

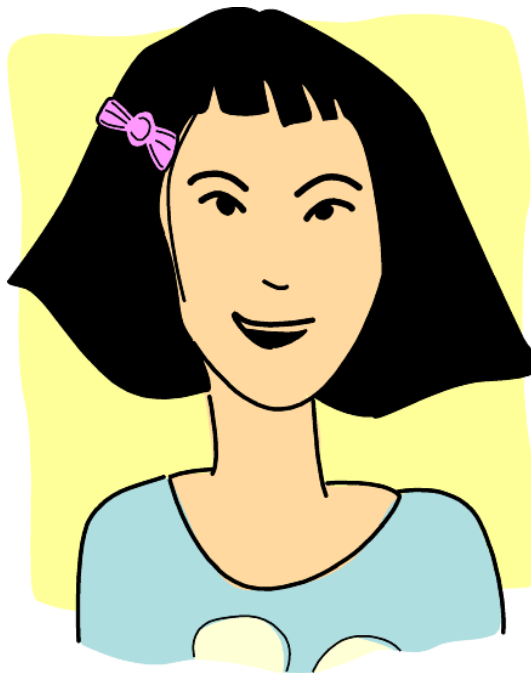


## Текстове и шрифтове

### Хелзинки, Финландия

Отправляме се към Хелзинки – столицата и най-големият град във Финландия. Разположен е в Южна Финландия, провинция Уусимаа, на северния бряг на Финския залив.

С Клиф и Ди Джей научихте доста за възможностите да рисувате във средата на Visual Basic 2010. Нещо обаче липсва. Какво ли е то? Питам се не е ли възможно да използваме писалките и четките за създаване на текстове с художествено украсени шрифтове. Аз мисля, че е възможно. Да проверим!





## Текстове

В графична среда вероятно една от най-скучните задачи – визуализирането на текст – е едно от най-сложните неща. Шрифтовете често са пропорционални и обикновено се доставят с разнообразни стилове и размери, което означава, че текстът трябва да бъде управляван по начин, подобен на другата графика. Все пак текстът не прилича толкова на линиите и запълнените области и затова управлението му изисква повече труд в компютърната графика. За печатаря, шрифтовете представляват произведение на изкуството с дълга история изкусен дизайн. Създателите на компютърни графични системи трябва да се справят с класическите типографски концепции, а програмистите на приложения да изучат тези концепции.

Най-важният принцип е, че текстът е предназначен за четене. Съществуват редица тънкости в дизайна на шрифта, изобразяването и разположението на страницата, които влияят на четивността дори, ако обикновеният потребител не ги забелязва съзнателно. Още повече, че текстът не е чисто съдържание. Неговият стил може да повлияе върху интерпретацията на съдържанието – положително или отрицателно. Ето защо в тази секция ще научим повече неща за шрифтовете.

## Същност на шрифта

Печатарите по правило означават конкретен шрифт с име и рисунък на шрифта (typeface name) и размер в пунктове. Всеки рисунък на шрифт принадлежи към дадена фамилия. Фамилиите от рисунъци имат прости имена като Bookman, Helvetica, Garamond и Times. Всяка фамилия често включва няколко варианта:



- Гредичките, които образуват знаковете, могат а бъдат тънки или дебели в различни степени, описвани в имената на шрифтовите рисунъци, като Helvetica Ultra Light, Helvetica Thin, Helvetica Light, Helvetica Bold, Helvetica Heavy и Helvetica Black.
- Шрифтовете на отделните знакове могат да бъдат по-тесни или по-широки от обикновено, при имена на шрифтови рисунъци като Helvetica Narrow, Helvetica Condensed и Helvetica Expanded.
- Знаковете могат да бъдат наклонени надясно, давайки ни имена на рисунъци като Helvetica Italic или Helvetica Oblique. Строго дефинирано, полегати (oblique) означава, че знаковете просто са наклонени, докато наклонени (italic) означава, че освен това знаковете са стилистично донякъде различни от съответния изправен шрифт.

Тези три варианта могат да бъдат комбинирани в едно отделно име на рисунък на шрифт – например Helvetica Bold Extended Oblique. Имената на рисунъците могат също да включват името на техния създател.

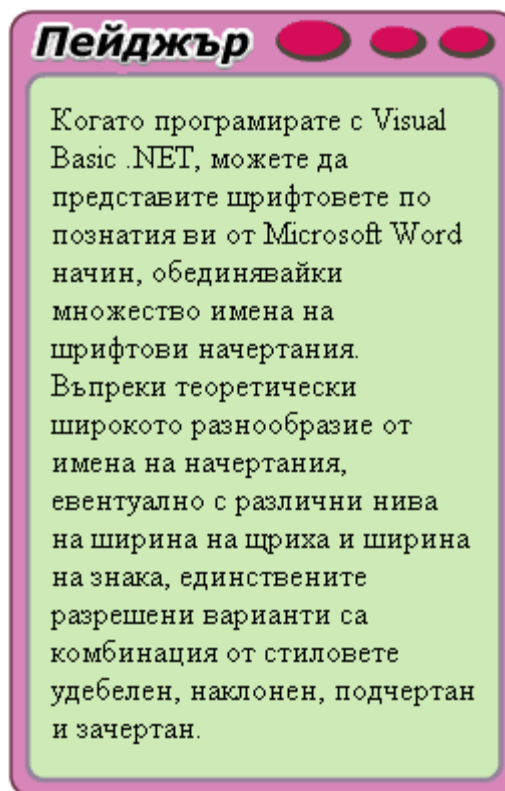


Може би най-познатите ви шрифтове са Courier New, Times New Roman, Arial и Symbol.

Courier е фамилия шрифтове с фиксирана стъпка, която наподобява текст, отпечатан на пишеща машина. Днес много малко текст се изписва с Courier. Изключение са само прозорците с команден ред, листингите на програми и шестнайсетичен код.

Times New Roman е клонинг на шрифта Times, първоначално създаден от Times of London и използван в много печатни материали. Той се счита за високо четивен. Arial е клонинг на Helvetica – популярен несерифен шрифт. Серифите представляват малки напречни чертички, с които завършват гредичките на буквите. Несерифният шрифт няма серифи. Шрифтът Symbol вместо букви включва често използвани специални знакове и символи.

Често в графичните среди обикновено думата шрифт се използва за означаване на фамилия от шрифтове. Освен това шрифтовите начертания наклонен и удебелен се представят като атрибути или стилове на конкретен шрифт.





## Височини на шрифтовете

Наред с името на шрифтовия рисунък, шрифтът се определя с помощта на вертикален размер с пунктове (точки). В традиционната типография 1 пункт е  $1/72$  от инча.

Размерът на шрифт се описва най-често като височина на знаковете в латинската азбука. Това са главните и малките букви от А до Z, без диакритичните знакове – от тези с най-възходящи детайли до тези с най-низходящи детайли.

Запознаването с най-често срещаните размери в пунктове е полезно, когато започвате да работите с шрифтовете. Подразбиращият се шрифт в Windows е 10 пункта. Подразбиращият се шрифт на формите е около 8 пункта. Размерът в пунктове понякога се означава като "em" размер. Терминът идва от размера на квадратното парче метал, използвано в стари времена за главната буква М. Днес с "em" се означават най-често хоризонтални мерки. Ширината на един "em" в конкретен шрифт е равна на хоризонталния размер в пунктове на шрифта.

Последователните редове на текста обикновено се подреждат на разстояние по-голямо от размера в пунктове. Това се налага поради наличието на диакритични знакове. Междуредието е и естетически необходимо, тъй като текстът се чете по-лесно, ако между редовете има малко "въздух".



## Създаване на шрифт

Сигурно сте забелязали, че повечето контроли имат свойство Font. То притежава набор от характеристики, които са описани в таблицата по-долу:

Характеристика	Описание
Name	Име на шрифтовия рисунок.
Size	Размер на шрифта.
Unit	Мерна единица.
Bold	Удебелен текст.
Italic	Наклонен текст.
Strikeout	Зачертан текст.
Underline	Подчертан текст.

Когато извеждате текст върху формата с метода DrawString, вие използвате шрифт по подразбиране. Можете да промените шрифта като създадете такъв по един от следните начини:

### Полезен съвет

Размерът на шрифтовете може да се задава в различни мерни единици – пиксел, инч, милиметър. Характеристиката Unit служи за

#### 1. Използване на съществуващ шрифт

```
Dim font = Me.Font
Dim fontItalic As New Font(font, FontStyle.Italic)
grfx.DrawString("Наклонен шрифт", fontItalic, Brush, 20, 20)
```

В примера по-горе първоначално се създава шрифт еднакъв с този на формата. След това от него се създава втори шрифт, който е наклонен. За целта се задава стил със стойност Italic. В таблицата по-долу можете да видите възможните стилове при дефиниране на шрифтове.

Стил	Описание
Regular	Нормален
Bold	Удебелен
Italic	Наклонен
Underline	Подчертан
Strikeout	Зачертан

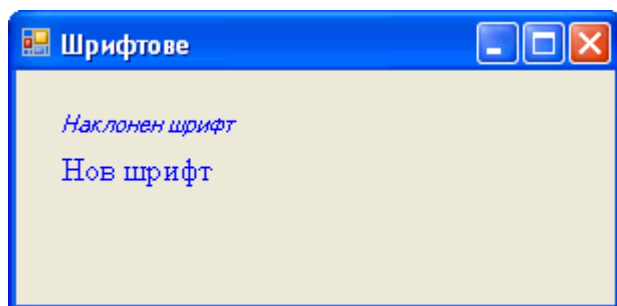
#### 2. Създаване на шрифт по име



```
Dim fontTimes As New Font("Times New Roman", 12)
grfx.DrawString("Нов шрифт", fontTimes, brush, 20, 40)
```

В този пример шрифтът се създава като се задава име ("Times New Roman") и размер в пунктове (12).

Ето как изглеждат текстовете, изписани със създадените шрифтове:





## Размер в пунктове

Ако вие е трудно да си представите мерната единица пункт, не е задължително да я използвате. В таблицата по-долу са изброени някои от мерните единици, които можете да ползвате:

Мерна единица	Описание
World	Мерна единица в пространствени координати.
Pixel	Мерна единица пиксел.
Point	Мерна единица 1/72 инча.
Inch	Мерна единица инчове.
Document	Мерна единица 1/300 инча.
Millimeter	Мерна единица милиметри.

Не се тревожете, ако някои от мерните единици не са ви ясни. Те са дадени, за да знаете за тяхното многообразие.

Следните редове с код дават един и същ резултат:

### Полезен съвет

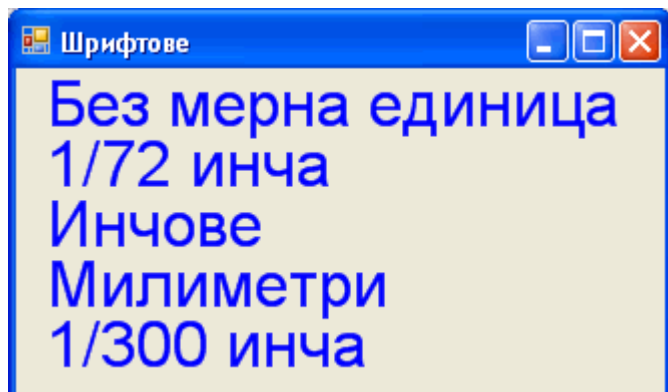
Накарайте учениците си да изберат мерната единица, която познават най-добре.

```
Dim font1 As New Font("Arial", 24)
Dim font2 As New Font("Arial", 24, GraphicsUnit.Point)
Dim font3 As New Font("Arial", 1 / 3.0F,
GraphicsUnit.Inch)
Dim font4 As New Font("Arial", 25.4F / 3, _
GraphicsUnit.Millimeter)
Dim font5 As New Font("Arial", 100,
GraphicsUnit.Document)
```

```
grfx.DrawString("Без мерна единица", font1, brush, 10,
0)
grfx.DrawString("1/72 инча", font2, brush, 10, 30)
grfx.DrawString("Инчове", font3, brush, 10, 60)
grfx.DrawString("Милиметри", font4, brush, 10, 90)
```



```
grfx.DrawString("1/300 инча", font1, brush, 10, 120)
```





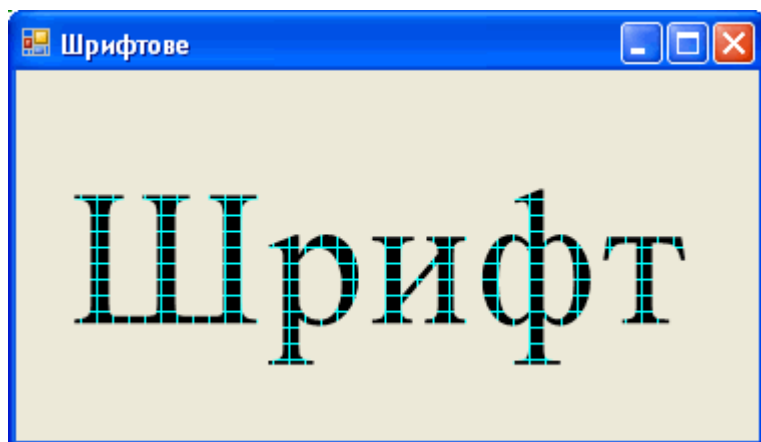
## Забавление с шрифтове

Тъй като знаковете на шрифта се определят с помощта на поредица от прави линии и криви, те се интегрират добре с останалата графичните инструменти от предходните две секции. Знаковете на шрифта могат да бъдат оцветени с всяка четка. Текстът би могъл да надмине неговата обичайна роля – да бъде четен – и да се стреми да стане нещо като изкуство.

Вече знаете как да създадете и използвате четка с шарка и шрифт. Остана само да комбинирате знанията си ето така:

```
Dim brush As New HatchBrush(4, Color.Aqua, Color.Black)
Dim font As New Font("Times New Roman", 72)
grfx.DrawString("Шрифт", font, brush, 10, 40)
```

Резултатът от изпълнението на горния код е следният:



Няма да ви отегчавам с примери за комбиниране на шрифтовете с останалите четки. Смятам, че вие ще се справите с това при решаването на задачите, които ще получите по-нататък.

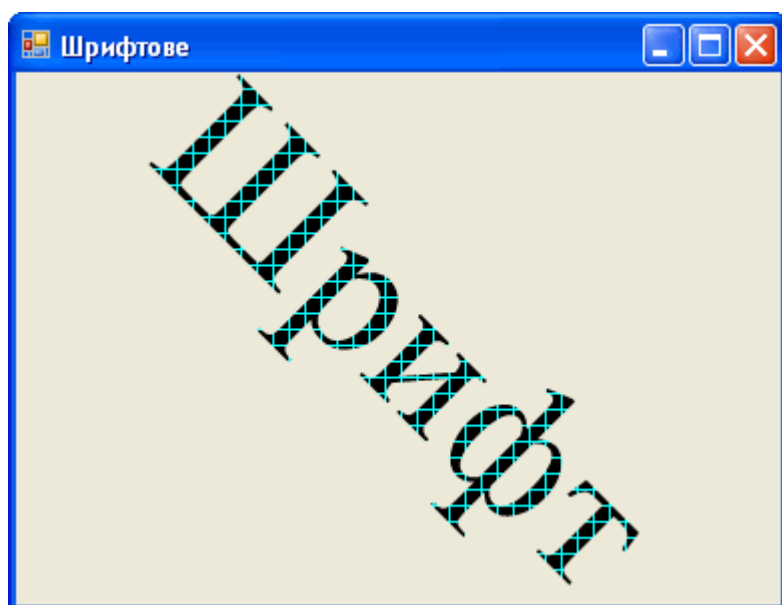


## Трансформации с шрифт

Текстът е обект на същите ефекти на мащабиране, ротация и навеждане като всеки друг графичен обект. Методът `RotateTransform` извършва завъртане на определен ъгъл. Ъгълът се задава като аргумент на метода. Ако искате да завъртите текста от предходния пример на 45 градуса преди изписването на текста с метода `DrawString` поставете следния ред:

```
grfx.RotateTransform(45)
```

Ще получите следния резултат:





Възможно е да "наведете" текста по определен от вас. Това става с метода `ScaleTransform`. Той работи с два аргумента, които могат да примат само стойности 1 и -1. Комбинацията от тях определя как ще се наведе текста.

Например, ред с код:

```
grfx.ScaleTransform(-1, -1)
```

води до следния резултат:



При използването на функциите `RotateTransform` и `ScaleTransform` можете да не получите резултат. Формата ви в този случай ще бъде празна. Това се дължи на обстоятелството, че след трансформацията изписаният текст остава извън границите на формата. Можете да зададете средата на формата като начална точка трансформация по следния начин:

```
grfx.TranslateTransform(Me.Size.Width/2, Me.Size.Height/2)
```

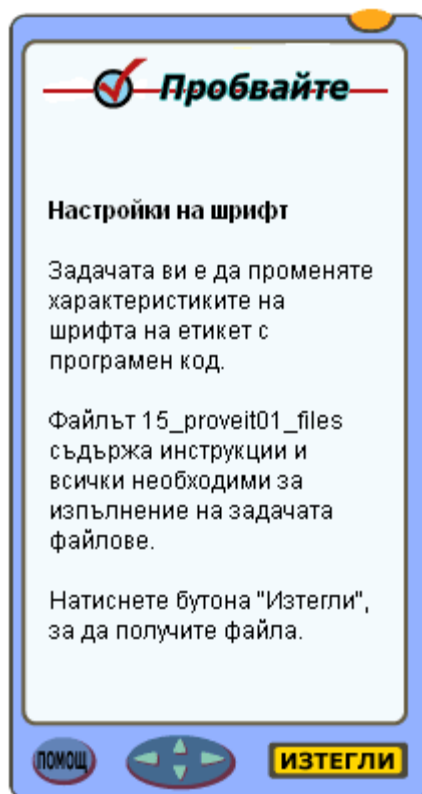
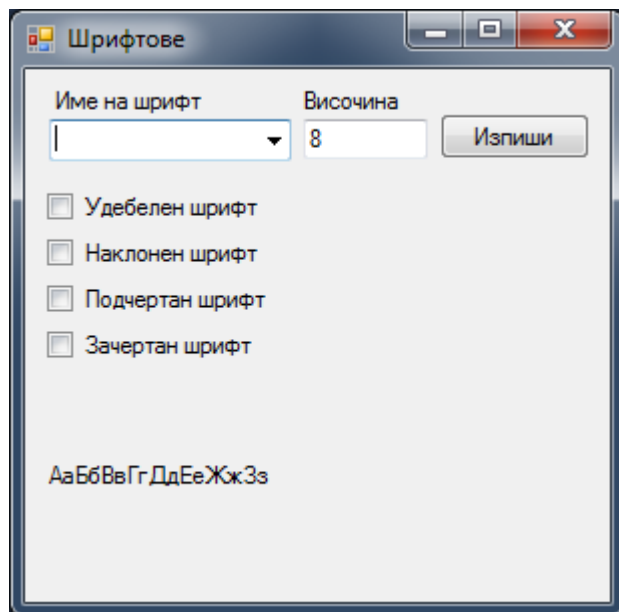
#### **Полезен съвет**

Когато искате да получите достъп до някои от свойствата на формата, използвайте обекта "Me". Обяснете неговото предназначение на учениците без да навлизате в терминологията на класовете и обектите.

## 15 Пробвайте 01    Настройки на шрифт

А сега да проверим наученото.

Създайте форма, която изглежда така:



Попълнете падащото меню с имена на шрифтове, които познавате. В обработчика на събитието Click декларирайте следните променливи:

Променлива	Тип
FontName	String
FontStyle	System.Drawing.FontStyle
FontSize	Integer

Променливата FontSize инициализирайте със стойността на свойството Text на текстовото поле.

Променливата FontStyle инициализирайте в зависимост от маркираното поле за отметка.

Променливата FontName инициализирайте с име на шрифтова фамилия в зависимост от индекса на избрания елемент от падащото меню.

Създайте шрифт като използвате трите променливи.

Присвоете създадения шрифт на свойството Font на етикета.



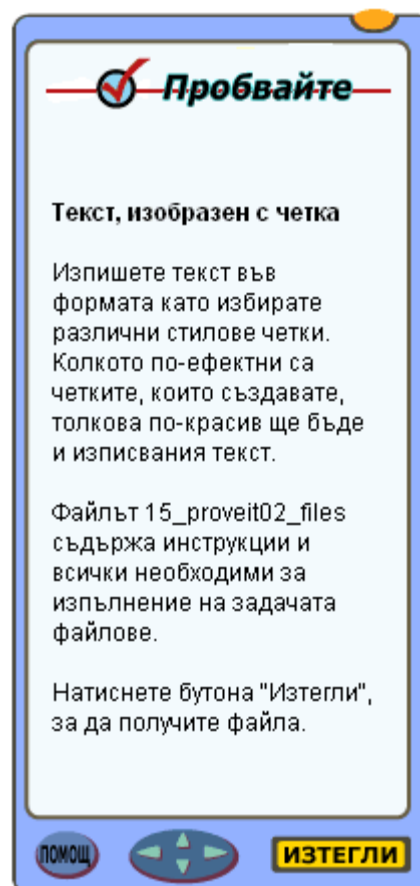
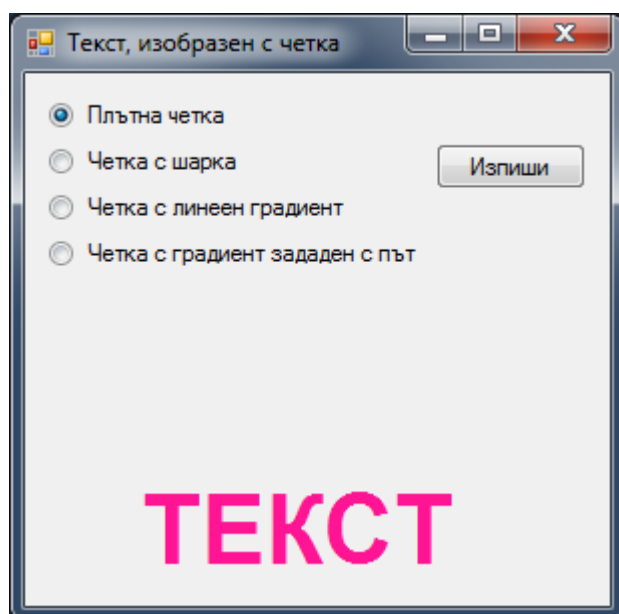
Изградете проекта и ако няма грешки изберете опцията Start Debugging от менюто Debug, за да стартирате програмата.

Ако програмата работи правилно, я покажете на вашия учител.

## 15 Пробвайте 02 Текст, изобразен с четка

За тези, които обичат да рисуват тази задача може би ще е по-интересна.

Създайте форма, подобна на следната:



В обработчика на събитието Click на бутона извикайте метода Invalidate.

В обработчика на събитието Paint създайте удебелен шрифт с по-голям размер, например 36. В зависимост от маркирания бутон създайте плътна четка, четка с шарка, с линеен градиент или с градиент, зададен с път. Изпишете примерен текст със създадения шрифт и съответната четка.

Изградете проекта и ако няма грешки изберете опцията Start Debugging от менюто Debug, за да стартирате програмата.

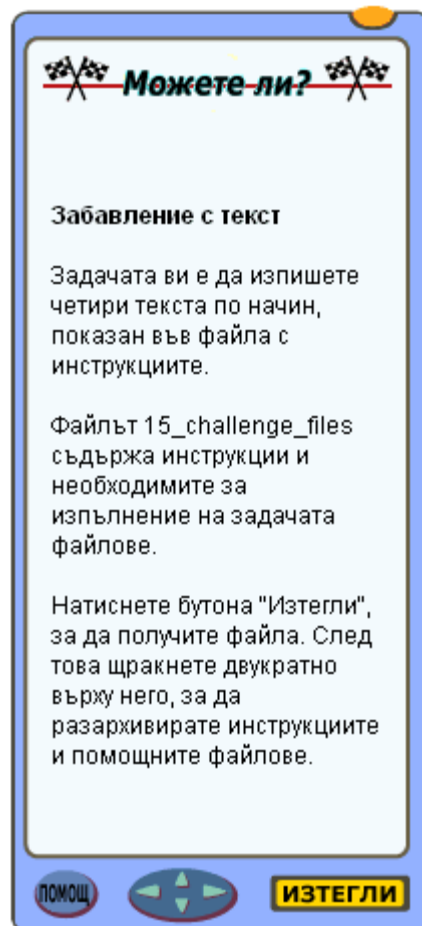
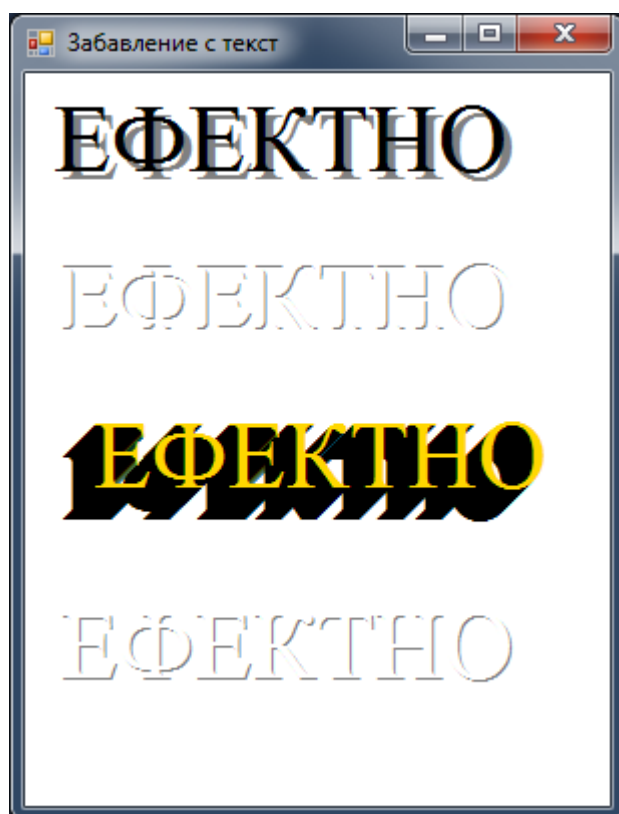
Ако програмата работи правилно, я покажете на вашия учител.



## 15 Можете ли? Забавление с текст

А сега е време за едно истинско предизвикателство!

Ефектите са постигнати само чрез отнемстване с няколко пиксела на изчертавания текст. Ако се затруднявате, потърсете помощта на вашия учител.



Изградете проекта и ако няма грешки изберете опцията Start Debugging от менюто Debug, за да стартирате програмата.

Ако програмата работи правилно, я покажете на вашия учител.

- 1

Какво се измерва с мерната единица пункт при работа с текстове?  
☐ А. Височината на шрифта  
☐ В. Ширината на буквите  
☐ С. Ширината на символните низове
- 2

Какво прави следния ред с код:  
`Dim f As New Font("Times New Roman", 12)?`  
☐ А. Изписва текст "Times New Roman".  
☐ В. Създава шрифт с име Times New Roman и височина 12 сантиметра.  
☐ С. Създава шрифт с име Times New Roman и височина 12 пункта.
- 3

Каква е ролята на метода RotateTransform?  
☐ А. Транслира текст.  
☐ В. Завърта текст на определен ъгъл.  
☐ С. Мащабира текст.
- 4

Какви аргументи приема методът DrawString?  
☐ А. Символен низ, шрифт, четка, координата x и координата y  
☐ В. Символен низ и шрифт  
☐ С. Символен низ, шрифт, четка и дължина на низа