

Уроки 35 і 36. Складні логічні вирази.

Вивчення нового матеріалу.

Слайд №1

Сьогодні ми створимо проект із бланком замовлення для піцерії.

Клієнт має вказати кількість кожної страви.
Програма має автоматично визначати вартість кожного найменування в замовленні, а після натискання кнопки **Розрахувати** – вивести сумарну вартість замовлення.

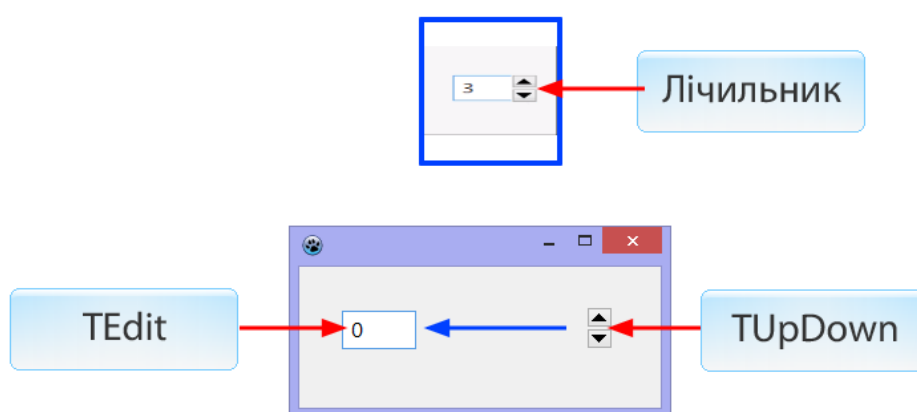
Найменування	Ціна, грн.	Кількість	Вартість, грн.
Піца	75	0	
Пиріжок	12	0	
Морозиво	17	0	
Тістечко	15	0	
Десерт	25	0	
Сік	8	0	

Вартість замовлення: грн.





Слайд №2

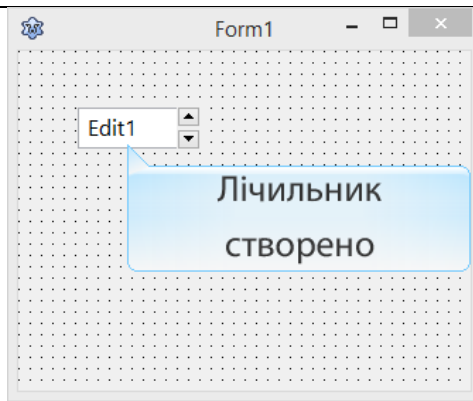
Як бачимо, для введення кількості страв певного найменування необхідно застосувати відразу два елементи керування: **Edit** та **UpDown**, які разом утворюють так званий **лічильник**.



Слайд №3

Створення на формі лічильника

- Оберіть інструмент  **TEdit**.
- Додайте елемент **Edit**, клацнувши на формі.
- Перейдіть на вкладку **Common Controls**.
- Оберіть інструмент  **TUpDown**.
- Розташуйте елемент **UpDown** на формі, клацнувши праворуч від текстового поля.
- Припасуйте елемент **UpDown** до елемента **Edit**.



Основними властивостями елементу **TUpDown** є:

Max – максимальне значення лічильника;

Min – мінімальне значення лічильника;

Position – поточне значення лічильника.

Новий матеріал до вправи 1.

Слайд №4

Введемо програмний код для пов'язування між собою лічильника **UpDown1** і текстового поля **Edit1**

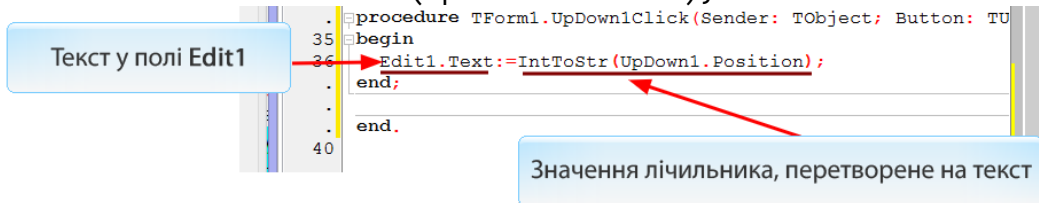
Слайд №5

Уведення коду

- Перейдіть до обробника події натискання стрілки лічильника **UpDown1**. Для цього двічі клацніть елемент **UpDown1** на формі.

- Уведіть код:

```
Edit1.Text:=IntToStr(UpDown1.Position);
```



Слайд №6

Перевірка роботи проекту

- Запустіть проект на виконання.
- Клацайте на лічильнику кнопку **вгору**.
- Потім клацайте кнопку **вниз**.

Вправа 1.

Вправа №1	Вправа 1 у Lazarus
	<ol style="list-style-type: none">Створіть форму згідно зразка.Надайте написам, що міститимуть значення ціни, імена P1, P2, ..., P6, а написам зі значенням вартості – C1, C2, ..., C6.Уведіть програмні коди для пов'язування між собою значень властивості Position лічильників TUpDown з полями TEdit.Щоб поля кількості можна було змінювати тільки через лічильник, установіть їхні властивості ReadOnly.Запустіть проект на виконання та перевірте роботу лічильників.

Це написи, а не текстові поля!

Підказка до вправи 1.

Підказка до вправи 1	Підказка до вправи 1
	<p>Для створення написів, що містять ціну та вартість, скористайтеся такими властивостями:</p> <ul style="list-style-type: none">• Autosize – для можливості змінення розміру;• Color – для заливки кольором;• Aligment і Layout – для центрування тексту в написі.

Вправа 2.

Вправа №2	Вправа 2 у Lazarus
	<p>Оголосіть змінні x1, x2, ..., x6 типу real для зберігання ціни та y1, y2, ..., y6 також типу real для вартості страв.</p>

Слайд №7

Уведемо програмний код для знаходження вартості піци в разі клацання лічильника



Слайд №8

Уведення коду

- Перейдіть до коду обробника події клацання лічильника **UpDown1**, клацнувши після першої команди.
- Уведіть код:

```
X1:=StrToFloat(P1.Caption);  
y1:=x1*UpDown1.Position;  
C1.Caption:=FloatToStr(y1);
```

```
procedure TForm1.UpDown1Click(Sender: TObject; Button: TUDBtnType;  
begin  
Edit1.Text:=IntToStr(UpDown1.Position);  
x1:=StrToFloat(P1.Caption);  
y1:=x1*UpDown1.Position;  
C1.Caption:=FloatToStr(y1);  
end;
```

Обчислення вартості

Присвоєння змінній x1 значення ціни

Присвоєння напису C1 значення вартості y1

Слайд №9

Перевірка роботи проекту

- Запустіть проект на виконання.
- Клацніть на першому лічильнику кнопку **вгору** три рази.

Найменування	Ціна, грн.	Кількість	Вартість, грн.
Піца	75	3	225
Пиріжок	12	0	
Морозиво	17	0	
Тістечко	15	0	
Десерт	25	0	
Сік	8	0	

Вартість замовлення: грн.

Вправа 3.

Вправа №3

Вправа 3 у Lazarus

Уведіть коди для обчислення вартості кожного найменування в разі натискання відповідного лічильника (об'єкта класу **TUpDown**)

Створіть код для одного лічильника, а потім скопіюйте його до подій, що пов'язані з іншими лічильниками, і змініть назви об'єктів та змінних

Вправа 4.

Вправа №4

Вправа 4 у Lazarus

Сформуйте програмний код для визначення загальної вартості замовлення в разі натискання кнопки **Розрахувати**



Слайд №10

У нашій піцерії для клієнтів пропонують різноманітні акції.

Для любителів соку при купівлі піци та морозива
сік можна купити на 50% дешевше.

Соковита насолода



Слайд №11

Якщо потрібно перевірити відразу декілька умов,
використовують операцію **And (i)**

Такий вираз називають **складеним** логічним виразом.
Він має такий загальний вигляд:

`(логічний вираз 1) And (логічний вираз 2)`

У цьому разі значення складеного логічного виразу
буде **істинним** тільки тоді, коли **обидва** вирази є істинними

Вправа 5 у Lazarus

Змініть програмний код для визначення вартості соку, якщо при купівлі піци та морозива сок буде коштувати на 50% менше.

Запустіть проект та перевірте його виконання



Для бажаючих смачно та ситно поїсти при купівлі тістечка або морозива піцу можна придбати на 10% дешевше.



Якщо має виконуватися одна із вказаних умов,
використовують операцію **Or (або)**

Такий вираз також називають **складеним**.

Він має такий загальний вигляд:

(логічний вираз 1) **or** (логічний вираз 2)

У цьому разі значення складеного логічного виразу
буде **істинним** тоді, коли **хоча б один** із виразів (1 чи 2) є істинним

Вправа 6.

Вправа 6

Вправа 6 у Lazarus

Сформуйте програмний код для визначення вартості
піци, якщо при купівлі морозива або тістечка вона буде
коштувати на 10% дешевше.

Перевірте, як діятиме програма при виборі різних
кількостей морозив і тістечок.

