**Практична робота № 12.**

**Тема: Введення й виведення даних, робота зі змінними.**

**Мета роботи:**Навчитися розміщувати на формі елементів керування та настроювати їх властивості, надавати властивостям елементів керування значення змінних, інших властивостей та констант; розробляти на базі форм програми, в яких дані вводяться і виводяться за допомогою елементів керування; призначення таких елементів керування як прапорець та група перемикачів, використовувати вікно введення для надання змінним значень.

**Обладнання:**ПК, проектор, програма Delphi, інструкція з практичної роботи.

**Література:** Кащеєв Л.Б. Основи візуального програмування: Навч. посібник – Х.:Веста, 2011. – 192 с.

**Теоретичні відомості.**

Як і в інших мовах програмування, у мові Pascal і в Delphi, яка успадкувала від неї основні конструкції дані поділяються на константи і змінні. У програмі вони визначаються ідентифікаторами (іменами), за якими до них можна звертатися для одержання поточних значень і включати їх в арифметичні, логічні або рядкові вирази.

Константами називаються елементи даних, значення яких установлені в описовій частині програми й далі не змінюються. наприклад, якщо ви використовуєте в програмі значення кількості учнів у вашому класі, то найкраще задати його константою, бо в разі обчислення, наприклад, тих чи інших параметрів успішності ця кількість не змінює свого значення. Якщо ж до вашого класу прийде новий учень, досить збільшити константу на одиницю, і всі розрахунки будуть виконуватися з новим значенням чисельності. Це значно простіше, ніж переглядати всю програму, змінюючи кількість, задану числом, на нове

значення. Константи описуються в спеціальному розділі, що починається зарезервованим словом const (від англ. constant — константа) у форматі:

сonst

<ідентифікатор> = <значення константи>;

тип користувацьких констант розпізнається компілятором без попереднього опису.

У Pascal є ряд констант, до значень яких можна звертатися без попереднього визначення, наприклад Pi. Їх називають зарезервованими константами. Змінними називають величини, які можуть змінювати свої значення в процесі виконання програми. Кожна змінна, як і константа, належить до певного типу даних. тип змінних має бути відомий компіляторові до початку трансляції виразів і операторів із використанням цих змінних. Крім того, згідно з відомим типом даних компілятор виділяє під змінні в оперативній пам’яті необхідне місце (потрібна кількість байтів повністю визначається застосовуваним типом). Змінні описуються в спеціальному розділі, що починається зарезер-

вованим словом var (від англ. variable — змінна), у форматі:

var

<ідентифікатор> : <тип>

для позначення змінних і сталих величин, різних процедур та функцій використовують

ідентифікатори — імена, що встановлюють відповідність між об’єктом і певним набором символів (латинських літер, знаків підкреслення й цифр; цифра не може

бути першим символом). Можливості програміста у виборі імен для своїх змінних досить широкі. не можна тільки використовувати як ідентифікатори зарезервовані слова, які компілятор сприймає як базові директиви або оператори програми. наприклад, не можна назвати змінну begin або for. не рекомендовано брати як змінні й імена стандартних ідентифікаторів. Стандартні ідентифікатори введені творцями мови для позначення типів даних, констант, процедур і функцій, наприклад: integer, sin, read, readln, write, writeln. стандартний ідентифікатор sin, поданий у цьому далеко не повному списку, викликає функцію, що обчислює синус заданого кута; ідентифікатори read і readln викликають процедуру, яка організовує введення даних, а writeіwriteln— процедуру, яка організовує виведення даних. на відміну від зарезервованих слів, кожен зі стандартних ідентифікаторів можна перевизначити, але після цього програму важко читати й змінювати, так що краще цією можливістю не користуватися. Користувацькі ідентифікатори програміст вводить для позначення констант, змінних, процедур і функцій у програмі. При цьому ідентифікатори мають бути унікальними: у кожному програмному блоці один і той самий ідентифікатор не може використовуватися для позначення більш ніж однієї змінної, функції або константи. Дублювання ідентифікаторів спричиняє помилку на етапі компіляції. В ідентифікатор мовою Delphi не можуть входити пробіли, спеціальні символи й літери будь-яких алфавітів, окрім англійського (латинського). Правильно обрані ідентифікатори значно полегшують читання й розуміння програми, а також зменшують імовірність появи помилок, якщо в неї вноситимуть зміни.У ході візуального проектування форми програміст виносить на поле форми необхідні йому відеокомпоненти. робиться це простим перетяганням мишею необхідного відеокомпонента з інструментальної лінійки компонентів на форму.Властивості форм і відеокомпонентів (або, як ще кажуть, їхні атрибути), по суті, є змінними і як такі можуть використовуватись у програмі. Їхня основна відмінність від користувацьких змінних — нерозривний зв’язок з об’єктом, якому вони належать. розроблений творцями Delphi метод визначає, що зміниться в поведінці об’єкта в разі зміни атрибута.

Опис типів даних указує компіляторові, як зберігати та обробляти інформацію в програмі, які процедури й функції можна застосувати до даного типу.До створення мови Pascal, попередниці Delphi, у мовах програмування була повна плутанина. В одних мовах (Fortran) тип даних визначався першою літерою імені, в інших (Basic) — останньою. Існувало і явне задавання типів.Однією з причин поширення Pascal (а за ним і Delphi) стала передова на той момент система типів даних, що була найповнішою і найбільш несуперечливою.типи даних поділяються на стандартні і створювані. стандартні типи вже визначені в мові, а змінну такого типу можна оголосити, про-сто пославшись на ім’я цього типу. створювані типи програміст мусить конструювати сам відповідно до призначення програми.типи даних, обумовлені користувачем, задаються:

* у розділі визначення типів програми;
* у модулі (y unit),
* у розділі interface або implementation;
* у підключеному зовнішньому модулі (y unit);
* усередині процедур;
* усередині функцій.

Оголошення типів діють у межах того блоку, у якому вони розміщені. Поза цим блоком посилатися на такі типи не можна. Усередині ж вони замінюють (перекривають) зовнішні типи з тим самим ім’ям. Оголошені типи даних можна застосовувати в межах ділянки їх

видимості.Оголошення типів у Pascal є для компілятора правилами, які він повинен «запам’ятати» на випадок, якщо раптом зустріне в програмі посилання на той чи інший тип.

саме по собі оголошення типу не вносить у програму ніяких змін. тип змінної обмежує її значення й ті операції, які можна виконувати з цими значеннями.Оголошення типів і змінних схематично можна подати так:type

typel = type\_definition1; //Кожному новому типу//присвоюється ім’я, потім він визначається через//уже наявні типи.

type2 = type\_definition2; //В одному розділі «type» можна//оголосити кілька типів. Найпростіше визначення//типу складається з імені типу, визначеного раніше.

type3 = typel;нові змінні оголошуються у var: var var1: type\_definitions;

Кожній новій змінній присвоюється спочатку ім’я, а потім — тип.До основних типів даних ObjectPascal належать:

* прості;
* рядкові — для зберігання послідовностей символів;
* структуровані;
* вказівні — для посилань на змінні заданих типів;
* процедурні — для звертання до процедур і функцій як до змінних;
* варіантні — для зберігання в одній змінній даних різних типів.

Зазвичай ідентифікатори типів (виділені для назви змінні) використовуються тільки у визначенні нових типів або оголошенні змінних.

У Delphi визначено й доповнено структуру типів ObjectPascal:

Simple(прості):ordinal(порядкові типи); integer(цілий);character (літерні типи); boolean(логічний);enumerated(перелічений); subrange(діапазон);real(дійсний); tdatetime(дата-час).Structured(структуровані, складені):array(масив); string(рядкові типи);listи stringlist(списки); set(множина);record(запис); file(файлові типи);textfile(текстові файли);

Class(класи);classreference(посилання на класи); interface(інтерфейси);pointer(покажчик);variant(варіант);procedural(процедурний);typeidentifier(ідентифікатор типу).

Оператор присвоювання — механізм у програмуванні, що дозволяє змінювати значення змінних або властивостей об’єктів шляхом заміни старих значень новими. З фізичної точки зору, операція присвоювання полягає в записуванні значень у комірку пам’яті, яка під час компіляції була виділена для зберігання цієї змінної. Загальний синтаксис простого присвоювання має такий вигляд: <вираз ліворуч>:=<вираз праворуч>. Компіляція з дальшим виконанням додатка здійснюється за допомогою команди Run(в основному меню) →Runабо «гарячої» клавіші F9(або вибір ). Виконання буде здійснюватися тільки у випадку, якщо в ході компіляції не виявлено помилок і завантажувальний модуль створено. Компіляція без дальшого виконання додатка здійснюється за допомогою команди Project

→CompileProject або «гарячих» клавіш Ctrl+F9. У результаті компіляції можуть бути зроблені зауваження — Hint, попередження — Warningі повідомлення про помилки — Error.

**Хід роботи**

1. Виконати завдання №1.Створити на диску D: папку MyProgram.

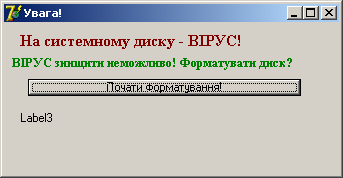
Завантажити середовище Delphi. Перегляньте вікна, що утворюють середовище Delphi. Спробуйте переміщати вікна, згортати, відновлювати.

1. Завдання №2. Базовий рівень – задовільно, добре (до 7 балів).

Створити проект, що реалізує такі дії:

a) Появу вікна за приведеним зразком;

b) Після натискання кнопки «Почати форматування!» забезпечити появу повідомлення «Це був жарт!».

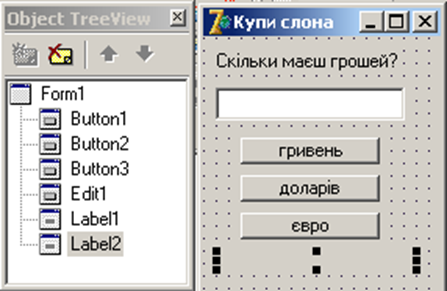


Зберегти проект в папці D:\MyProgram з іменем Program1 (Меню File - Save Project As).

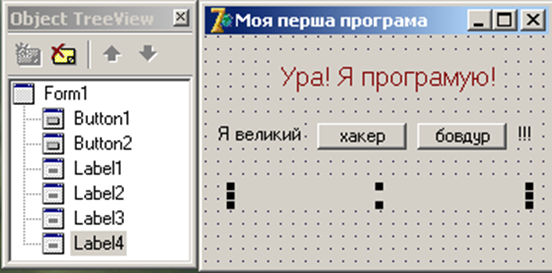
1. Завдання №3. Завдання, яке оцінюється добре, відмінно по варіантам (до 11 балів).

Створити відповідну форму за зразком, але надати шрифтам у формі властивостіза вашим розсудом та збережіть у папці D:\MyProgram з іменем Program2

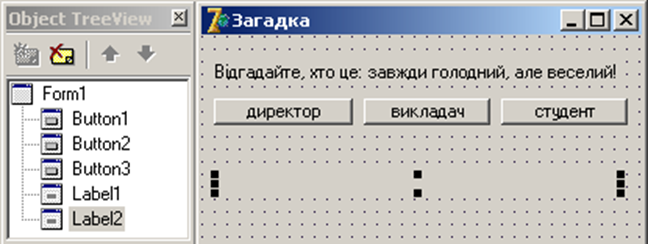
І варіант.Після натискання будь-якої клавіші повинно з’явитись повідомлення «Маю зайвого слона. Купи!».



ІІ варіант. Після натискання будь-якої клавіші повинно з’явитись повідомлення «Та це вже занадто!».



ІІІ варіант. Після натискання клавіші «директор», повинно з’явитись повідомлення «він не голодний!», після натискання клавіші «викладач», повинно з’явитись повідомлення «він не веселий!», після натискання клавіші «студент», повинно з’явитись повідомлення «Так, він завжди голодний, але веселий!».



1. Оформити звіт скринами відповідних завдань. Зробити висновки.
2. Контрольні питання:

* Дайте визначення поняття «константа». У чому полягає особливість її використання?
* Що собою являють змінні? наведіть приклади їх описів.
* Дайте визначення поняття «ідентифікатор».
* Чим відрізняються стандартні і користувацькі ідентифікатори?
* У чому полягає різниця між змінними і властивостями об’єктів?
* Яке призначення мають типи даних? Де вони описуються?
* Перелічіть типи даних, що належать до простих типів.
* Перелічіть типи даних, що належать до структурованих (складених) типів.
* Для чого призначена операція присвоювання?
* Як можна відкомпілювати додаток у Delphi?
* Наведіть приклади помилок, які можуть виникнути в програмі.