

# ІНФОРМАТИКА



8



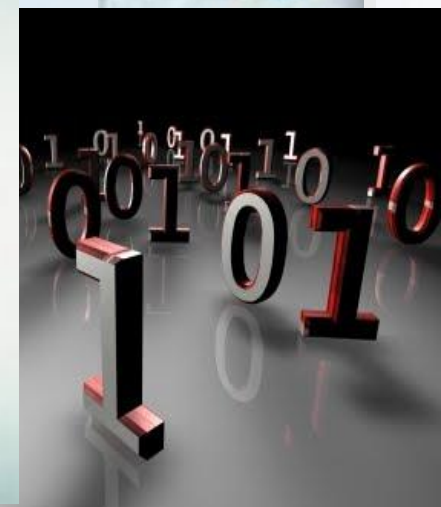
# Програмування





# ОСНОВИ ПОДІЙНО- ТА ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ

```
string n  
if(parameters.contains("name"))  
    hql += " and p.name = :name";  
}  
if(parameters.contains("age"))  
    hql += " and p.age = :age";  
}  
TypedQuery<Person> query = em.createQuery(hql);  
if(parameters.contains("name"))  
    query.setParameter("name", name);  
if(parameters.contains("age"))  
    query.setParameter("age", age);  
return query.getResultList();
```







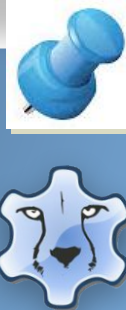
# ОСНОВИ ПОДІЙНО- ТА ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ



**Ти дізнаєшся:**



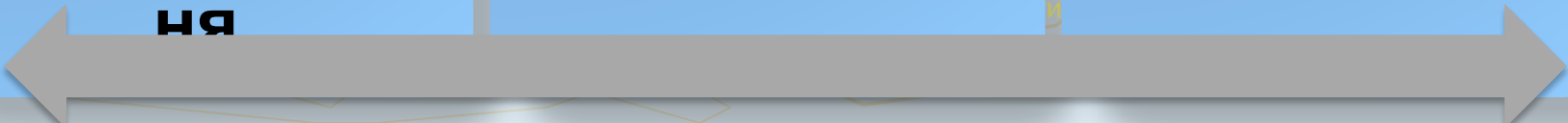
**Як змінити  
значення  
властивостей  
об'єктів у  
середовищі  
програмуван  
ня**



**Як об'єкти  
можуть  
«реагувати»  
на події**



**Як у проєкті  
використовую  
ть елемент  
управління  
кнопка**





ОСНОВИ ПОДІЙНО- ТА  
ОБ'ЄКТНО-ОРІЄНТОВАНОГО  
ПРОГРАМУВАННЯ



# Поняття об'єкту та його властивостей і





## ОПРАЦЮВАННЯ ОБ'ЄКТІВ МУЛЬТИМЕДІА



### *Пригадайте:*

- ☐ які дії можна виконати з елементом управління кнопка у програмах із графічним інтерфейсом;
- ☐ для чого використовуються кнопки управління вікном в ОС Windows;
- ☐ яку середовищі Скретч можна створити об'єкт кнопка;
- ☐ за допомогою яких команд у середовищі Скретч можна передати дію одного об'єкта іншому.

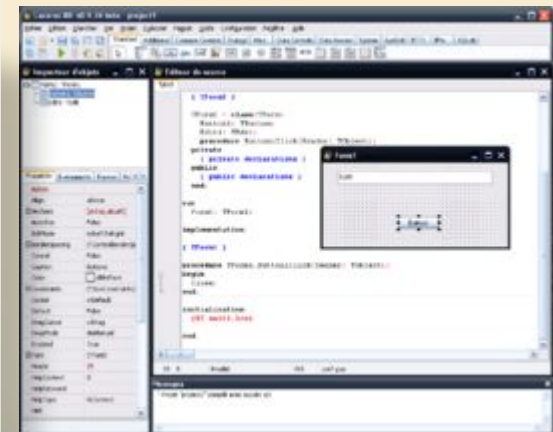
# Основи подійно- та об'єктно-орієнтованого програмування

## Значення властивостей об'єктів

Вивчаємо

Усі елементи управління, які можна розміщувати на формі проекту, як і сама форма, є об'єктами і мають властивості об'єкта, зокрема напису та форми, **статичним** способом.

Це означає, що значення властивостей об'єктів встановлюються до запуску програми на виконання



Чайчик О.Ф., вчитель інформатики

## Програмування

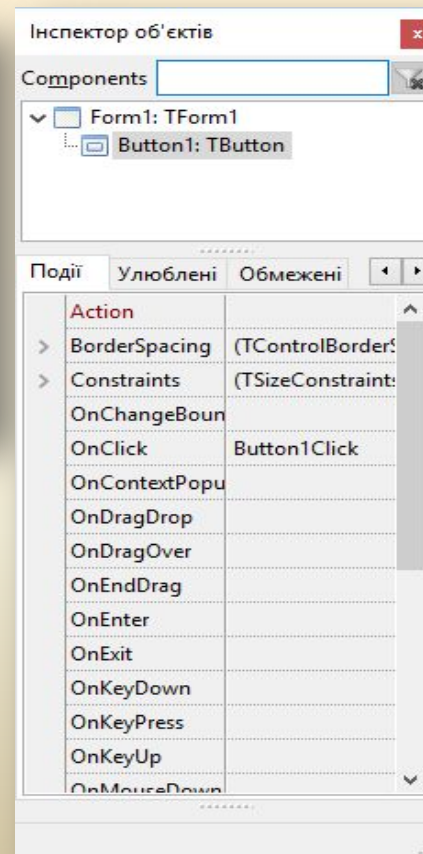
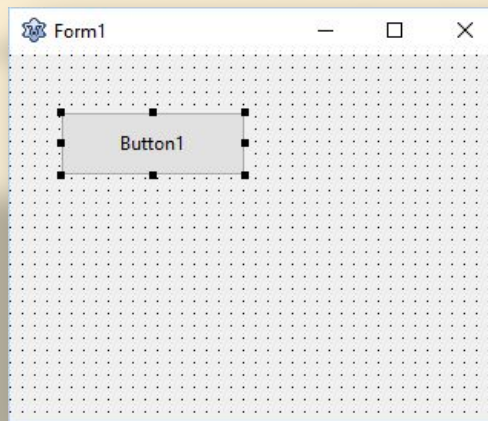


# Основи подійно- та об'єктно-орієнтованого програмування

## Значення властивостей об'єктів

Вивчаємо

Користувач спочатку виділяє у **вікні дизайнера** форми об'єкт, значення властивості якого необхідно змінити, а потім у вікні **Інспектор об'єктів** у таблиці властивостей знаходить назву відповідної властивості та праворуч від цієї назви вибирає або вводить із клавіатури потрібне значення



Чащук О.Ф., вчитель інформатики

Програмування

# Основи подійно- та об'єктно-орієнтованого програмування

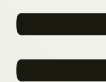
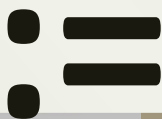
## Значення властивостей об'єктів

Вивчаємо

Для зміни значень властивостей можна застосувати спосіб — **динамічний**, коли значення властивостей можна змінити в процесі виконання програми за допомогою команди надання значень — **присвоювання**

*Lazarus*

*Python*



Чашиж О.Ф., вчитель інформатики

Програмування



# Основи подійно- та об'єктно-орієнтованого програмування

## Значення властивостей об'єктів

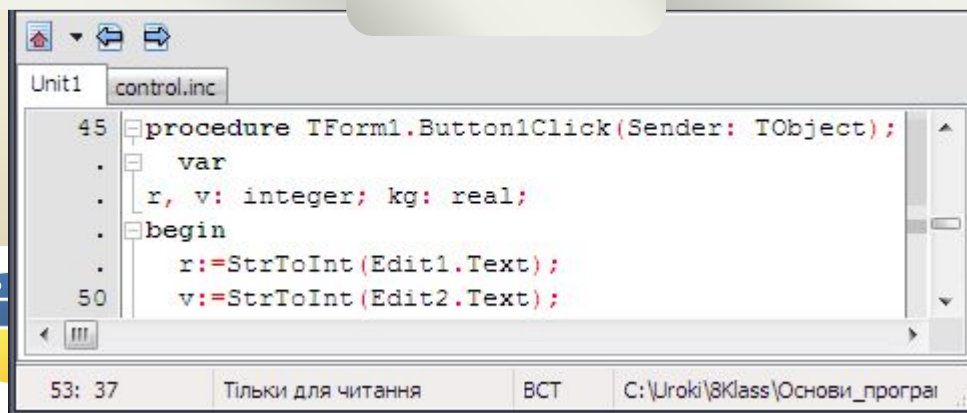
Вивчаємо

Команди записують у програмному коді, і значення властивості буде змінюватися тільки після запуску програми на виконання

ім'я властивості

:=

значення



```
Unit1 control.inc
45 procedure TForm1.Button1Click(Sender: TObject);
.   var
.     r, v: integer; kg: real;
.   begin
.     r:=StrToInt(Edit1.Text);
50     v:=StrToInt(Edit2.Text);
```

53: 37    Тільки для читання    ВСТ    C:\Uroki\8Klass\Основи\_програи

Чашик О.Ф., вчитель інформатики

Програмування

# Основи подійно- та об'єктно-орієнтованого програмування

## Значення властивостей об'єктів

Вивчаємо

Для звернення до властивостей об'єктів у програмному коді використовують такий спосіб запису:

***Ім'я форми. Ім'я об'єкта. Ім'я властивості***



Чащук О.Ф., вчитель інформатики

Програмування

# Основи подійно- та об'єктно-орієнтованого програмування

## Значення властивостей об'єктів

Вивчаємо

Якщо проект містить лише одну форму або розглядаються об'єкти поточної форми, то вказувати ім'я форми не обов'язково

```
Label1.Visible := True;
```

***Label1***  
***(Напис1)***



***Visible***

***(Видимість)***



Чашик О.Ф., вчитель інформатики

Програмування





# Основи подійно- та об'єктно-орієнтованого програмування

## Значення властивостей об'єктів

Вивчаємо

Властивість **Visible (Видимість)**  
може набувати одного з двох  
значень:

**True**

Об'єкт буде  
відображено



**False**

Невидимий або  
«прихований»  
об'єкт



Програмування

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»  
Інститут інформатики

# Основи подійно- та об'єктно-орієнтованого програмування

## Значення властивостей об'єктів

Вивчаємо

У вікні редактора коду середовища програмування Lazarus після введення імені об'єкта і крапки відкривається список доступних властивостей та дій, які можна описати для об'єкта. Щоб додати в код ім'я потрібної властивості, достатньо обрати її мишею зі списку

```
begin
  Label1.
end;

end.
```

property	Color: TColor
function	AdjustFontForOptimalFill: Boolean
property	Alignment: TAlignment
function	CalcFittingFontHeight (const Font: TFont): Integer
function	ColorIsStored: boolean
constructor	Create (TheOwner: TComponent): TLabel

Чашник О.Ф., вчитель інформатики

Програмування

# Основи подійно- та об'єктно-орієнтованого програмування

## Реагування об'єктів на події

Вивчаємо

Під час виконання програми об'єкти можуть «реагувати» на деякі події, зокрема, дії користувача:

В результаті настання деякої події, що передбачена для об'єкта, виконується певний набір команд

клацання мишею на об'єкті

натиснення на клавіатурі деякої клавіші чи комбінації клавіш

вибір деякої вказівки меню

зміну розмірів вікна ...

Чайчук О.Ф., вчитель інформатики

Програмування

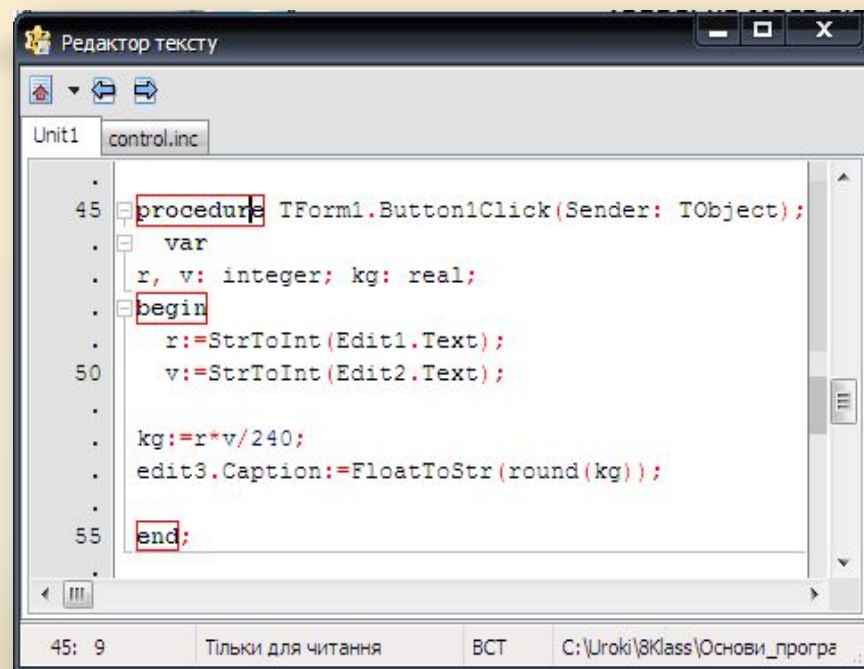


# Основи подійно- та об'єктно-орієнтованого програмування

## Реагування об'єктів на події

Вивчаємо

Щоб описати дії, які мають виконуватися при настанні деякої події для об'єкта, необхідно написати **програмний код — підпрограму**, яка в середовищі розпочинається зі службового слова



```
Unit1 control.inc
.
45 procedure TForm1.Button1Click(Sender: TObject);
.
.   var
.   r, v: integer; kg: real;
.
.   begin
.       r:=StrToInt(Edit1.Text);
50       v:=StrToInt(Edit2.Text);
.
.       kg:=r*v/240;
.       edit3.Caption:=FloatToStr(round(kg));
.
55   end;
```

Чашик О.Ф., вчитель інформатики

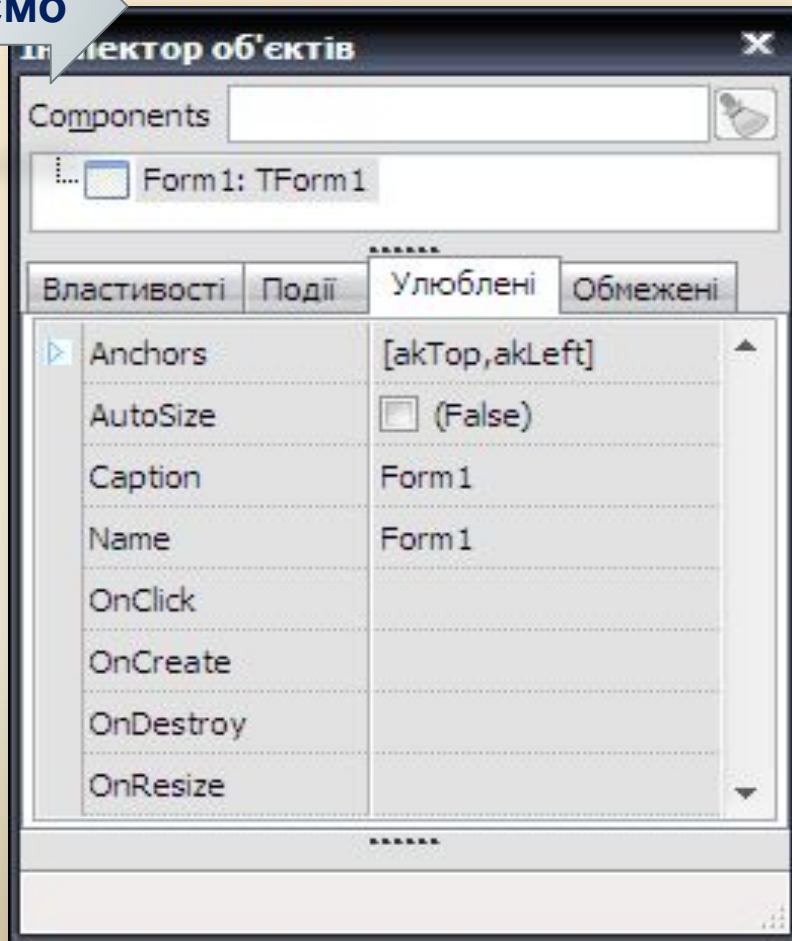
Програмування

# Основи подійно- та об'єктно-орієнтованого програмування

## Реагування об'єктів на події

Вивчаємо

Список усіх доступних подій, які можуть опрацьовуватися, для кожного об'єкта в середовищі програмування Lazarus подається в таблиці на вкладці



Чашук О.Ф., вчитель інформатики

Програмування

# Основи подійно- та об'єктно-орієнтованого програмування

## Реагування об'єктів на події

Вивчаємо

Список подій на вкладці  
*Улюблені*

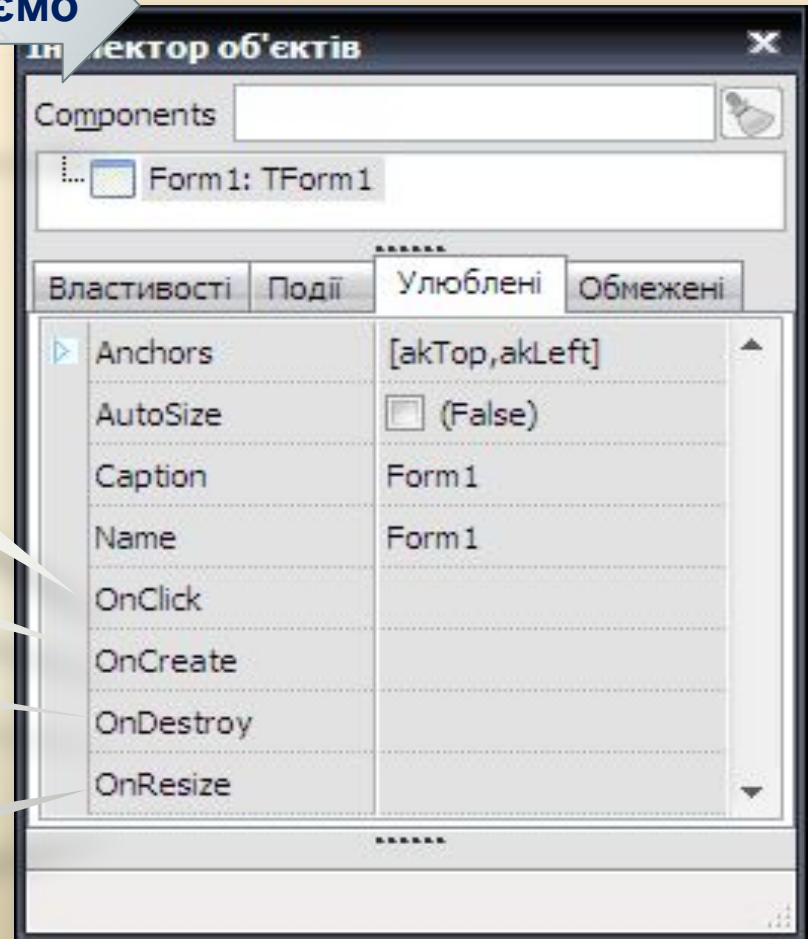
Подія:

Натиснення миші

Створення форми

Видалення форми

зміна розмірів вікна  
форми



Чащук О.Ф., вчитель інформатики

Програмування



# Основи подійно- та об'єктно-орієнтованого програмування

## Реагування об'єктів на події

Вивчаємо

Якщо двічі клацнути у клітинці таблиці справа від обраної події, то у вікні редактора коду з'являється фрагмент програмного коду для опису реакції на подію

Процедура

Ім'я об'єкта

Ім'я події

```
procedure TForm1.FormCreate(Sender: TObject);  
begin  
    |  
end;
```

Початок програмного коду

Кінець програмного коду

Чайник О.Ф., вчитель інформатики

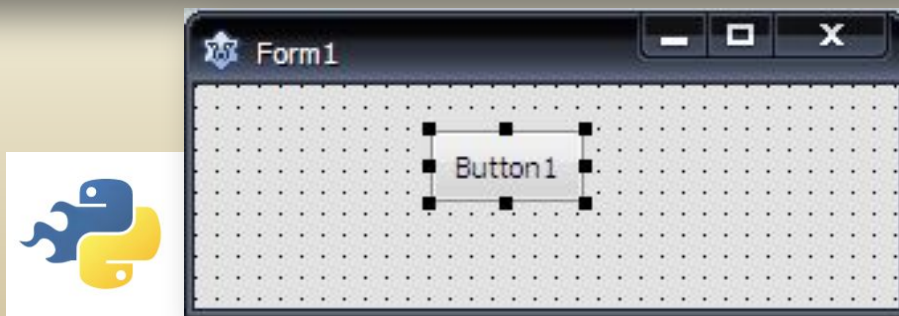
Програмування

# Основи подійно- та об'єктно-орієнтованого програмування

## Використання елемента управління *кнопка*

Вивчаємо

Кнопки, які створюють у середовищі програмування за допомогою компонента **Button** (*Кнопка*), використовують, що під час виконання програми після їх натиснення виконувався деякий набір команд



Чашик О.Ф., вчитель інформатики

Програмування

# Основи подійно- та об'єктно-орієнтованого програмування

## Використання елемента управління *кнопка*



Властивості

Вивчаємо



**Кнопки** мають властивості, призначення яких є аналогічним до властивостей інших об'єктів



Caption

підпис

Enabled

увімкнення

Font

шрифт

Height

висота

Width

Ширина

Left

відступ від краю екранної форми ліворуч

Name

ім'я

Visible

видимість

Чащук О.Ф., вчитель інформатики

Програмування



# Основи подійно- та об'єктно-орієнтованого програмування

## Використання елемента управління *кнопка*

Вивчаємо

З об'єктом кнопка найчастіше пов'язують подію **OnClick**. Щоб перейти до редактора програмного коду можна двічі клацнути на кнопці у вікні дизайне форми. У вікні редактора коду додається процедура опрацювання події – натиснення кнопки

Ім'я події натиснення кнопки *Button1*

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
end;  
end;
```

Початок програмного коду

Кінець програмного коду

Чашник О.Ф., вчитель інформатики

Програмування



# Домашнє завдання



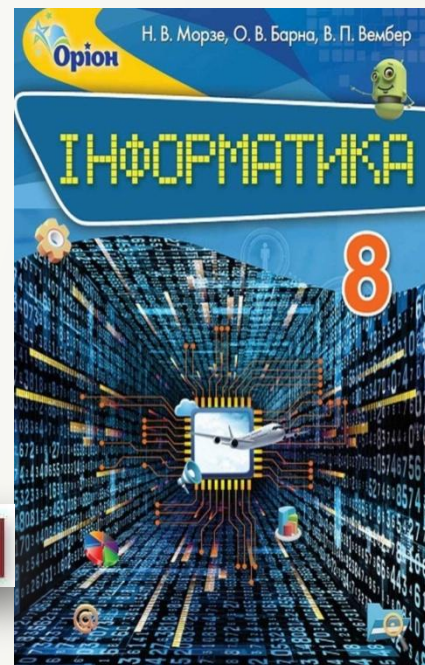
- ❑ Вивчити §18 с.120-124
- ❑ Опрацювати всі запитання і завдання з рубрик



ДІЄМО

ПРАЦЮЄМО САМОСТІЙНО

ПРАЦЮЄМО В ПАРАХ



- ❑ Заповнити словничок



Присвоювання, властивості об'єкта



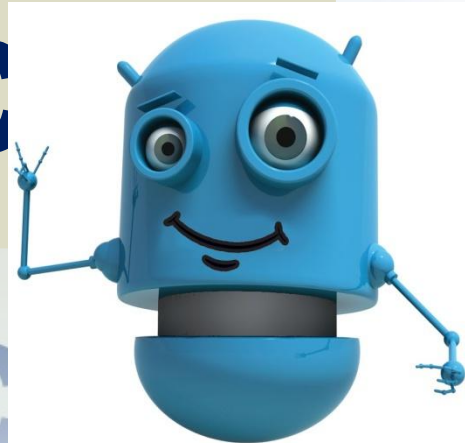
Розгада

й



ебус

Чашук О.Ф., вчитель інформатики ЗОШ №23,  
Луцьк





# Розгадай ребус



**КНОПКА**





# Робота з комп'ютером



ДІЄМО

**Вправа 1. Прості і складені числа**

**Вправа 2. Чарівні кнопки**





# Працюємо за комп'ютером

