

УДК 65.262.1

С.Б. Данилевич, В.А. Кирвас

Харьковский гуманитарный университет «Народная украинская академия»

СОЗДАНИЕ УЧЕБНОЙ ДЕЛОВОЙ ИГРОВОЙ ПРОГРАММЫ СРЕДСТВАМИ VBA MS EXCEL

Приведены рекомендации по созданию деловой компьютерной игры с применением VBA MS Excel.

информационные технологии, деловая игра, редактор VBA

Введение

Внедрение информационных технологий в образование позволяет расширять объемы учебной информации и уменьшить время ее усвоения. Применение в учебном процессе электронных учебников, компьютерных тестирующих, тренинговых программ, учебных деловых игр и т.п. [1 – 4] повышает мотивацию к изучению дисциплины, способствует активному восприятию студентами учебной информации.

Несмотря на наличие большого количества специализированных обучающих программ закупка, установка и освоение сложных и дорогих средств практически недоступна большинству педагогов. В то же время MS Excel остается наиболее распространенной компьютерной программой. Являясь универсальной, она позволяет моделировать довольно сложные процессы, встречающиеся на практике, применять теоретические методы анализа. Часто не имея возможности привлечь специалистов, педагоги "обречены" быть универсалами, постоянно постигать новые области знаний, учиться применять новейшие информационные технологии, в том числе осваивать методы имитационного моделирования. К преимуществам компьютерного моделирования [5] относится возможность создания наглядных образов, что важно для формирования многих научных понятий, выявить свойства и закономерности исследуемых систем, решить конкретные практические задачи. Применяя редактор Visual Basic (VBA) [6, 7], можно расширить функциональные возможности Excel в направлении создания программ для деловых игр.

Целью статьи является разработка методических и практических рекомендаций по созданию деловой игровой программы средствами электронных таблиц MS Excel.

Изложение основного материала

Формулировка игры. На компьютере в соответствующем диалоговом окне представляются случайным образом цитаты известных экономистов. Необходимо определить их авторов.

Действия пользователя. Выбрать автора, щелкнув по кнопке с его фамилией левой кнопкой компьютерной мыши.

Создание базы данных цитат и их авторов.

Список цитат и их, например, 16-ти авторов [8] целесообразно поместить на отдельный лист *Цитаты* в книге MS Excel (рис. 1).

Проектирование интерфейса пользователя.

1. В редакторе VBA (клавиши *Alt+F11*) выбрать команду меню *Insert – UserForm*.

2. Для полученной заготовки формы на панели свойств (рис. 2) (команда *Properties Window (F4)* на панели инструментов редактора Visual Basic) дадим имя (*Name*) форме (например, *GEForm*) и установим свойства: *Height = 345* (высота) *Width = 300* (ширина), *Capture* (заголовок) (например, *Великие экономисты*). *BackColor* (цвет фона) – выбрать из палитры цветов, *ShowModal* – выбрать *False* (немодальность формы позволяет выходить из нее, не завершив диалога).

3. Используя стандартную панель инструментов (рис. 3) *ToolBox* создадим элемент управления *Label* (надпись). Дадим ему имя (например, *Quotation*) установим свойства: *Height = 225* (высота), *Width = 190* (ширина), *BackColor* (цвет фона) из палитры цветов, *Capture* (заголовок) (например, *Quotation*). Поместим его в левом верхнем углу формы.

4. Создадим всего 16 (по числу авторов цитат) элементов управления *Label* (надпись) один под другим в правой части формы (рис. 4). Выровним положение надписей по правому краю, выделив все надписи (щелкнув мышкой сначала по верхней, а

	A	B	C	D	E
1	№ п/п	Авторы	Цитата 1	Цитата 2	Цитата 3
2	1	Ф. Кенз	Государство, населенное одними лишь торговцами и ремесленниками, может существовать только за счет доходов от земельной собственности, получаемых за границей.	Основная задача правительства заключается в облегчении при помощи развития торговли сбыта продовольственных товаров собственного производства.	Портной, шьющий одежду, сапожник, выпускающий обувь, создают не больше богатств, чем музыканты, дающие концерт.
3	2	А. Тюрго	Ошибка, когда моралисты рассматривали отдачу в рост как преступление.	Цену процента можно рассматривать как своего рода уровень, ниже которого прекращается всякий труд, всякая обработка земли, всякая промышленность, всякая торговля.	Заем под проценты есть не что иное, как торговля, в которой заимодавец - человек, продающий пользование своими деньгами, а заемщик - человек, покупающий его.
4	3	Ш. Фурье	При социетарном строе гурманство является источником мудрости, просвещения и дел социального согласия.	Социетарный кантон будет вести переговоры о купле-продаже только один единственный раз вместо трехсот противоречивых переговоров, занимающих триста глав семейств.	Прибыльная часть торговли - ростовщичество. Надо, чтобы правительство завладело этой отраслью при посредстве казенных форм.
	4	Н. Чернышевский	Общинное владение - это единственный род собственности, охранение которого не доставляет правительству никаких забот и не требует от него ровно никаких	С развитием просвещения и здравого взгляда на жизнь будут постоянно ослабевать до нуля разные слабости и	Их всех родов собственности общинное владение есть тот род, который наиболее предохраняет частную жизнь от административного

Рис. 1. Пример оформления данных на листе *Цитаты*

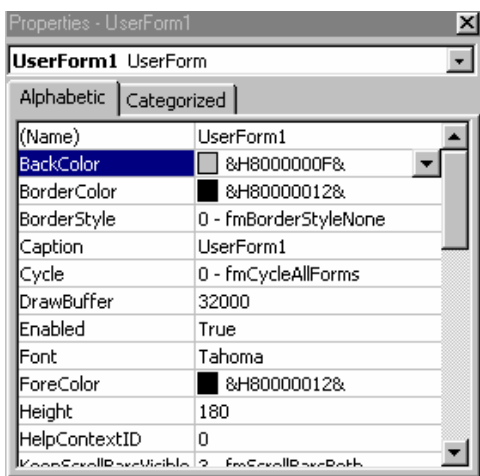


Рис. 2. Панель свойств



Рис. 3. Панель инструментов

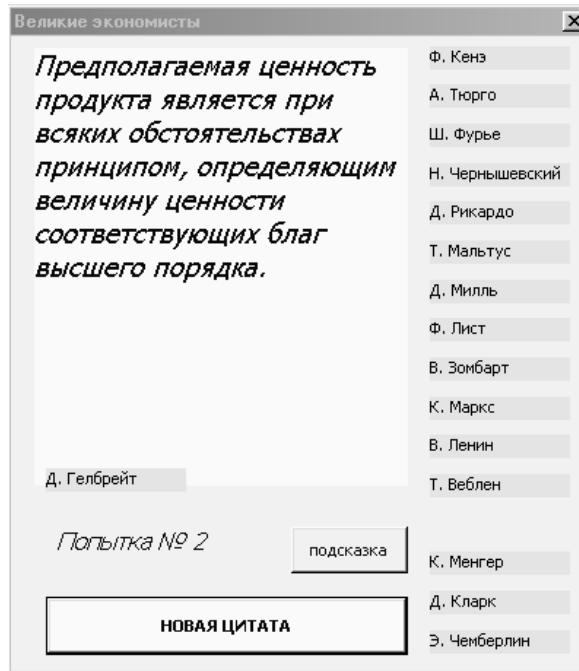


Рис. 4. Макет формы

Введение программных объектов и их инициализация.

1. В меню *Tools* выберем команду *Options*. В затем при нажатой клавише *Shift* по нижней), дадим команду меню *Format – Make Same Size – Both*. Можно воспользоваться и правой кнопкой мыши. Дадим имена надписям (например, *Author1 – Author16*) и установим свойства: *Heigh - 12t* (высо-

та), *Width* = 72 (ширина), *BackColor* (цвет фона) из палитры. появившемся диалоговом окне на вкладке *Editor* установим *Require Variable Declaration* (требование: правильно объявлять и в дальнейшем использовать переменные).

2. Создадим стандартный модуль (меню *Insert – Module*) и на панели свойств модуля изменим *Name* с *Module1* на *GEModule*.

3. Закрепим программно начальные положения надписей, поскольку в дальнейшем они будут меняться. Положение цитаты изменяться не будет. Чтобы указать, что надписи принадлежат форме и чтобы сократить текст программы, запишем конструкцию:

```
With GEFForm
```

```
...
```

```
End With
```

4. Между *With GEFForm* и *End With* поместим текст:

```
.Author1.Left = 215
```

```
.Author1.Top = 5
```

Первый объект расположим слева сверху формы. Конкретное значение свойств объекта иметь положение слева (*.Left*) и сверху (*.Top*) можно определить в окне свойств, переместив объект в нужное положение. Аналогично для остальных.

```
.Author2.Left = 215
```

```
.Author2.Top = 25
```

```
.Author3.Left = 215
```

```
.Author3.Top = 45
```

```
...
```

```
.Author16.Left = 215
```

```
.Author16.Top = 305
```

5. Чтобы в дальнейшем можно было изменять фамилии авторов и соответственно цитаты, привяжем заголовки объектов надписи *Author1 – Author16* и *Quotation* к соответствующим ячейкам *Excel*, содержащим соответствующие фамилии авторов и их цитаты. После текста программы, в котором указывалось положение надписей, запишем (воспользуемся возможностью копировать с помощью мыши перетаскивая выделенный текст при нажатой кнопке *Ctrl*):

```
.Author1.Caption = Worksheets("Авторы").Cells(2, 2)
```

```
.Author2.Caption = Worksheets("Авторы").Cells(3, 2)
```

```
...
```

```
.Author16.Caption = Worksheets("Авторы").Cells(17, 2)
```

6. Цитата должна выбираться случайным образом. Воспользуемся встроенной функцией *Rnd* (от английского слова *random* – беспорядочный). Выбор цитаты будет осуществляться из диапазона ячеек *C2 – C17*. Чтобы указать номер строки объявим переменную *i* сразу после инструкции *Option Explicit*, так как она будет использоваться и в других модулях:

Public i As Integer 'Объявляется глобальная целочисленная переменная, которая будет принимать случайные значения.

7. После слов *Public Sub InitialStates()* зададим начальное значение:

```
i = 0.
```

8. Встроенная функция *Rnd* принимает значения в диапазоне от 0 до 0,99... Необходимый диапазон строк: от 2 до 17. Потому используем выражение: $i = \text{Fix}((16 * \text{Rnd}) + 2)$, где встроенная функция *Fix* отбрасывает десятичные знаки. Так, если $\text{Rnd}=0,99$ (максимальное значение), то $(16 * \text{Rnd}) + 2 = 17,84$, а $\text{Fix}(17,84) = 17$.

9. Аналогично, для столбцов *Excel* объявим переменную *k*, которая будет принимать значения от 2 (столбец В) до 5 (столбец Е) и запишем код:

```
k = Int((3 * Rnd) + 3).
```

Здесь воспользуемся функцией *Int*, которая округляет до целых аргумент.

10. Случайный выбор цитаты осуществим выражением:

```
Quotation.Caption = Worksheets  
("Цитаты").Cells(i, k).
```

11. Введем начальные состояния объектов *Author1 – Author16*. С помощью команды меню *Insert – Procedure...* введем процедуру *InitialStates()*:

Процедура *InitialStates()* будет вызываться на события инициализации и активизации формы:

```
Private Sub Author2_Click()
```

```
Call Choice(Me.Author2)
```

```
End Sub
```

```
Private Sub UserForm_Activate()
```

```
InitialStates
```

```
End Sub
```

Выбор автора цитаты щелчком мыши

1. С помощью команды меню *Insert – Procedure...* введем процедуру для перемещения объектов *Author1 – Author16* с правой части формы в место расположения цитаты:

```
Public Sub CoiceAuthor()
```

```
'Помещение предполагаемого автора к цитате
```

```
St = "Left" 'Изменяем состояние
```

```
With GEFForm
```

```
'Меня координаты надписи, помещаем ее в цитату
```

```
Lb.Left = .Quotation.Width - Lb.Width
```

```
Lb.Top = .Quotation.Height - Lb.Height
```

```
End With
```

```
End Sub
```

2. Щелкнув на форме по объекту *Author1*, получим обработчик события *Click*:

```
Private Sub Author1_Click()
```

'Вызов процедуры *Choice* с указанием конкретного объекта и переменной.

```
Call Choice(Me.Author1)
```

```
End Sub
```

3. Аналогично и для остальных объектов. Для облегчения работы пользуйтесь возможностями копировать и заменять значения (*Edit – Replace...*).

4. В случае, если мы захотим изменить автора, обеспечим связь между соответствующими ячейка-

ми и надписями формы. Для этого в процедуру инициализации добавим

With GForm

.Author1.Caption = Worksheets("Цитаты").Range("b2")

...

.Author16.Caption = Worksheets("Цитаты").Range("b17")

End With

Тестирование состояния и организация диалога.

Создадим форму с поздравлением (рис. 5), которая будет появляться при правильном ответе.



Рис. 5. Форма с поздравлением

1. В редакторе VBA (*Alt+F11*) дайте команду меню *Insert – UserForm*.

2. Для полученной заготовки формы на панели свойств (команда *Properties Window* на панели инструментов) дайте имя форме (например, *Congratulation*) и установите свойства: *Height = 148,5* (высота) *Width = 240* (ширина), *Capture* (заголовок) (например, *Congratulation*). *BackColor* (цвет фона) из палитры цветов.

3. Создайте элемент управления *Label* (надпись) со стандартной панели инструментов *ToolBox*, установите свойства: *Height* (высота), *Width* (ширина), *BackColor* (цвет фона) из палитры цветов, *Capture* (заголовок) (например, *Поздравляю*). Разместите его в левом верхнем углу формы. Можно заготовить рисунок и указать его адрес в свойстве *Picture*.

4. Создадим кнопку для скрытия формы с поздравлением. Установите соответствующие свойства (например, заголовок кнопки: «Заккрыть») и в модуле этой формы запишите код:

```
Private Sub CommandButton1_Click()
```

```
Me.Hide
```

```
End Sub
```

5. В модуле формы *GEModule* создайте кнопку, при нажатии на которую появится новая цитата и форма примет исходное состояние (*InitialStates*):

```
Private Sub CommandButton1_Click()
```

```
InitialStates
```

```
End SubEnd Sub
```

6. Для вызова главной формы игры создадим макрос:

```
Sub OpenGForm()
```

```
GForm.Show
```

```
End Sub
```

Установим на какой-либо панели инструментов (*Форматирование* или *Стандартная*) или в главном меню программы кнопку запуска формы. Для этого дадим команду меню *Сервис – Настройка*. В появившемся диалоговом окне выберем вкладку *Команды* и в ее левой части выберем команду *Макросы*. Перетащим левой кнопкой мыши значок кнопки из правой части окна на выбранную панель инструментов.

Выводы

На основе анализа актуальных задач в образовании сделан вывод что, внедрение новых компьютерных технологий и в частности деловых компьютерных игр позволяет расширять объемы учебной информации и уменьшить время ее усвоения.

Разработаны методические и практические рекомендации по созданию деловой игровой программы средствами MS Excel с применением редактора VBA. Использование данных рекомендаций поможет преподавателям создание пособий для самостоятельной работы студентов по соответствующим темам учебной дисциплины.

Список литературы

1. Польщикова О.Н. Деловая игра как метод активного обучения на уроках информатики // Информатика и образование. – 2004. – № 5. – С. 49-55.
2. Змиевская Е.В. Учебная деловая игра в организации самостоятельной работы студентов педагогических вузов. – Дисс.... кандидата педагогических наук. – М., 2003. – 169 с.
3. Кочеткова А.И. Сборник методических разработок по проведению семинара «Деловая игра». Вып. 1. – М.: Финансовая академия при Правительстве РФ, 2000.
4. Дж. фон Нейман, О. Моргенштерн. Теория игр и экономическое поведение. – М.: Наука, 1970. – 708 с.
5. Цисарь И.В., Нейман В.Г. Компьютерное моделирование экономики. – М.: "Диалог-МИФИ", 2002. – 304 с.
6. Данилевич С.Б., Кирвас В.А. Имитационное моделирование ценообразования монополии средствами Ms Excel (VBA). БИЗНЕСИНФОРМ. – X.: ИД «ИНЖЭК». – 2005. – № 7-8. – С. 58-61.
7. Биллиг В.А. Средства разработки VBA-программиста. Офисное программирование. Т. 1. – М.: Издательско-торговый дом "Русская Редакция", 2001. – 480 с.
8. Корнейчук Б.В. Микроэкономика. – С.-Пб.: Питер, 2003. – 157 с.

Поступила в редколлегию 9.03.2007

Рецензент: д-р техн. наук, проф. Е.И. Бобыр, Харьковский гуманитарный университет «Народная украинская академия», Харьков.