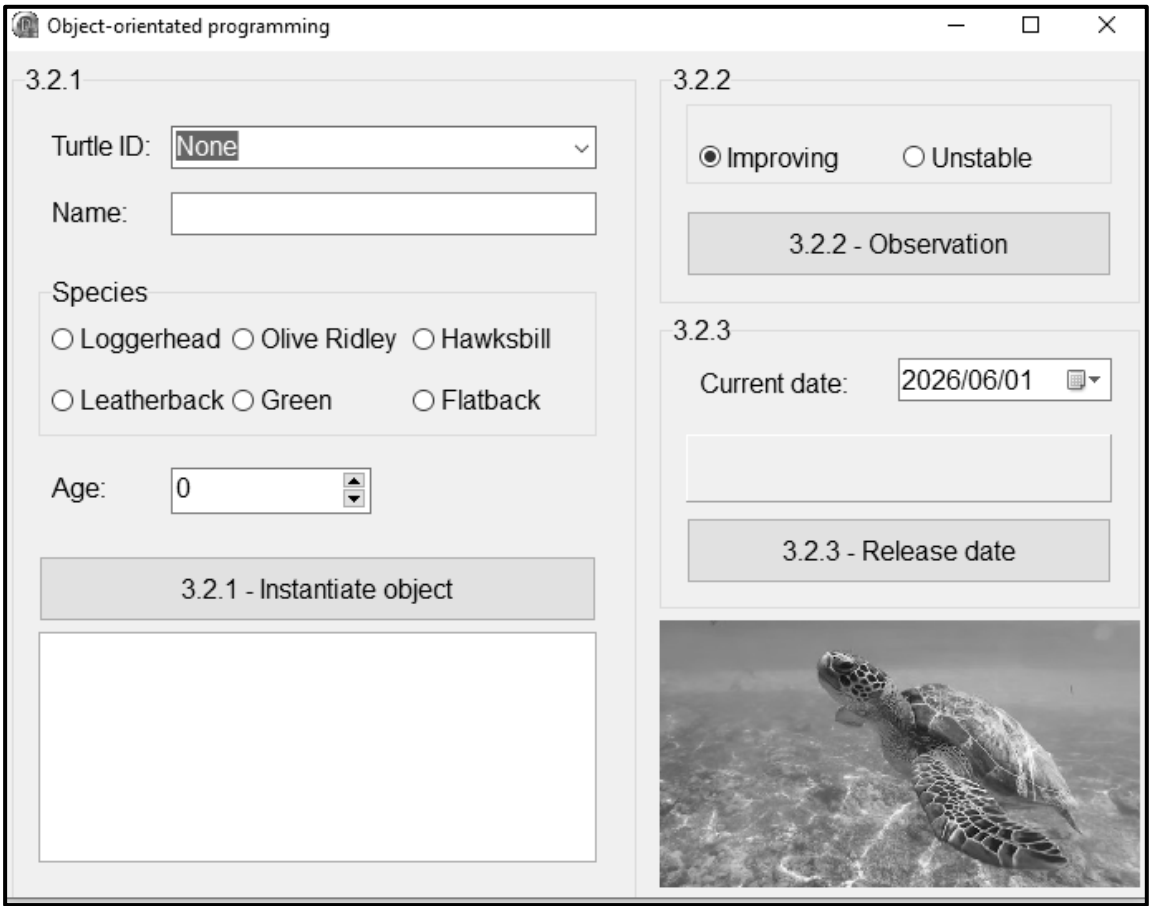
VRAAG 3: OBJEK-GEORIËNTEERDE PROGRAMMERING

'n Mariene reddingsentrum het hulp nodig om inligting oor beseerde seeskilpaaie tydens die skilpaaie se rehabilitasieproses aan te teken en op te dateer.

Doen die volgende:

- Maak die onvolledige program in die **Question 3**-lêergids oop.
- Maak die onvolledige objekklas **SeaTurtle_U.pas** oop.
- Sleutel jou eksamennommer as kommentaar in die eerste reël van beide die **Question3_U.pas**-lêer en die **SeaTurtle_U.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans beperkte funksionaliteit.

Voorbeeld van grafiese gebruikerskoppelvlak (GGK ('GUI')):



LET WEL:

Die datum wat op jou koppelvlak vertoon word, kan verskil van die datum wat hierbo vertoon word omdat die datum die huidige stelseldatum sal weerspieël.

- Voltooi die kode soos gespesifiseer in VRAAG 3.1 en VRAAG 3.2 wat volg.

LET WEL:

- Jy mag NIE enige addisionele attribute of gebruikersgedefinieerde metodes byvoeg NIE, tensy dit uitdruklik in die vraag genoem word.
- Die eenheidslêer **Populate_U** is voorsien om die program met data te vul. MOENIE die inhoud van die **Populate_U**-eenheid verander NIE.

3.1 Die onvolledige objekklas (**TSeaTurtle**) wat voorsien is, bevat die verklaring van ses attribute wat 'n **SeaTurtle**-objek beskryf.

Attribuut	Tipe	Beskrywing
fTurtleID	String	Unieke identifiseerder van die skilpad
fName	String	'n Naam wat vir die skilpad gegee is
fSpecies	String	Skilpadspesie
fAge	Integer	Geskatte ouderdom in jaar
fInjurySeverity	Integer	Beseringsvlak van 1 (matig) tot 10 (kritiek)
fRehabDays	Integer	Dae in rehabilitasie bestee

Die volgende metodes is voorsien:

- 'n Gedeeltelik voltooide **Constructor**-metode
- 'n Voltooide **toString**-metode

Voltooi die kode in die objekklas soos wat in VRAAG 3.1.1 tot VRAAG 3.1.6 wat volg, beskryf word.

3.1.1 Skryf kode vir 'n hulp('auxiliary')-metode met die naam **generateTurtleID** om 'n unieke **TurtleID** vir 'n seeskilpad te genereer en terug te stuur.

Genereer 'n unieke **TurtleID** in hoofletters deur die formaat wat hieronder voorsien is, te gebruik:

```
<eerste drie letters van die seeskilpadspesie><laaste twee letters van die seeskilpad se naam><fAge>
```

Voorbeeld van 'n **TurtleID** as die naam van die skilpad Sandy is, die spesie Loggerhead is en die ouderdom 48 is:

```
LOGDY48
```

(5)

3.1.2 Die onvolledige **Constructor**-metode ontvang vier parameters en ken die parameterwaardes aan die onderskeie attribute toe. Die waarde 5 is aan die **fInjurySeverity**-attribuut toegeken en die waarde 60 is aan **fRehabDays** toegeken.

Skryf kode om die konstruktor-metode soos volg te voltooi:

As die **TurtleID**-parameter 'None' is, roep die hulpmetode **generateTurtleID** om 'n unieke TurtleID aan die **fTurtleID**-attribuut toe te ken. (4)

3.1.3 Skryf kode vir 'n metode met die naam **injuryObservation** wat die waarneming ('observation') as 'n stringparameter ontvang.

Waarneming ('Observation')-parameter	**Doen die volgende:**
'Improving'	Verminder die huidige erns van die besering van die objek met die waarde van twee. As die erns van die besering 'n negatiewe waarde is, stel die waarde van die erns van die besering op een.
'Unstable'	Verander die erns van die besering na tien.

(6)

3.1.4 Skryf kode vir 'n toegang('accessor')-metode met die naam **getInjurySeverity** om die **fInjurySeverity**-attribuut terug te stuur. (2)

3.1.5 Skryf kode vir 'n wysiging('mutator')-metode met die naam **setRehabDays** wat die aantal dae as 'n heelgetalparameter ontvang en die **fRehabDays**-attribuut verander na die waarde wat ontvang is. (3)

3.1.6 Skryf kode vir 'n metode met die naam **estimateReleaseDate** wat die huidige datum as 'n parameter ontvang.

Bereken die geskatte vrylatingsdatum deur die **fRehabDays** by die huidige datum te tel en stuur die resultaat terug. (3)

3.2 'n Onvolledige program is in die **Question 3**-lêergids voorsien. Die program bevat kode vir die objekklas om toeganklik te wees en verklaar 'n globale objek-veranderlike met die naam **objSeaTurtle**.

Skryf kode om die take wat in VRAAG 3.2.1 tot VRAAG 3.2.3 beskryf word, uit te voer.

3.2.1 **Knoppie [3.2.1 - Instantiate object]**

Die kombinasieblokkie ('combo box') **cmbQ3_2_1** bevat die **TurtleID**'s van al die seeskilpaaie wat voorheen gemerk is. As 'n skilpad voorheen gemerk is, sal die toevoerkomponente met die inligting van die seeskilpad met die **TurtleID** wat in die kombinasieblokkie geselekteer is, gevul word.

As die geselekteerde **TurtleID** 'None' is, moet die gebruiker inligting in die toevoerkomponente **edtQ3_2_1**, **rgpQ3_2_1** en **spnQ3_2_1** insleutel.

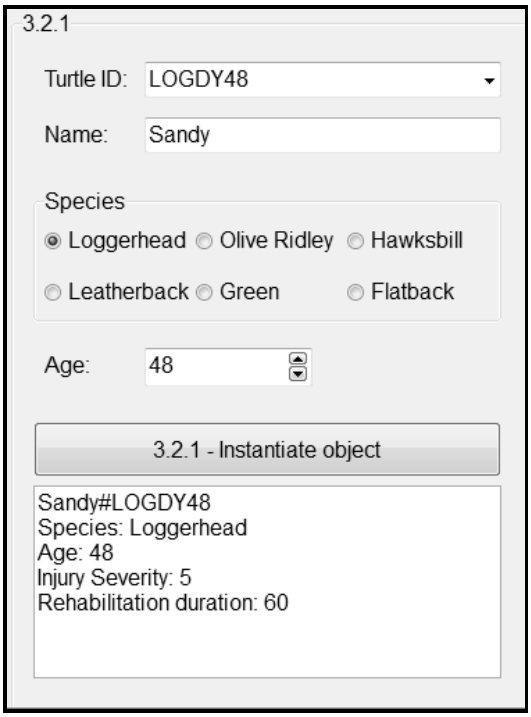
Kode is voorsien om:

- Die toevoerkomponente te vul wanneer 'n **TurtleID** in die kombinasieblokkie **cmbQ3_2_1** geselekteer is
- Die **TurtleID**, naam, spesie en ouderdom uit die relevante komponente te onttrek, en aan die onderskeie veranderlikes toe te ken
- Die 'rich edit'-komponent **redQ3_2** skoon te maak

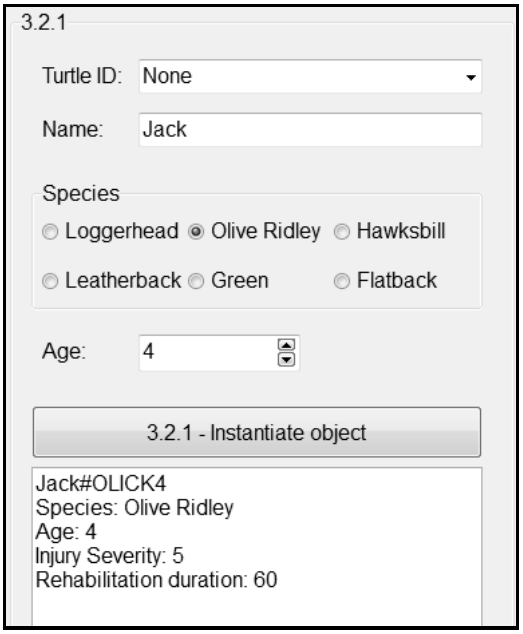
Skryf kode om die volgende te doen:

- Gebruik die inligting wat onttrek is om die **objSeaTurtle**-objek te instansieer.
- Roep die **toString**-metode van die objek om die seeskilpad se inligting in die 'rich edit'-komponent **redQ3_2** te vertoon.

Voorbeeld van toevoer en afvoer as 'n bestaande **TurtleID** (LOGDY48) geselekteer is:



Voorbeeld van toevoer en afvoer as die geselekteerde **TurtleID** 'NONE' was, die naam van die skilpad wat ingesleutel is, Jack was, Olive Ridley as spesie geselekteer is en 4 as ouderdom geselekteer is:



(5)

3.2.2 Knoppie [3.2.2 - Observation]

Daaglikse waarnemings sal gedoen word om die herstel van elke seeskilpad te monitor. Die erns van die besering word op grond van die waarnemings aangepas om die vordering van die herstel of terugslae van die skilpad weer te gee.

Die gebruiker moet 'n waarnemingstatus uit die radiogroep ('radio group') **rgpQ3_2_2** selekteer.

Kode is voorsien om:

- Die waarneming ('observation'), óf 'Improving' óf 'Unstable', uit die radiogroep **rgpQ3_2_2** te onttrek en in die veranderlike met die naam **sObservation** stoor
- Die 'rich edit'-komponent **redQ3_2** skoon te maak

Skryf kode om die volgende te doen:

- Roep die **injuryObservation**-metode van die objek deur die onttrekte waarneming as 'n argument te gebruik om die erns van die besering aan te pas.
- Op grond van die aangepaste erns van die besering, dateer die rehabilitasiedae-attribuut (**fRehabDays**) op grond van die volgende op:

Erns van besering	Rehabilitasiedae
1–4	30
5–7	60
8–10	120

- Roep die **toString**-metode van die objek om die seeskilpad se opgedateerde inligting in die 'rich edit'-komponent **redQ3_2** te vertoon.

Voorbeeld van afvoer as 'Improving' as waarneming geselekteer is:

```
Sandy#LOGDY48
Species: Loggerhead
Age: 48
Injury Severity: 3
Rehabilitation duration: 30
```

Voorbeeld van afvoer as 'Unstable' as waarneming geselekteer is:

```
Sandy#LOGDY48
Species: Loggerhead
Age: 48
Injury Severity: 10
Rehabilitation duration: 120
```

(9)

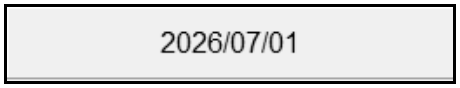
3.2.3 **Knoppie [3.2.3 - Release date]**

Die personeel by die mariene sentrum moet 'n geskatte vrylatingsdatum vir die seeskilpad, om veilig na sy natuurlike oseaanhabitat terug te keer, bereken.

Kode is voorsien om die datum uit die datum-tyd-uitsoeker('date time picker')-komponent **dtpQ3_2_3Current** te onttrek en in die veranderlike **dCurrentDate** te stoor.

Skryf kode om die relevante metode te roep om die vrylatingsdatum te bepaal en die resultaat in paneel ('panel') **pnlQ3_2_3** te vertoon.

Voorbeeld van toevoer en afvoer as die **TurtleID** LOGDY48 was, die waarneming 'Improving' was, die huidige datum 2026/06/01 was en die rehabilitasiedae 30 was:



(3)

- Sleutel jou eksamennommer as kommentaar in die eerste reël van die objekklas en die vormklas in.
- Stoor jou program.
- Druk die kode in die objekklas en die vormklas indien dit vereis word.

TOTAAL AFDELING C: 40

AFDELING D

VRAAG 4: PROBLEEMOPLOSSINGPROGRAMMERING

VRAAG 4 bestaan uit TWEE dele:

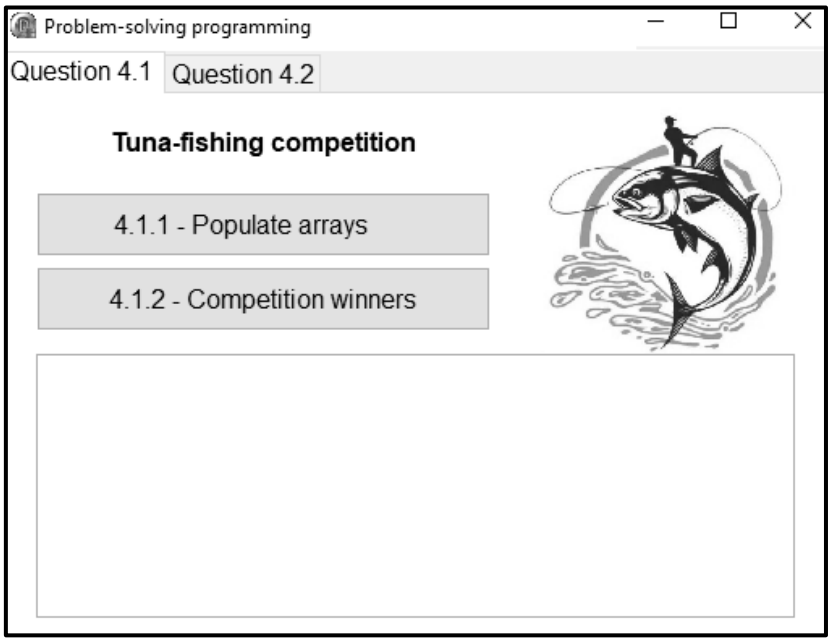
- VRAAG 4.1: Tuna-hengelkompetisie
- VRAAG 4.2: Pascal se driehoek

Doen die volgende:

- Maak die onvolledige program in die **Question 4**-lêergids oop.
- Sleutel jou eksamennommer as kommentaar in die eerste reël van die **Question4_U.pas**-lêer in.
- Kompileer en voer die program uit. Die program het tans beperkte funksionaliteit.

4.1 'n Aantal deelnemers het onlangs aan 'n tuna-hengelkompetisie deelgeneem.

Voorbeeld van grafiese gebruikerskoppelvlak (GGK ('GUI')):



Die volgende is voorsien:

- Drie globale parallelle skikkings ('arrays') is verklaar, elk met 'n grootte van 30:

```
arrSurnameName: array [1 .. 30] of String;
arrNumFish: array [1 .. 30] of Integer;
arrTotalWeight: array [1 .. 30] of Real;
```

- Die verklaring van 'n globale heelgetalveranderlike, **iNumEntries**, om op hoogte van die werklike getal inskrywings in die skikkings te bly:

```
iNumEntries: Integer = 0;
```

- 'n Tekslêer **Fishing.txt**, wat die uitslae van al die deelnemers wat aan die hengelkompetisie deelgeneem het, in die volgende formaat bevat:

```
Inskrywingsnommer van die deelnemer
Van, naam van die deelnemer
Gewig van elke vangs (in kg), elk op 'n aparte reël
Oop reël
```

Voorbeeld van die uitslae van die eerste vier deelnemers wat in die tekslêer gestoor is:

```
#001
Vermaak, John
288.90
98.32
188.21

#002
Petersen, Lindiwe

#003
Davids, Peter
381.21
90.22

#004
Johnson, Gaby
510.12
```

Verduideliking van die inligting van die eerste twee deelnemers wat in die tekslêer gestoor is:

- Vermaak John, met inskrywingsnommer #001, het drie visse gevang. Die gewig van elke vis (in kg) is op aparte reëls gelys.

- Petersen Lindiwe, met inskrywingsnommer #002, het geen visse gevang nie.

4.1.1 Knoppie [4.1.1 - Populate arrays]

Die inhoud van die tekslêer **Fishing.txt** moet gebruik word om die drie parallelle skikkings **arrSurnameName**, **arrNumFish** en **arrTotalWeight** te vul.

Skryf kode om die volgende te doen:

- Maak die tekslêer, **Fishing.txt**, wat voorsien is, oop om uit die lêer te lees. Jy mag aanneem dat die lêer bestaan.

- Lees deur die tekslêer en vul die drie parallelle skikkings met die relevante data uit die tekslêer soos hieronder verduidelik word:
    - Die skikking **arrSurnameName** moet die van en naam van die deelnemer, geskei deur 'n komma, bevat, bv. Vermaak, John.
    - Die skikking **arrNumFish** moet die getal visse wat deur die deelnemer gevang is, bevat, bv. John Vermaak het drie visse gevang.
    - Die skikking **arrTotalWeight** moet die totale gewig van die visse wat deur die deelnemer gevang is, bevat, bv. die totale gewig van die visse wat deur John Vermaak gevang is, is 575,43 kg.
    - As 'n deelnemer nie enige visse gevang het nie, moet 'n waarde van nul (0) in die **arrNumFish**- en **arrTotalWeight**-skikkings vir daardie deelnemer gestoor word.

LET WEL: 'n Prosedure met die naam **displayInfo** is voorsien en word aan die einde geroep om die inhoud van die skikkings te vertoon.

Voorbeeld van afvoer van die uitslae van die eerste sewe deelnemers:

Surname and Name	Number of fish	Total weight
Vermaak, John	3	575.43
Petersen, Lindiwe	0	0
Davids, Peter	2	471.43
Johnson, Gaby	1	510.12
Smith, John	4	1596.23
Jones, Robert	3	1502.74
Williams, Richard	4	1546.09

(14)

4.1.2 Knoppie [4.1.2 - Competition winners]

Die wenner is die deelnemer wat visse met die hoogste gemiddelde gewig gevang het.

Kode is voorsien om:

- Die 'rich edit'-komponent **redQ4_1** skoon te maak
- Veranderlike **rHighestAvg** te verklaar en te inisialiseer

Skryf kode om die naam van die deelnemer wat visse met die hoogste gemiddelde gewig gevang het, te bepaal en te vertoon. As meer as een deelnemer kwalifiseer om 'n wenner te wees, vertoon die name van al die wenners op aparte reëls in die 'rich edit'-komponent **redQ4_1**.

LET WEL: Kode is voorsien om die hoogste gemiddelde gewig te vertoon.

Voorbeeld van afvoer:

```
Taylor, Roger
Lewis, Ellen

Highest average: 591.87
```

(10)

4.2 Knoppie [4.2 - Pascal's triangle]

Pascal se driehoek is 'n driehoekige rangskikking van getalle wat verskillende wiskundige patrone en verwantskappe illustreer. Elke getal in die driehoek is die som van die twee getalle diagonaal bokant dit, wat begin met 1 aan die bopunt en 1'e al langs die sye af.

As Pascal se driehoek in 'n twee-dimensionele (2D-) skikking voorgestel word, sal die getalle, soos hieronder getoon word, gerangskik wees.

						1						
					1		1					
				1		2		1				
			1		3		3		1			
		1		4		6		4		1		
	1		5		10		10		5		1	
1		6		15		20		15		6		1

Die volgende is voorsien:

- 'n 2D-skikking, **arrPattern[1..7,1..15]**, wat verklaar en geïnisialiseer is:

```
arrPattern: array [1 .. 7, 1 .. 15] of Integer =
  ((0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0),
   (0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0),
   (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),
   (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),
   (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),
   (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),
   (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0));
```

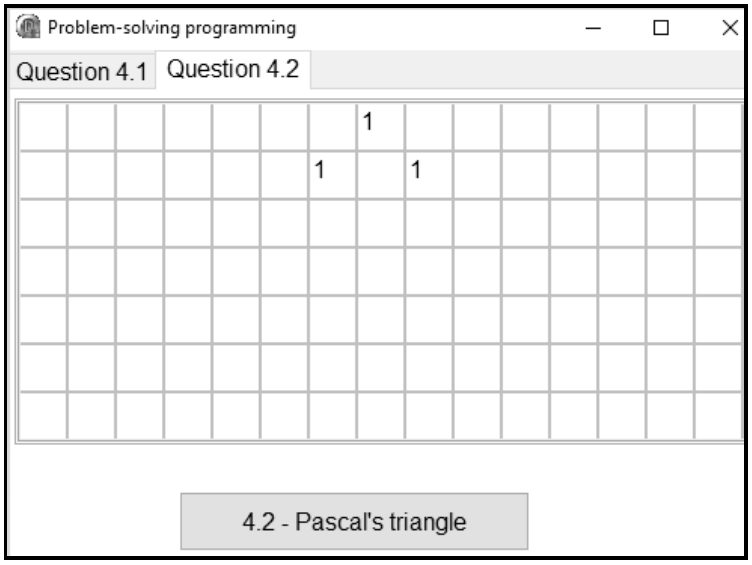
LET WEL:

Die enigste nie-nul-waardes wat tans in die 2D-skikking **arrPattern**, wat voorsien is, voorkom, weerspieël die waardes in die eerste twee rye van Pascal se driehoek. Hierdie waardes is in die skikking in die volgende posisies gesentreer:

```
arrPattern[1, 8] := 1;
arrPattern[2, 7] := 1;
arrPattern[2, 9] := 1;
```

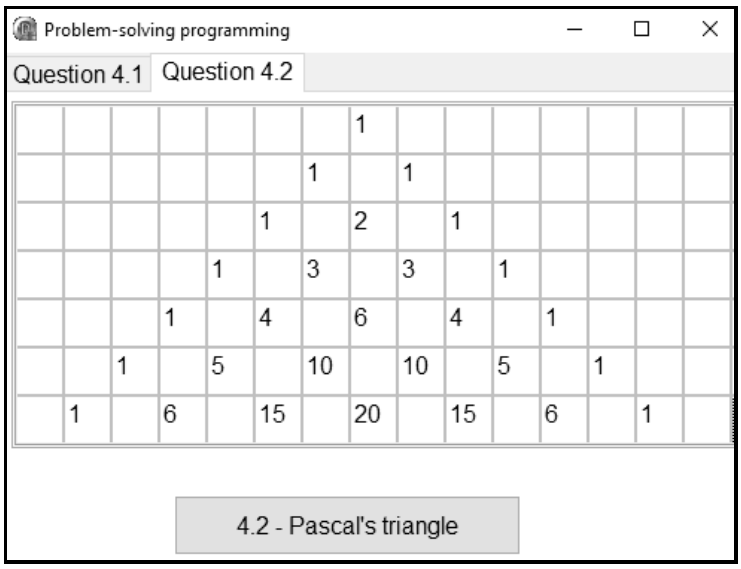
- 'n Prosedure met die naam **display2D** om die inhoud van die 2D-skikking in StringGrid **sgdQ4_2** te vertoon

Voorbeeld van grafiese gebruikerskoppelvlak (GGK('GUI')) met die eerste twee rye in StringGrid **sgdQ4_2** wat gevul is met die waardes wat aan die **arrPattern**-skikking toegeken is, soos op die vorige bladsy verduidelik is:



Skryf kode om Pascal se driehoek in die skikking **arrPattern** tot by ry 7 te voltooi deur herhaalstrukture te gebruik.

Voorbeeld van afvoer as die skikking met waardes tot by ry 7 van Pascal se driehoek gevul is:



LET WEL: Kode is voorsien om die **display2D**-prosedure te roep. (6)

- Sleutel jou eksamennommer as kommentaar in die eerste reël van die programlêer in.
- Stoor jou program.
- Druk die kode indien dit vereis word.

TOTAAL AFDELING D: 30
GROOTTOTAAL: 150

Kopiereg voorbehou

INLIGTINGSTEGNOLOGIE V1

DATABASIS-INLIGTING VIR VRAAG 2:

Die ontwerp van die **MarineDB**-databasistabelle is soos volg:

Tabel: **tblHabitats**

Hierdie tabel bevat die besonderhede van elke habitat.

Veldnaam	Datatipe	Beskrywing
HabitatID	Short Text	Unieke ID van die habitat, begin met 'Habt', gevolg deur 'n getal
HabitatName	Short Text	Naam van die habitat
Depth	Number	Diepte van die habitat in meter
MinTemp	Number	Algemene minimum temperatuur (°C) van die habitat
MaxTemp	Number	Algemene maksimum temperatuur (°C) van die habitat
SalinityConst	Short Text	Die soutkonsentrasie van die habitat in die formaat Minimum-Maksimum
Description	Short Text	Kort beskrywing van die habitat

Voorbeeld van die rekords van die **tblHabitats**-tabel:

HabitatID	HabitatName	Depth	MinTemp	MaxTemp	SalinityConst	Description
Habt1	Coral Reef	15	24	30	32-37	Warm shallow marine environment rich in biodiversity
Habt2	Seagrass Meadow	5	10	30	30-35	Shallow coastal environment with seagrass beds
Habt3	Open Ocean (Pelagic)	2000	2	28	33-37	Vast water column beyond the continental shelf
Habt4	Deep Sea (Abyssal)	4500	1	4	34-36	Cold high-pressure zone with unique creatures
Habt5	Hydrothermal Vent	3500	60	400	34-36	Extreme environment with superheated water
Habt6	Kelp Forest	20	5	20	30-35	Coastal ecosystem dominated by brown seaweed
Habt7	Mangrove Swamp	2.5	20	35	10-35	Coastal ecosystem with salt-tolerant trees
Habt8	Arctic Ocean	500	-2	5	32-35	Cold waters with seasonal ice cover
Habt9	Antarctic Ice Shelf	250	-2	2	33-35	Ice-covered marine ecosystem

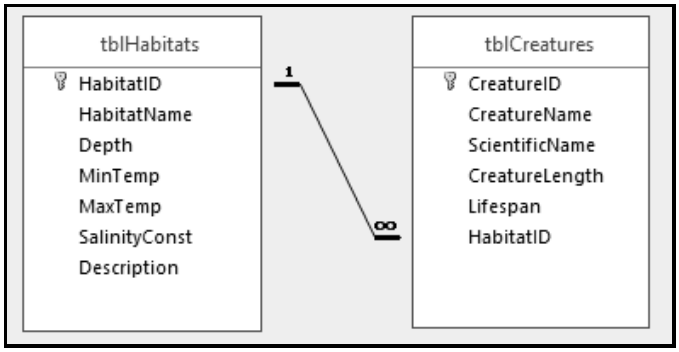
Tabel: **tblCreatures**

Hierdie tabel bevat die besonderhede van elke seedier.

Veldnaam	Datatipe	Beskrywing
CreatureID	Number	Unieke ID van die seedier
CreatureName	Short Text	Naam van die seedier
ScientificName	Short Text	Wetenskaplike naam van die seedier
CreatureLength	Number	Die lengte van die seedier in meter (m)
Lifespan	Number	Die geskatte leeftyd van die seedier in jaar. As die waarde onbekend is, word die veld leeg gelaat.
HabitatID	Short text	Die ID van die habitat, wat die habitat waartoe die seedier behoort, identifiseer

Voorbeeld van die eerste tien rekords van die **tblCreatures**-tabel:

CreatureID	CreatureName	ScientificName	CreatureLength	Lifespan	HabitatID
1	Vampire squid	Vampyroteuthis infernalis	0.3		Habt3
2	Japanese spider crab	Macrocheira kaempferi	3.7	75	Habt3
3	Lionfish	Pterois	0.3	12	Habt1
4	Manta ray	Manta birostris	7.2	45	Habt2
5	Deep-sea hatchetfish	Argyropelecus	0.18	4	Habt3
6	Electric eel	Electrophorus electricus	1.3	12	Habt6
7	Cuttlefish	Sepia officinalis	0.6	1	Habt5
8	Dumbo octopus	Grimpoteuthis	0.33	4	Habt3
9	Giant isopod	Bathynomus giganteus	0.51		Habt3
10	Flying fish	Exocoetidae	0.35	7	Habt2



BEPLANNINGSBLADSY 1

Kopiereg voorbehou

BEPLANNINGSBLADSY 2

Kopiereg voorbehou

Eksamenplakker	
	150

INLIGTINGSTEGNOLOGIE V1 – MEI/JUNIE 2026
INLIGTINGSBLAD *(moet NÁ AFLOOP VAN die 3 uur-sessie deur die kandidaat ingevul word)*

SENTRUMNOMMER: _____

EKSAMENNOMMER: _____

WERKSTASIENOMMER: _____

Weergawe van Delphi wat tydens die INLIGTINGSTEGNOLOGIE SS/NSS MEI/JUNIE 2026-eksamen gebruik is:

Maak 'n kruisie in die toepaslike blokkie (**X**)	Delphi 2010	Delphi XE	Delphi 10.3	Delphi Community	Delphi 11	Delphi 12	Ander: (*Spesifiseer*) _____

NAAM VAN LÊERGIDS ('FOLDER'): _____

Kandidaat moet afmerk as die lêernaam (lêername) wat vir elke antwoord gebruik is, gestoor is en/of 'n poging aangewend is.

Vraag-nommer	Lêernaam	Gestoor (✓)	Gepoog (✓)	Maksimum Punt	Punt Toegeken	Nasiener/ Kode
1	Question1_P.dproj			**40**		
2	Question2_P.dproj			**40**		
3	SeaTurtle_U.pas			**23**		
	Question3_P.dproj			**17**		
4	Question4_P.dproj			**30**		
TOTAAL				**150**		

Kommentaar: *(slegs vir kantoor/nasiener se gebruik)*

_____

_____

Kopiereg voorbehou