



Bài tập 4 – C#

Windows Form Application - Basic



Mục tiêu:

- Sử dụng Visual Studio .NET 2005 (hoặc 2008) tạo ứng dụng dạng Windows Forms.
- Làm quen với việc sử dụng các control cơ bản trên form chính
 - o CheckBox: cho phép user chọn một option
 - o Label: hiển thị các thông tin chỉ dẫn
 - o Button: cho phép user kích chọn để gọi chức năng cài sẵn
 - o TextBox: hộp nhập liệu thông tin
 - o NumericUpDown: sử dụng nút up down để nhập giá trị số
 - o MessageBox: hiển thị thông tin đến user
- Khai báo trình xử lý sự kiện Click cho button trên Form: viết code xử lý cho trình xử lý sự kiện Click.

Nội dung:

- Tạo một ứng dụng Windows Form cơ bản tính tiền công dịch vụ cho một lần đi khám tại phòng nha.
- Với mỗi khách hàng, các dịch vụ cung cấp gồm: tẩy răng, cạo vôi, chụp hình răng và trám răng. Mỗi loại sẽ có chi phí riêng. Cuối cùng tính tổng các chi phí mà người khách phải trả. Lưu ý: chỉ tính tiền khi phân thông tin tên khách hàng đã

được nhập (nếu thông tin này chưa có thì chương trình phát sinh MessageBox cảnh báo).

- Ứng dụng có giao diện đơn giản như hình 1 bên dưới.

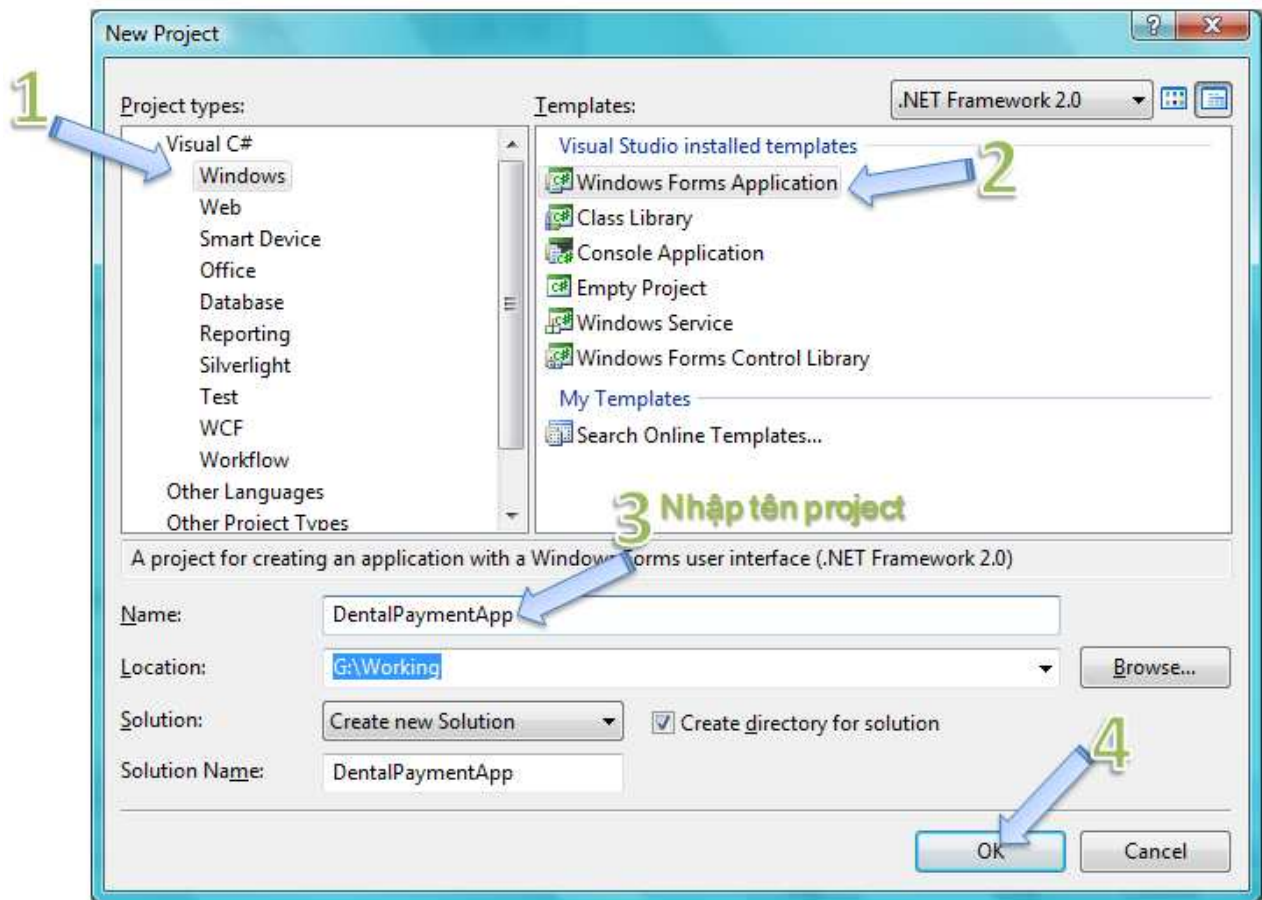
Service	Price
<input type="checkbox"/> Cạo vôi	\$100.000
<input type="checkbox"/> Tẩy trắng	\$1.200.000
<input type="checkbox"/> Chụp hình răng	\$200.000
Trám răng <input type="text" value="0"/>	\$80.000/cái

Hình 1: Màn hình chính của ứng dụng

Hướng dẫn:

1. Tạo ứng dụng Windows Form có tên **DentalPaymentApp** theo các bước sau
 - a. Tạo project mới (Ctrl+Shift +N),
 - b. Trong cửa sổ new project chọn Visual C# - Windows
 - c. Phần template chọn Windows Forms Application
 - d. Đặt tên project trong phần Name

Xem hình 2 mô tả các bước tạo ứng dụng Windows Form



Hình 2: Tạo ứng dụng Windows Form

2. Sau khi hoàn tất các bước trên VS.NET sẽ phát sinh ra một project Windows Form mẫu, cho phép người lập trình bắt đầu xây dựng các ứng dụng. Giao diện của VS.NET 2005 cho ứng dụng vừa tạo có dạng như hình 3 bên dưới.

Màn hình VS.NET cho ứng dụng Windows Form bao gồm các phần cơ bản

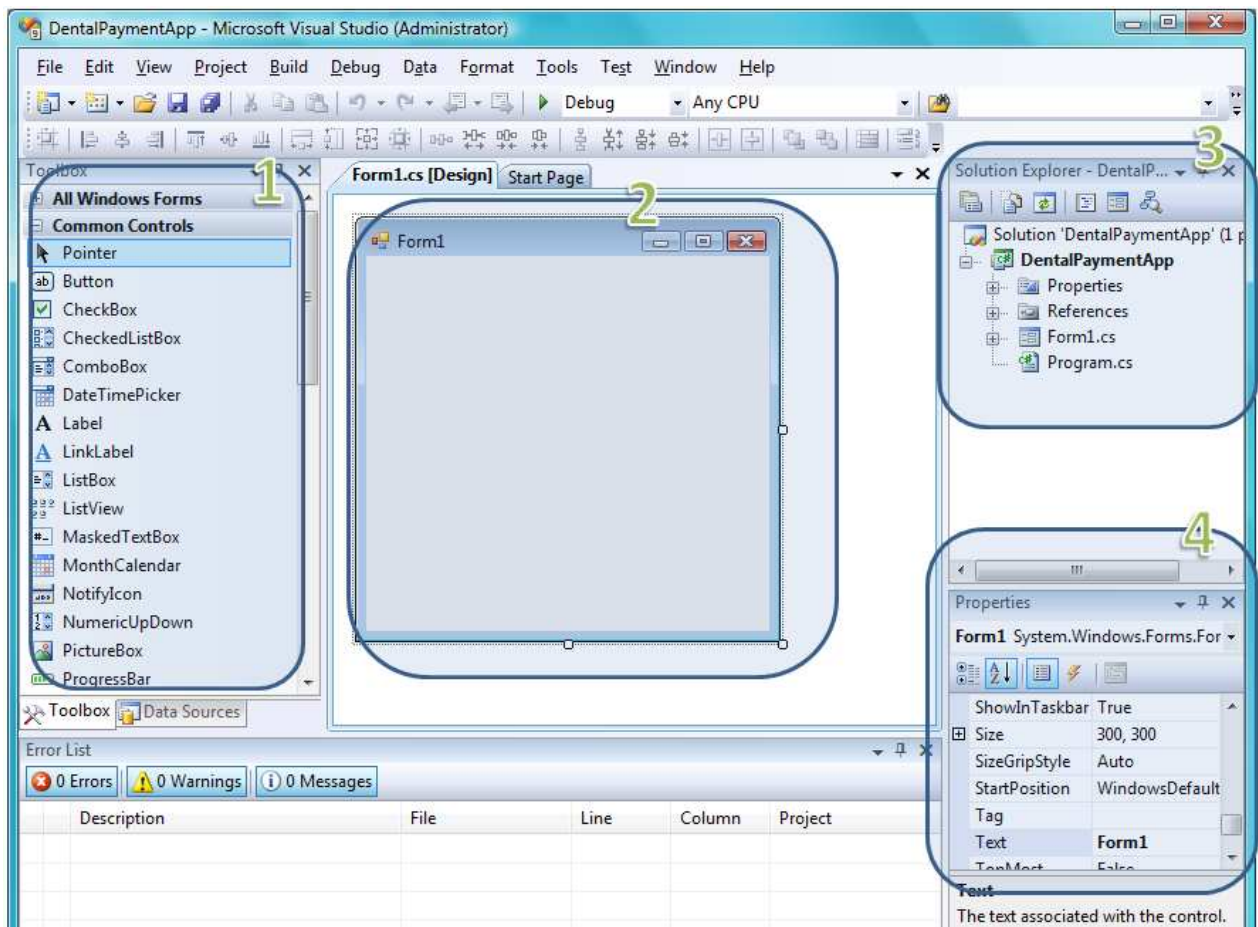
(1): Toolbox: chứa các control cho phép kéo thả vào form

(2): Màn hình thiết kế form, có thể chuyển sang phần code editor...

(3): Cửa sổ Solution Explorer: cho phép người lập trình có thể quản lý các thành phần trong project, hỗ trợ định vị nhanh chóng đến các file mã nguồn.

(4): Cửa sổ property: cho phép user có thể custom lại các thành phần control trên form như: thiết lập các thuộc tính cho control, form, component, cho phép khai báo trình xử lý sự kiện của các control trên form...

(Trước khi viết chương trình, sinh viên nên làm quen với các thành phần trên giao diện này)



Hình 3: Màn hình VS. NET phục vụ cho việc tạo project Windows Form

3. Thiết kế form theo mô tả như sau

Bảng 1: Mô tả các control trên form

STT	Name	Control	Thiết lập các thuộc tính cho control
1	lblTitle	Label	Text = “Dental Payment Form”, Font = “Impact, Size = 17”, ForeColor = Blue
2	lblName	Label	Text = “Tên khách hàng”
3	txtName	TextBox	
4	chkClean	CheckBox	Text = “Cạo vôi”
5	lblCleanCost	Label	Text=”\$100000”
6	chkWhitening	CheckBox	Text=”Tẩy trắng”
7	lblWhiteningCost	Label	Text=”\$1200000”
8	chkXRray	CheckBox	Text=”Chụp hình răng”
9	lblXRrayCost	Label	Text=”\$200000”

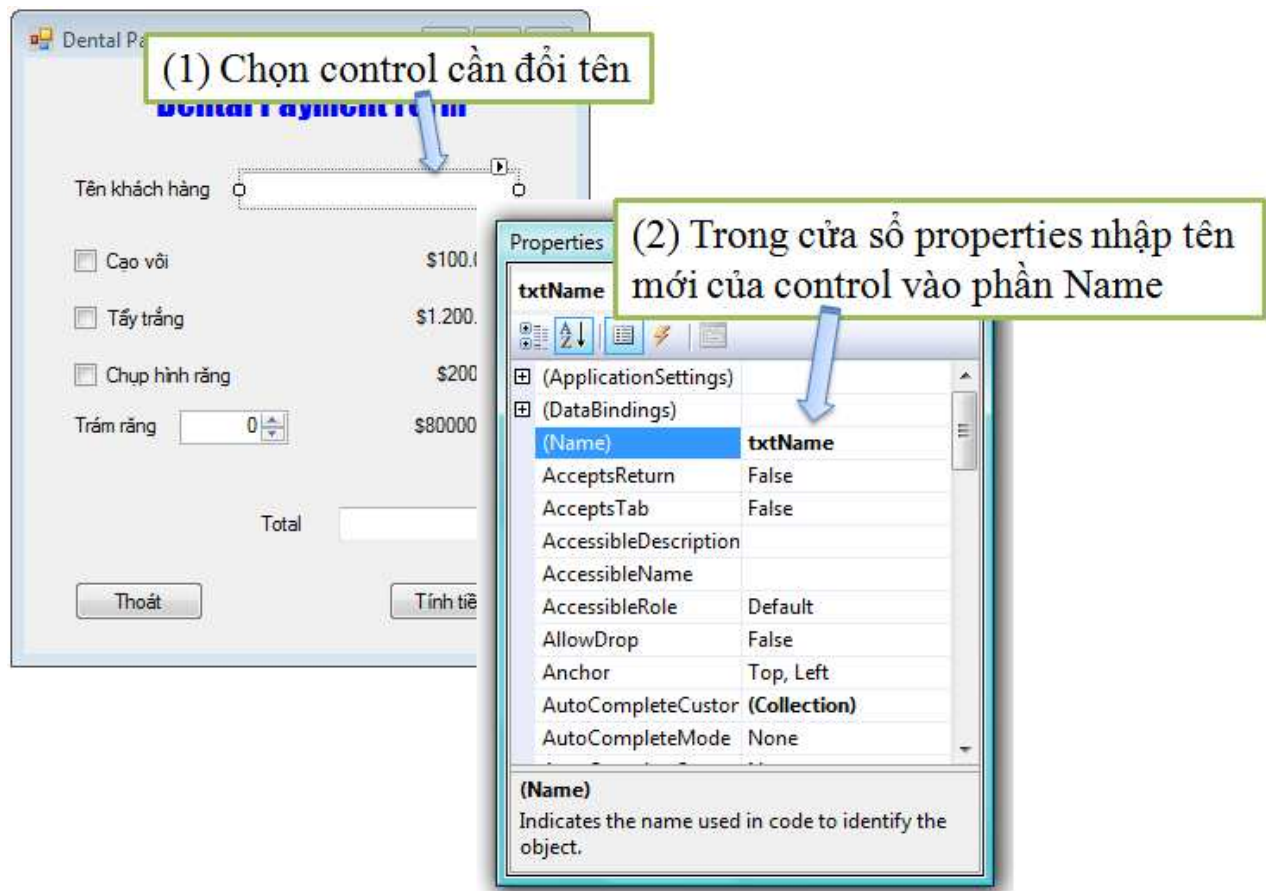
10	lblFilling	Label	Text=""Trám răng"
11	numFilling	NumericUpDown	
12	lblFillCost	Label	Text=""\$80000"
13	lblTotal	Label	Text=""Total"
14	txtTotal	TextBox	Enable = False
15	btnExit	Button	Text=""Thoát"
16	btnCalc	Button	Text=""Tính tiền"



Hình 4: Giao diện của Form chương trình

Cách thực hiện:

- Chọn trong ToolBox control tương ứng rồi kéo thả vào vị trí xác định trên form.
- Sử dụng cửa sổ properties thiết lập các thông tin:
 - o Đổi tên của control theo mô tả ở bảng trên
 - o Thiết lập các giá trị cho control theo mô tả ở bảng 1

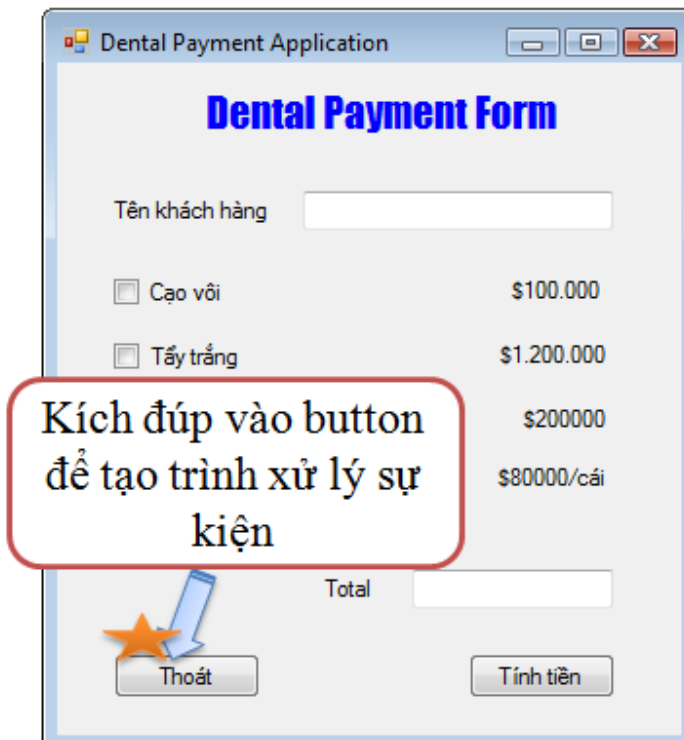


Hình 5: Minh họa việc đổi tên của TextBox trên Form

4. Tạo trình xử lý sự kiện click cho button “Thoát”: chức năng này khi thực hiện sẽ kết thúc ứng dụng (đóng form lại)

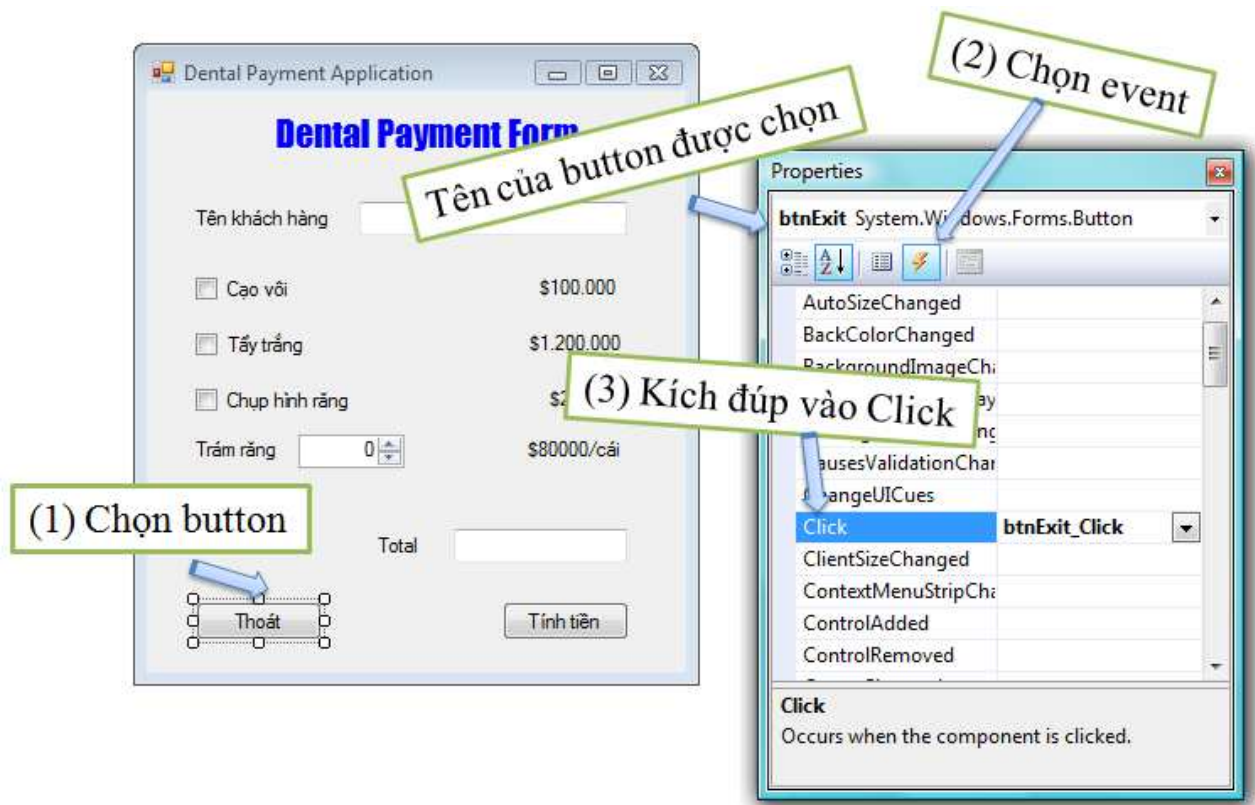
Cách tạo trình xử lý sự kiện: có thể làm theo một trong hai cách

- **Cách 1**: Kích đúp vào button cần tạo trình xử lý sự kiện trong màn hình Form design view: khi đó VS sẽ tạo trình xử lý sự kiện gắn với sự kiện Click của button “Thoát”



Hình 6: Minh họa cách kích đúp vào button để tạo event handler

- **Cách 2:** chọn button cần tạo trình xử lý, sau đó kích tab event trong cửa sổ Properties, kích đúp vào mục Click trong cửa sổ event.



Hình 7: Minh họa các bước khai báo event handler từ cửa sổ properties của button
 Nội dung của trình xử lý sự kiện Click của button btnExit như sau

```
private void btnExit_Click(object sender, EventArgs e)
{
    // đóng form lại ==> kết thúc ứng dụng
    this.Close();
}
```

5. Tạo chức năng tính tiền, chức năng này được kích hoạt khi button “Tính tiền” được chọn.
 Mô tả chức năng **GetPay()** như sau (GetPay() là phương thức thành viên của lớp Form chính:

- + Kiểm tra xem tên khách hàng có được nhập hay không?
- + Nếu chưa: → xuất thông báo, yêu cầu nhập tên khách.
- + Đã nhập: thực hiện các bước sau

```
Total = 0
If (cạo vôi) Total += 100.000
If ( tẩy trắng) Total += 1.200.000
If ( chụp hình răng) Total +=200000
Total += (số răng trám)*80000
```


Xuất số tiền ra TextBox txtTotal

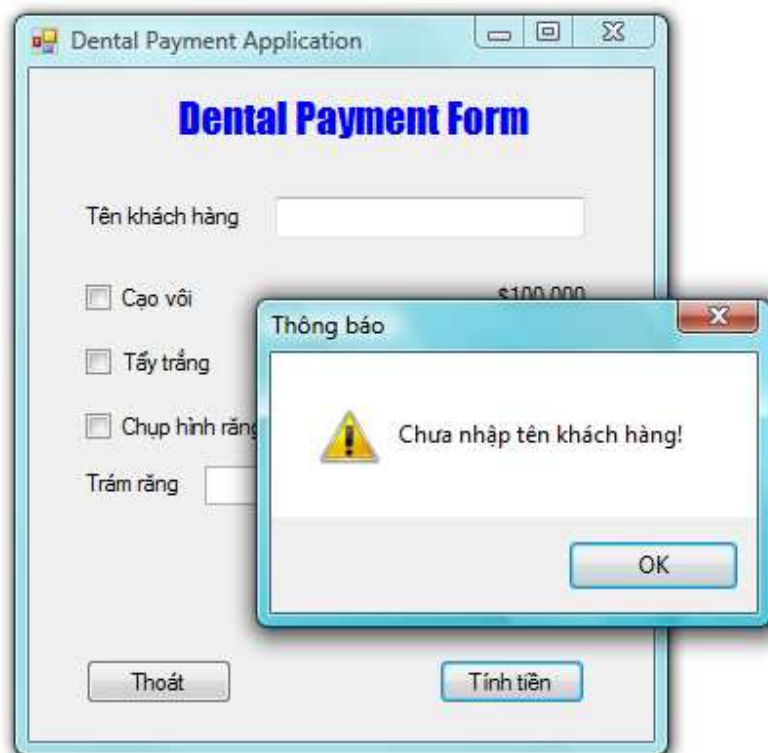
Sinh viên tự viết phương thức GetPay theo mô tả bên trên!

Tạo trình xử lý sự kiện cho button btnCalc rồi trong trình xử lý sự kiện này gọi chức năng GetPay.

```
private void btnCalc_Click(object sender, EventArgs e)
{
    // gọi chức năng tính tiền
    GetPay();
}
```

Kết quả chương trình

1. Trường hợp không nhập tên khách hàng: phát sinh message box cảnh báo



Hình 8: Không tính tiền khi chưa nhập tên khách hàng

2. Màn hình sau khi tính tiền cho khách

Item	Price
<input checked="" type="checkbox"/> Cạo vôi	\$100.000
<input type="checkbox"/> Tẩy trắng	\$1.200.000
<input checked="" type="checkbox"/> Chụp hình răng	\$200.000
Trám răng (2)	\$80.000/cái
Total	\$460000

Hình 9: Màn hình tính tiền cho khách

Nâng cao:

- Tạo một Data Grid lưu các thông tin cho mỗi lần tính tiền, mỗi bill tính tiền của khách sẽ được lưu trên 1 dòng của Data Grid.
- Bổ sung chức năng lưu trữ các bill ra file dạng text.

