

**Alessandro Bazzolo – 498905 STI**

**SOFTWARE PER CALCOLO PREMI DI UNA  
COMPAGNIA ASSICURATIVA**

## **INDICE**

RELAZIONE ESPERIENZA DI STAGE.....	pag. 3
RICHIESTA DI IMPLEMENTAZIONE.....	pag. 5
OBIETTIVO DELLA TESI.....	pag. 6
PROGETTO DELL'APPLICAZIONE.....	pag. 6
SVILUPPO DELL'APPLICAZIONE.....	pag. 8
CONFRONTO INI – XML.....	pag. 19
CONCLUSIONI.....	pag. 19
RIFERIMENTI BIBLIOGRAFICI.....	pag. 20
CODICE.....	pag. 20

## **RELAZIONE ESPERIENZA DI STAGE alla ComdataTech [ex MetSogeda]**

### **ComdataTech**

La ComdataTech (ex MetSogeda) è un'azienda di servizi informatici e fa parte del gruppo Comdata da gennaio 2007. Comdata è leader in Italia nell'organizzazione di servizi di supporto al cliente ed opera sul mercato in qualità di Partner Industriale stabilendo rapporti di collaborazioni strategiche con il cliente. In particolare, ComdataTech a Padova, si occupa principalmente di gestire i seguenti servizi ai clienti:

Call Center per alcune aziende ospedaliere e per alcune compagnie assicurative

Help Desk su applicativi utilizzati dalle compagnie assicurative

Data Entry per alcuni istituti bancari e compagnie assicurative

L'area in cui sono impiegato è il Servizio Statistico, e mi occupo di fornire rapporti ai clienti sull'efficienza del servizio fornito. I clienti (a seconda delle loro esigenze) richiedono report giornalieri, settimanali, mensili, trimestrali, semestrali o annuali che provvedo a fornire utilizzando gli strumenti che descriverò in seguito. Recentemente sono stato accorpato all'Area Sviluppo, che si occupa di produrre applicativi per i clienti e per utilizzo interno.

Tralasciando l'area di Data Entry, per la quale non esistono attualmente servizi di reportistica, cercherò di spiegare brevemente come funziona la gestione dei servizi offerti dal Call Center e dall'Help Desk. Prendiamo come esempio il caso di una compagnia assicurativa che abbia affidato alla nostra azienda la gestione dell'assistenza su un certo software. Quando l'impiegato dell'agenzia riscontra un problema con il software in questione, chiama il numero verde dell'assistenza. Risponderà un operatore del Call Center che aprirà una segnalazione al gruppo di operatori di Help Desk che si occupano della risoluzione di quello specifico tipo di problema. L'HD richiamerà il cliente e collegandosi al suo computer da remoto cercherà di risolvere il problema.

Il sistema utilizzato per gestire le segnalazioni delle agenzie è il Trouble Ticketing. Questo sistema consiste nell'utilizzare un applicativo in cui possano essere inseriti i dati importanti di ogni singola chiamata da parte dell'operatore del Call Center, in modo da farlo visualizzare poi dall'operatore di Help Desk. La procedura è resa possibile mediante l'utilizzo di un applicativo della BMC: Remedy 6.3, che permette di inserire e modificare dati all'interno della base dati su cui si appoggia.

## **Crystal Reports 9.0**

**Crystal Reports** è uno strumento di programmazione utilizzato per produrre report, ovvero stampe di dati. È stato creato dalla Crystal Decisions che poi, nel dicembre 2003, è stata acquisita dalla Business Objects.

I punti di forza del prodotto sono dati dalla sua semplicità di utilizzo, dalla velocità di produzione di template di report, facilmente gestibili e modificabili, e dall'integrabilità con altri prodotti/programmi soprattutto su piattaforme Microsoft; il suo limite maggiore è dato dalle performance non ottimali in presenza di volumi di dati elevati.

Nella realtà aziendale in cui mi trovo, Crystal Reports viene usato dal reparto Servizio Statistico per produrre riepiloghi di dati da inviare ai nostri clienti in modo che possano valutare il livello del nostro servizio. Capita spesso che i clienti, o risorse interne all'azienda, abbiano bisogno di informazioni o di estrazioni di dati dai vari database, con Crystal posso estrarre i dati in maniera semplice e veloce rendendoli anche più intuitivi da comprendere di una brutale query con MySQL.

## **24x7 Scheduler 3.4.24**

**24x7 Scheduler** è un programma della SoftTree Technologies che utilizzo per pianificare il lancio di report senza doverlo fare manualmente. È installato su un Server sempre acceso e genera report a qualunque ora del giorno e della notte.

## **Delphi 7.0**

**Delphi** è sia un linguaggio di programmazione sia un ambiente di sviluppo. È stato creato dalla Borland. Alla fine del 2006 Borland ha creato una società controllata, CodeGear, alla quale sono stati trasferiti tutti gli ambienti di sviluppo, incluso Delphi.

Il linguaggio Delphi, precedentemente conosciuto come Object Pascal (una versione di Pascal orientata agli oggetti), è stato sviluppato inizialmente per Microsoft Windows, ma negli ultimi anni sono state pubblicate versioni per GNU/Linux (Kylix) e per il framework .NET di Microsoft. È molto utilizzato per lo sviluppo di applicazioni desktop e aziendali che utilizzano database, ma essendo uno strumento di sviluppo di carattere generico lo si può utilizzare per molti tipi di progetti.

## RICHIESTA DI IMPLEMENTAZIONE

Come progetto di sviluppo, viene richiesta la creazione di un applicativo che gestisca il calcolo dei premi di una nota compagnia assicurativa. Precedentemente, questo calcolo veniva eseguito manualmente dagli operatori, moltiplicando un premio base per dei coefficienti contenuti nelle celle di un foglio elettronico. I coefficienti venivano ricopiati a mano in una determinata cella e, grazie a formule pre-impostate, consentivano il calcolo del premio richiesto (si veda figura).

Premio base 202,51 CLASSE E con massimale 775.000,00 euro			
premio base 202,51			
classi di merito	coeff.	classi di merito	coeff.
1C	0,760	4	0,940
1B	0,790	5	0,970
1A	0,820	6	1,000
1	0,850	7	1,200
2	0,880	8	1,500
3	0,910	9	2,000
Massimale RC1			
			coeff.
		1300.000,00	1,04
		1.600.000,00	1,07
		2.100.000,00	1,09
		2.600.000,00	1,09
		3.700.000,00	1,12
		5.200.000,00	1,14
Zona territoriale			
	coeff.	Zona territoriale	coeff.
1	1,72	10	0,81
2	1,57	11	0,75
3	1,43	12	0,70
4	1,30	13	0,65
5	1,20	14	0,60
6	1,10	15	0,55
7	1,00	16	0,50
8	0,93	17	0,45
9	0,87		
uso professionale			
	coeff.		
si	1,50		
no	1,00		
condizione RIVALSA			
rivalsa per guidatori inferiore a 26 anni:			
	coeff.		
rivalsa fino a 1.500,00 coeff. di premio	0,900		
nessuna rivalsa	1,150		
intestatario al PRFA inferiore a 26 anni			
	coeff.		
	1,000		
sesso			
	coeff.		
M	1,00		
F	0,99		
Atlands	1,00		
trasportati			
	coeff.		
si	1,35		
no	1,00		

calcolo manuale premio rca		
classe di merito	1,000	202,51
massimale rc	2,000	405,02
età assicurato	1,000	405,02
zona territoriale	2,000	810,04
uso professionale	1,000	810,04
condizione rivalsa	2,000	1620,08
sesso	1,000	1620,08
trasportati	2,000	3240,16
interessi di frazionamento	0%	3240,16
<b>PREMIO IMPONIBILE</b>		<b>3240,16</b>
<b>PREMIO LORDO</b>		<b>3985,40</b>

Tipo frazionamento	% di aumento
Annuale	0%
Trimestrale	5%
Quadrimestrale	4%
Semestrale	3%

- esempio del foglio elettronico -

Ovviamente questa procedura, oltre ad essere dispendiosa in termini di tempo, dava la possibilità all'operatore di commettere errori di copiatura, e non risultava perciò sicura.

Inoltre esistono diverse classi di veicoli, ed era presente un foglio elettronico per ognuna di esse, credo sia inutile sottolineare come ci fosse bisogno di un software che potesse rendere semplice e sicuro il calcolo richiesto.

Il programma deve prevedere anche la possibilità che i coefficienti utili al calcolo dei premi possano variare nel tempo, così come il premio base. L'operatore deve essere in grado di poter aggiornare i dati in qualsiasi

momento senza dover ricorrere all'intervento dello sviluppatore dell'applicativo.

Questo progetto mi è stato assegnato per

- 1) Fornire ai nostri operatori uno strumento di facile utilizzo per il calcolo dei premi
- 2) Farmi avere la possibilità di gestire nella totalità un progetto di sviluppo Delphi e di conseguenza migliorare il mio status di programmatore

## **OBIETTIVO DELLA TESI**

Considerando questo progetto come un'occasione per apprendere il più possibile, si è pensato di svilupparlo utilizzando la tecnologia xml abbinata al linguaggio di programmazione Delphi.

L'obiettivo è confrontarsi per la prima volta col mondo dello sviluppo di software, arricchire il bagaglio di conoscenze e generare un applicativo funzionale che possa semplificare il lavoro dei miei colleghi

Essendo un progetto che richiede la conoscenza di diversi oggetti Delphi, ci si prefigge di utilizzarne il più possibile in modo da poter familiarizzare con essi in vista di un possibile utilizzo futuro.

Come secondo obiettivo ci si pone quello di impraticarsi con la tecnologia xml per la parte di aggiornamento dei dati, nonostante nel progetto iniziale, questa sezione dell'applicativo si sia gestita mediante un file .ini.

Infine si metteranno a confronto i due metodi evidenziando vantaggi e svantaggi di ciascuno rispetto all'altro.

## **PROGETTO DELL' APPLICAZIONE**

Si intende strutturare questo software utilizzando una maschera iniziale che possa permettere all'utilizzatore di scegliere quale tipo di veicolo deve assicurare.

Le classi possibili sono le seguenti:

RC Settore I e II

Incendi

RC Settore III Autobus

Settore IV Autocarri Bonus/Malus

Settore IV Autocarri Franchigia

Settore IV Motocarri Tar Fissa

Settore V Ciclomotori Franchigia Fissa

Settore V Ciclomotori No Claim Pejus

Settore V Motocicli Franchigia Fissa  
Settore V Motocicli No Claim Pejus  
Settore VI Macchine Operatrici  
Settore VII Macchine Agricole

Ognuna con un diverso premio base, e con diversi parametri di incremento/decremento.

Si darà la possibilità all'operatore di scegliere quale settore vuole gestire nella maschera iniziale, e mediante l'apertura di un pannello, si faranno apparire tutte le possibili varianti di quella specifica classe, così come mostrato in figura.



– maschera iniziale –

Ogni pannello apparirà nella sezione di destra, consentendo a chi utilizza il programma di impostare i parametri variabili per il calcolo del premio desiderato.

Non ci saranno campi modificabili, proprio per evitare qualsiasi tipo di errore da parte di un utilizzatore poco esperto.

Nella figura successiva possiamo vedere un esempio di uno dei pannelli descritti in precedenza.

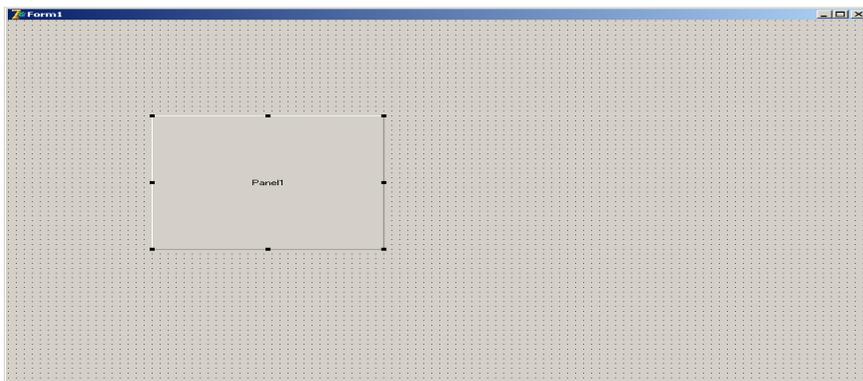
- esempio di pannello RC Settore I e II -

Ogni casella combinata avrà una serie di possibilità di scelta, ad ognuna di esse sarà inoltre abbinato un valore. Questi valori che influenzeranno il calcolo del premio, saranno caricati da un file .ini che li conterrà tutti.

## SVILUPPO DELL'APPLICAZIONE

Andiamo ora ad analizzare l'implementazione dell'applicazione, descrivendo punto per punto gli strumenti usati e le modalità di utilizzo. Innanzitutto la prima necessità è avere un pannello per ogni settore dell'applicativo.

In Delphi si utilizza un oggetto di tipo TPanel, il quale genera un pannello di dimensione impostabile.



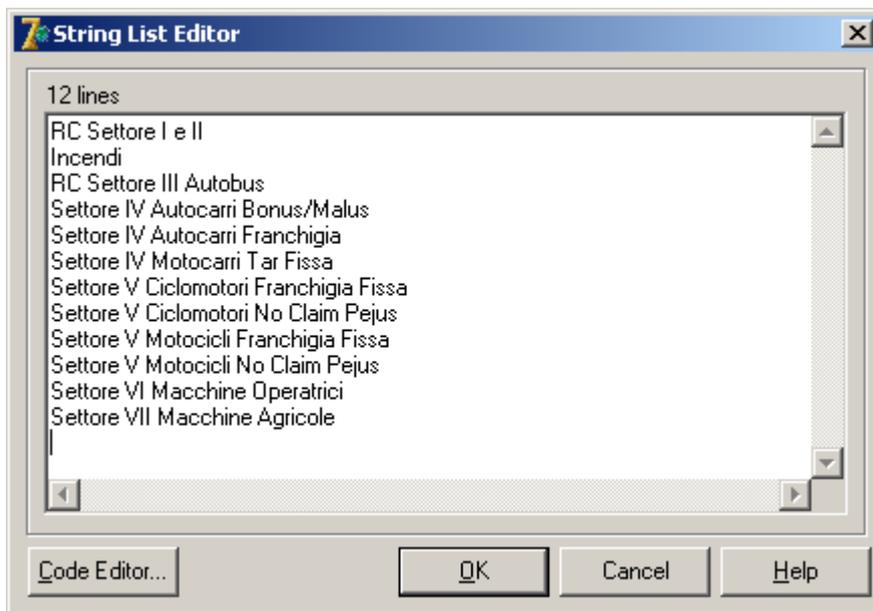
Si è creato un TPanel alla volta e mediante la proprietà Visible, è stato impostato che al lancio dell'applicativo, nessuno dei pannelli risulti visibile.

Panel1.Visible:=False;

Sfruttando questa proprietà si può far apparire il pannello corrispondente al verificarsi di uno specifico evento

In questo caso si vuole che il pannello appaia nel momento in cui si necessita di gestire il calcolo premi dello specifico settore.

Per permettere all'utente di scegliere quale settore gestire, si è utilizzato un oggetto TComboBox. Esso non è altro che una casella combinata, ossia un campo testuale che contiene una lista di stringhe. I settori citati prima, sono stati impostati nella proprietà ItemString dell'oggetto (si veda figura)



Si è quindi creato un oggetto TButton e un evento associato al click del bottone stesso. Quando l'utente clicca sul bottone, viene controllato il valore del TComboBox, e se è impostato uno dei settori sopraindicati, viene reso visibile il pannello corrispondente, facendo rimanere invisibili gli altri.

Veniamo alla struttura di ogni pannello. Gli oggetti utilizzati per calcolare i premi di ogni singolo settore sono: TLabel, TEdit, TComboBox, TRadioGroup e TButton.

Il calcolo del premio, come già detto in precedenza, è il risultato della moltiplicazione tra il premio base e alcuni coefficienti che variano in base all'opzione scelta. Prendiamo ad esempio l'età del conducente. Ad ogni

fascia di età è associato un coefficiente, come possiamo vedere dalla figura del foglio elettronico utilizzata in precedenza.  
 Si è deciso di creare una struttura simile anche nel mio applicativo, associando ognuno di questi coefficienti alla proprietà Text di un oggetto TEdit non visibile.

The screenshot shows the 'Calcolo Premi V 1.08' application interface. The top part features a form with several dropdown menus and input fields for configuring insurance parameters. The parameters are:

- MASSIMALE: 775.000
- ZONA TERRITORIALE: 1
- FRAZIONAMENTO: Annuale
- BONUS "2 RUOTE": Si
- ETA': < 22
- CLASSE: 1C
- SESSO: M
- CILINDRATA: 51-399
- TIPO VEICOLO: Motocicletta

Below the form is a large table of coefficients. The table has columns for CLASSE, FRAZIONAMENTO, ZONA TERRITORIALE, MASSIMALE, and CILINDRATA. The coefficients are mostly 1.00, with some variations for higher values. The table is organized into sections for different parameters like SESSO, BONUS "DUE RUOTE", and CONDIZIONE RIVALSA.

CLASSE	FRAZIONAMENTO	ZONA TERRITORIALE	MASSIMALE	CILINDRATA
1C	1.00	Annuale	1.00	10
1B	1.00	Trimestrale	1.00	11
1A	1.00	Quadrimestrale	1.00	12
1	1.00	Semestrale	1.00	13
2	1.00	SESSO	1.00	14
3	1.00	M	1.00	15
4	1.00	F	1.00	16
5	1.00	Azienda	1.00	17
6	1.00	BONUS "DUE RUOTE"	1.00	18
7	1.00	Si	1.00	19
8	1.00	No	1.00	20
9	1.00	TRASPORATI	1.00	21
6	1.00	30-49	1.00	7
7	1.00	> 50	1.00	8
8	1.00	M	1.00	16
9	1.00	F	1.00	17
	1.00	Azienda	1.00	9

Additional parameters and coefficients shown in the interface:

- ETA': < 22 (1.00), 23-29 (1.00), 30-49 (1.00), > 50 (1.00)
- TIPO VEICOLO: Motocicletta (1.00), Scooter (1.00)
- CONDIZIONE RIVALSA: si rivalsa (1.00), no rivalsa (1.00), < 26 anni (1.00)
- PREMIO BASE: 100 (1.00), 1.23 (1.00)

- i coefficienti non appariranno nella versione data all'utente -

I TEdit non sono altro che campi di testo modificabile, ne viene creato uno in corrispondenza del premio base di quel settore e di ogni coefficiente. Il valore che sarà loro assegnato potrebbe essere fornito manualmente dallo sviluppatore, ma in questo modo in caso di variazione del valore del premio o del coefficiente, sarebbe compito dello sviluppatore stesso andare a modificare la cifra, ed ovviamente non vogliamo dover ricorrere ad un metodo simile.

La soluzione più semplice è creare un file ini, che contenga tutti i valori corrispondenti al premio e ai coefficienti, e caricarli sui TEdit corrispondenti all'avvio dell'applicativo.

Innanzitutto si deve permettere al programma di interfacciarsi con il file. Associa al click di un bottone una funzione che associ ad una variabile il nome del file più il percorso del file stesso (che nel nostro caso è nella stessa cartella dove si trova l'eseguibile)

```
myFile:=TIniFile.Create(sPath+'Test.ini');
```

I File ini sono strutturati in questo modo:

```
[RC Settore I e II]
Classe1H=0,334
Classe1G=0,346
Classe1F=0,348
Classe1E=0,351
Classe1D=0,370
Classe1C=0,390
```

...

e nel codice di Delphi vengono richiamati con i comandi seguenti

```
Edit153.text:=myFile.Readstring('RC Settore I e II','Classe1H','EMPTY');
Edit154.text:=myFile.Readstring('RC Settore I e II','Classe1G','EMPTY');
Edit155.text:=myFile.Readstring('RC Settore I e II','Classe1F','EMPTY');
Edit156.text:=myFile.Readstring('RC Settore I e II','Classe1E','EMPTY');
Edit157.text:=myFile.Readstring('RC Settore I e II','Classe1D','EMPTY');
Edit158.text:=myFile.Readstring('RC Settore I e II','Classe1C','EMPTY');
```

...

La proprietà readstring delle variabili di tipo TiniFile va a cercare innanzitutto il valore dell'etichetta RC Settore I e II, poi scende da quel punto del file finché non trova la stringa Classe1H, e associa il valore che trova dopo il segno '='. Nel caso non sia in grado di trovarlo associa al TEdit il valore EMPTY.

Siamo giunti ad avere i valori corretti dei coefficienti e del premio base. Ora si deve rendere possibile all'utente la selezione per esempio della classe di merito corrispondente, e far fare al programma il calcolo.

L'utente, dopo aver effettuato la scelta del settore, si troverà davanti ad una maschera in cui dovrà impostare i valori desiderati nei TComboBox, e nei TRadioGroup e poi cliccare sul TButton, che in base a quanto impostato, calcolerà il premio.

Abbiamo già visto come funziona un TComboBox, gli oggetti di tipo TRadioGroup sono abbastanza simili, mostrano infatti una lista di stringhe selezionabili una alla volta. Contrassegnando una voce, infatti, il contrassegno sparirà dalla voce precedentemente selezionata. Mediante la proprietà ItemIndex è possibile selezionarne una di default.



Al click del bottone viene associata una procedura che va a verificare i valori dei TComboBox e dei TRadioGroup, e quindi moltiplicare il premio base per i coefficienti precedentemente caricati dal file ini.

L'istruzione if, comune a molti linguaggi di programmazione, permette di eseguire determinati comandi solo al verificarsi di certe condizioni. Nel caso seguente eseguirà le istruzioni tra il "begin" e l'"end" solo se il valore della TComboBox 13 sarà '1H'.

```
if(ComboBox13.Text = '1H')
then begin
...
end;
```

Per calcolare il premio dobbiamo innanzitutto assegnare ad una variabile il valore del premio base, che chiameremo per esempio TotaleAuto. Il valore del TEdit viene, allo stesso modo dei coefficienti, caricato dal file ini. Si pone però un problema: il valore caricato è una stringa di testo.

Questo significa che con essa non sarà possibile effettuare operazioni di calcolo.

Per risolvere il problema dichiaro comunque una variabile di tipo numerico (float), e ci assegno il valore del TEdit convertito a numero. L'istruzione che mi permette di fare ciò è StrtoFloat

```
TotaleAuto:= StrToFloat(Edit269.Text);
```

Abbiamo quindi il nostro valore del premio base, ora sarà sufficiente creare una if per ogni possibile valore dei TComboBox e dei TRadioGroup in cui moltiplicare TotaleAuto per il coefficiente, e salvare il nuovo valore sempre nella stessa variabile come da sintassi seguente.

```
if(ComboBox13.Text = '1H')
then begin
TotaleAuto:=TotaleAuto*StrToFloat(Edit153.Text);
Label163.Caption:=FloatToStr(TotaleAuto);
end;
```

Il TLabel non è altro che un testo stampato a video. Viene messo in ogni condizione, in maniera tale che quando il programma avrà passato tutte le if, quel numero rappresenterà il valore del premio finale. Ovviamente il TLabel richiede una stringa, sicchè si dovrà convertire di nuovo il nostro valore numerico con il comando FloatToStr.

Il TRadioGroup non ha la proprietà text, sicchè nella condizione andiamo a controllare la posizione nella lista

```
If(RadioGroup3.ItemIndex=0) then ...
```

Nelle varie casistiche del programma figurano dei coefficienti che dipendono da più di un parametro. Per esempio la fascia di età e il sesso. Un maschio di 22 anni ha il premio moltiplicato per un coefficiente diverso rispetto ad una femmina della stessa età.

Per far sì che il nostro programma tenga conto di entrambi i parametri, è sufficiente aggiungere una condizione alla if.

Nel caso dell'esempio potremmo avere:

```
if((ComboBox18.Text = '22-23') and (RadioGroup3.ItemIndex=0))
then begin
...
end;
```

Questa versione dell'applicativo (che chiameremo versione 1) è stata consegnata ai colleghi che l'hanno utilizzata e trovata di facile comprensione ed estremamente funzionale.

Nonostante ciò, sfruttando il fatto che da luglio 2008, la compagnia assicurativa ha modificato il modo di calcolare i premi introducendo casistiche nuove, si è pensato di produrre una seconda versione dell'applicativo, prendendo ovviamente come base la versione 1, che potesse:

- 1) Rendere ancora più semplice l'utilizzo dove possibile mediante la creazione di un menu

- 2) Rendere il codice più facilmente modificabile dal programmatore ricorrendo a funzioni
- 3) Evitare di utilizzare gli oggetti TEdit, giacché ci si è resi conto che con appropriate variabili, il codice risulta più lineare
- 4) Accorpate più settori possibili in modo da risparmiare tempo nel momento in cui si dovrà modificarli
- 5) Evitare quindi di usare un numero troppo elevato di pannelli, che generano notevole confusione se sovrapposti ed impostare nel codice le coordinate della loro posizione
- 6) Rendere l'applicativo utilizzabile su ogni tipo di piattaforma sostituendo il file .ini con un file .xml

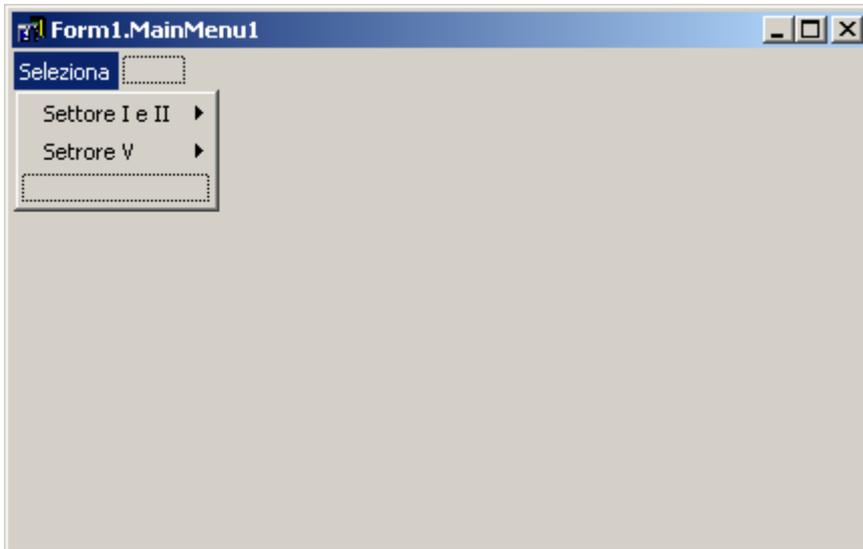
Si è ritenuto necessario inserire un menu per facilitare l'utente nella scelta del settore. Le nuove tariffe, infatti, prevedono un minimo di tre fogli excel ciascuno a seconda che si tratti di affari in corso o affari nuovi. L'introduzione di un menu ha permesso di rendere più intuitiva la selezione, come mostrato in figura



In Delphi 7 esiste un componente chiamato TMainMenu che permette l'implementazione di quanto descritto. Attraverso la sua proprietà Item,

è possibile impostare le varie voci che desideriamo inserire nel menu stesso.

Allo stesso modo se si desidera creare sub menu è sufficiente, dopo aver inserito una voce, di cliccare sopra di essa con il tasto destro e selezionare la voce "Create Sub Menu".



Ogni voce inserita all'interno di un menu, porterà alla creazione automatica di un oggetto TMenuItem, in modo che il click su una singola voce possa rappresentare un evento indipendente. Nel nostro caso, l'apertura del pannello corrispondente alla voce selezionata.

Si è pensato di inserire una variabile di tipo string chiamata Settore, che assumesse un valore diverso a seconda di quale voce del menu venisse selezionata.

Esempio

```
Settore:='Settore I e II - Franchigia - Affari Nuovo';
```

in questo modo si rende possibile effettuare i calcoli desiderati per il settore selezionato usando come condizione della if il valore di Settore anziché il valore del TComboBox come si faceva nella prima versione dell'applicativo.

Nella nuova versione dell'applicativo, inoltre, si è pensato di evitare ridondanze di codice utilizzando funzioni che possano venire richiamate ogni qualvolta si necessita il codice in esse contenuto.

La prima funzione creata è RESET.

A differenza della prima versione dell'applicativo, in cui ogni pannello aveva al suo interno due TLabel in cui veniva visualizzato il valore del

premio, la v2.0 presenta due soli oggetti all'esterno dell'area di visualizzazione dei pannelli. La funzione reset imposta a 0,00 i valori di queste etichette e rende invisibili tutti i pannelli. Inoltre richiama altre funzioni di reset specifiche di ogni settore, il cui scopo verrà tra poco descritto. La sintassi utilizzata è la seguente:

```
procedure TForm1.BRESET;  
begin  
  LTOTALE.Caption:='0,00';  
  LLORDO.Caption:='0,00';  
  PAUTO.Visible:=False;  
  PMOTO.Visible:=False;  
  ...  
  RESETPAUTO;  
  RESETPMOTO;  
  ...  
end;
```

dove PAUTO e PMOTO sono i pannelli relativi ai settori, mentre RESETPAUTO sono i loro reset.

Per chiamare una funzione in Delphi è sufficiente dichiararla tra le procedure utilizzate ed evocarla con il nome selezionato come fosse un'istruzione.

Da un'analisi dei fogli elettronici contenenti i dati e i coefficienti di ogni settore, è emerso che per esempio nel settore auto, i tre fogli hanno campi in comune a tutti, campi in comune tra due fogli e campi specifici di uno solo. Si è quindi pensato di creare dei sottopannelli che raggruppino questi campi.

Per semplificare la comprensione viene di seguito fornito un esempio:

Nel settore Auto, la classe di merito, i cavalli fiscali e la zona territoriale sono comuni a tutti e tre i fogli elettronici, motivo per cui i Combo Box che gestiscono il settaggio dei loro valori possono essere posizionati direttamente sul pannello principale del settore.

La franchigia, invece, è un'esclusiva di uno dei fogli, quindi si è creato un pannello secondario all'interno del principale che contiene tutti i Combo Box che sono contenuti solo in quel foglio. In questo modo si otterranno al massimo sei pannelli oltre al principale, in dettaglio: solo foglio 1, solo foglio 2, solo foglio 3, fogli 1e2, fogli 1e3, fogli 2e3.

Essi, inizialmente invisibili, verranno visualizzati al click del menu corrispondente. La funzione RESETPAUTO ha lo scopo di riportarli allo stato invisibile.

In questo modo ogni volta che viene invocata la funzione RESET, tutti i pannelli e sottopannelli scompaiono, e i valori del premio vengono settati a 0,00.

Per quanto riguarda i pannelli, dato che si è appurata la scomodità in fase di modifica di spostarli per scoprire quelli piazzati sotto, nella nuova versione i pannelli non saranno sovrapposti, ma tramite le proprietà Top e Left, saranno piazzati nella posizione desiderata all'apertura della Form.

Inoltre si è provveduto a creare una funzione che esegua i calcoli relativi al premio per ogni foglio. In questo modo reperire le informazioni risulta estremamente più facile, essendo esse suddivise e non più in un unico immenso susseguirsi di iterazioni if.

A questo punto, cliccando sul bottone di calcolo, ci sarà un controllo sul valore di Settore (impostato alla scelta da menu) dopodiché verranno eseguite le operazioni di calcolo come nella prima versione del programma.

Una differenza sostanziale, però, è data dal fatto che i coefficienti non vengano più caricati su oggetti di tipo TEdit. Ci si è infatti resi conto che ciò era poco utile. Nella v2.0, si è pensato ad utilizzare una variabile temporanea di tipo string chiamata Coefficiente.

Ad essa sarà assegnato il valore del coefficiente giusto il tempo di moltiplicarlo per il premio base.

Il vantaggio di procedere in questa maniera è dato dal fatto che non bisogna prestare attenzione al nome di ogni edit, non dovendo "salvare" su un oggetto di questo tipo il valore del coefficiente.

I dati, come precedentemente affermato, non saranno caricati da un file ini, ma da un file xml.

Il confronto tra le due tecnologie è rimandato all'apposita sezione, quello che si spiega ora, è come fare in Delphi a caricare dati da un xml.

Innanzitutto il formato del file xml è il seguente:

```
<?xml version="1.0"?>
```

```
<CALCOLOPREMI>
```

```

<GEN>

<Lordo>1,23</Lordo>

<Trimestrale>1,05</Trimestrale>
<Quadrimestrale>1,04</Quadrimestrale>
<Semestrale>1,03</Semestrale>

</GEN>

<SETTORE_I_e_II>

<PremioBase>466,37</PremioBase>

<Classe1H>0,334</Classe1H>
<Classe1G>0,346</Classe1G>
...
</SETTORE_I_e_II>
...
</CALCOLOPREMI>

```

Il Tag principale è CALCOLOPREMI, poi c'è un Tag GEN in cui sono contenuti gli attributi comuni a tutti i settori, e poi se n'è creato uno per settore.

Per agganciare il file xml all'applicativo, si deve ricorrere all'oggetto XmlDocument ed impostare il percorso del file nell'attributo FileName dell'oggetto stesso.

A questo punto il nostro applicativo ha agganciato il file, ora bisogna far sì che i valori dei vari coefficienti vengano correttamente caricati nella variabile.

Innanzitutto è necessario posizionarsi nel nodo giusto; nell'esempio precedente si hanno GEN e SETTORE\_I\_e\_II.

In questo caso al nodo viene assegnato un numero crescente a partire da zero in base all'ordine in cui appare nel file xml. GEN sarà quindi il numero 0, SETTORE\_I\_e\_II il numero 1 e così via per gli altri settori.

Mediante il comando

```
XMLNode := XMLDati.DocumentElement.ChildNodes[0];
```

(dove XMLDati è il nome dell'oggetto di tipo XmlDocument, e XMLNode una variabile di tipo XmlNode che serve specificatamente allo scopo di memorizzare il nodo) ci posizioniamo su GEN, e possiamo avere accesso a tutti i dati contenuti tra il tag di apertura e quello di chiusura.

I nostri coefficienti si trovano, come mostrato nell'esempio, tra due nodi figlio (i cosiddetti childnodes). Lo scopo prefissato è estrapolare il valore desiderato e caricarlo sulla variabile Coefficiente di cui si è parlato nelle pagine precedenti.

Come nel caso del file INI, è sufficiente una riga di codice per compiere l'operazione desiderata

```
Coefficiente:=XMLNode.ChildNodes['Trimestrale'].Text;
```

Grazie alla proprietà ChildNodes delle variabili di tipo XmlNode, è possibile in questo caso estrarre il valore del coefficiente relativo al frazionamento trimestrale e caricarlo sulla variabile.

A questo punto il gioco è fatto, ed è possibile procedere allo stesso modo della versione precedente dell'applicativo, con le conversioni numero-stringa, per ottenere la stampa a video del premio corretto.

## **CONFRONTO INI – XML**

Analizzando le due diverse procedure si può affermare che:

Il file INI è di più semplice utilizzo, non necessita di una sintassi complicata ed è più intuitivo rispetto ad un file xml, dove ci sono varie limitazioni, come l'impossibilità di chiamare un tag con un nome che inizi per numero, o l'obbligo di posizionarsi ogni volta sul nodo che contiene i childnodes d'interesse. Non bisogna trascurare, però, che lo standard xml è universalmente riconosciuto da tutti i sistemi operativi, al contrario dei file ini, che sono uno standard di Microsoft Windows. Questa non è una limitazione da poco, e si ritiene che possa essere sufficiente a far preferire la tecnologia xml.

## **CONCLUSIONI**

Il progetto descritto, è certamente un'ottima base per un'aspirante programmatore, che può maturare molta esperienza con l'ambiente Delphi, e con l'ambiente sviluppo in generale.

Non si sono presentate grosse difficoltà nella realizzazione, grazie alla notevole quantità di informazioni reperibili sul web e sui manuali.

L'applicativo Calcolo Premi è tuttora utilizzato dai nostri operatori di HelpDesk.

Viene giudicata positiva l'esperienza di stage alla MetSogeda (poi ComdataTech), in quanto ha certamente contribuito ad arricchire il mio bagaglio di conoscenze in campo informatico.

Sottolineo inoltre che, dopo i sei mesi di stage, l'azienda mi ha assunto come lavoratore a tempo indeterminato.

## **RIFERIMENTI BIBLIOGRAFICI**

Wikipedia enciclopedia libera ([www.wikipedia.it](http://www.wikipedia.it))

Mastering Delphi 6 – Marco Cantù – Sybex

Mastering Delphi 7 – Marco Cantù – Sybex

XML, conoscere il linguaggio xml significa poter comunicare veramente con tutti – Massimo Canducci – Apogeo

Delphi 7 Help

[www.delphiedintorni.it](http://www.delphiedintorni.it)

## **CODICE**

Di seguito il codice utilizzato per realizzare l'applicativo. Per evitare un troppo elevato numero di righe, sono state inserite solo le sezioni riguardanti il settore auto ed il settore motocicli essendo perfettamente uguale il meccanismo di implementazione degli altri settori.

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Windows, Messages, Math, SysUtils, Variants, Classes, Graphics, Controls,  
Forms, Dialogs, StdCtrls, ExtCtrls, xmldom, XMLIntf, msxmldom, XMLDoc, jpeg,  
Menus;
```

```
type
```

```
TForm1 = class(TForm)  
    LTOTALE: TLabel;  
    LLORDO: TLabel;  
    PAUTO: TPanel;  
    Label665: TLabel;  
    CBclasseDiMeritoI: TComboBox;  
    Label670: TLabel;  
    Label671: TLabel;  
    CBCavalliFiscaliI: TComboBox;  
    CBZonaTerritorialeI: TComboBox;  
    PAUTO_AN_AC: TPanel;  
    Label289: TLabel;  
    Label290: TLabel;  
    Label293: TLabel;  
    CBAalimentazioneI: TComboBox;  
    CBEtaProprietarioI: TComboBox;
```

CBRivalsaI: TComboBox;  
Label294: TLabel;  
Label295: TLabel;  
Label296: TLabel;  
Label297: TLabel;  
RGSessoI: TRadioGroup;  
CBProvenienzaI: TComboBox;  
Label1: TLabel;  
PAUTO\_AC: TPanel;  
Label291: TLabel;  
CBProfessioneI: TComboBox;  
XMLDati: TXMLDocument;  
Label2: TLabel;  
Label3: TLabel;  
Image1: TImage;  
Image2: TImage;  
Label303: TLabel;  
CBFrazionamentoI: TComboBox;  
BCALCOLA\_I\_II: TButton;  
RGPlurisinistrataI: TRadioGroup;  
PAUTO\_AN: TPanel;  
PAUTO\_AN\_FR: TPanel;  
PAUTO\_FR: TPanel;  
Label5: TLabel;  
CBEtaProfessioneII: TComboBox;  
Label292: TLabel;  
CBEtaVeicoloI: TComboBox;  
Label300: TLabel;  
RGGuidaEsclusivaI: TRadioGroup;  
Label6: TLabel;  
CBEtaVeicoloII: TComboBox;  
Label7: TLabel;  
RGRiservataI: TRadioGroup;  
Label286: TLabel;  
CBMassimaleI\_AC: TComboBox;  
Label4: TLabel;  
CBMassimaleI\_AN: TComboBox;  
Label8: TLabel;  
Label9: TLabel;  
CBRischioI: TComboBox;  
Label10: TLabel;  
CBAnniPatenteI: TComboBox;  
RGFranchigiaI: TRadioGroup;  
Label11: TLabel;  
CBFranchigiaI: TComboBox;  
PMOTO: TPanel;  
PMOTO\_P: TPanel;  
PMOTO\_AN\_P: TPanel;  
PMOTO\_AN\_AC: TPanel;  
PMOTO\_AC: TPanel;  
MainMenu1: TMainMenu;  
Seleziona1: TMenuItem;  
SettoreIeII1: TMenuItem;  
AffariNuovo1: TMenuItem;  
AffariinCorso1: TMenuItem;  
BMFranchigiaAffariNuovi1: TMenuItem;  
SettoreV1: TMenuItem;  
Ciclomotori1: TMenuItem;  
Motocicli1: TMenuItem;  
FranchigiaFissaAffariNuovo1: TMenuItem;  
FranchigiaFissaAffariinCorso1: TMenuItem;  
NoClaimPejus1: TMenuItem;  
Label12: TLabel;  
Label13: TLabel;  
CBCilindrataVM: TComboBox;

```

CBTipoVeicoloVM: TComboBox;
BCALCOLA_VM: TButton;
Label15: TLabel;
CBZonaVM_AN: TComboBox;
Label16: TLabel;
CBFranchigiaVM_AN: TComboBox;
Label14: TLabel;
CBMassimaleVM_AN_P: TComboBox;
Label17: TLabel;
CBMassimaleVM_AC: TComboBox;
Label18: TLabel;
CBclasseDiMeritoVM: TComboBox;
Label19: TLabel;
CBZonaVM_P: TComboBox;
Label20: TLabel;
CBEtaAssicuratoVM: TComboBox;
Label21: TLabel;
CBAnniPatenteVM: TComboBox;
Label22: TLabel;
Label23: TLabel;
RG2ROUTEVM: TRadioGroup;
procedure BCALCOLA_I_IIClick(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure BRESET;
procedure BTOTALE;
procedure FRAZIONAMENTO_;
procedure PREMIIOBASE_;
procedure SETTORE_I_II_ALL;
procedure SETTORE_I_II_AN_AC;
procedure SETTORE_I_II_AC;
procedure SETTORE_I_II_AN;
procedure SETTORE_I_II_AN_FR;
procedure SETTORE_I_II_FR;
procedure RESETPAUTO;
procedure RESETPMOTO;
procedure AffariNuovo1Click(Sender: TObject);
procedure AffariinCorso1Click(Sender: TObject);
procedure BMFranchigiaAffariNuovi1Click(Sender: TObject);
procedure FranchigiaFissaAffariNuovo1Click(Sender: TObject);
procedure FranchigiaFissaAffariinCorso1Click(Sender: TObject);
procedure NoClaimPejus1Click(Sender: TObject);
procedure BCALCOLA_VMClick(Sender: TObject);
procedure SETTORE_VM_ALL;
procedure SETTORE_VM_AN_P;
procedure SETTORE_VM_AN_AC;
procedure SETTORE_VM_AC;
procedure SETTORE_VM_P;

private
  Totale,PremioBase:Real;
  Coefficiente,Settore:string;
  XMLNode: IXMLNode;
public
  { Public declarations }
end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.BRESET;
begin

```

```

LTOTALE.Caption:='0,00';
LLORDO.Caption:='0,00';
PAUTO.Visible:=False;
PMOTO.Visible:=False;
RESETPAUTO;
RESETPMOTO;
end;

procedure TForm1.FormShow(Sender: TObject);
var sPath:string;
begin
Form1.ClientWidth:=555;
PAUTO_AC.Top:=40;
PAUTO_FR.Top:=90;
PMOTO.Left:=0;
PMOTO.Top:=104;
PAUTO.Left:=0;
PAUTO.Top:=104;
sPath:=ExtractFilePath(Application.ExeName);
if (sPath[length(sPath)] <> '\') then
  sPath := sPath + '\';
XMLDATi.FileName:=sPath+'Dati.xml';
XMLDati.Active:=true;
end;

procedure TForm1.PREMIOBASE_;
begin
Premiobase:=StrtoFloat(XMLNode.ChildNodes['PremioBase'].Text);
Totale:=PremioBase;
end;

procedure TForm1.FRAZIONAMENTO_;
begin
XMLNode := XMLDati.DocumentElement.ChildNodes[0];
if(CBFrazionamentoI.Text = 'Trimestrale')
  then begin
    Coefficiente:=XMLNode.ChildNodes['Trimestrale'].Text;
    Totale:=Totale*StrtoFloat(Coefficiente);
  end;
if(CBFrazionamentoI.Text = 'Quadrimestrale')
  then begin
    Coefficiente:=XMLNode.ChildNodes['Quadrimestrale'].Text;
    Totale:=Totale*StrtoFloat(Coefficiente);
  end;
if(CBFrazionamentoI.Text = 'Semestrale')
  then begin
    Coefficiente:=XMLNode.ChildNodes['Semestrale'].Text;
    Totale:=Totale*StrtoFloat(Coefficiente);
  end;
end;

procedure TForm1.SETTORE_I_II_ALL;
begin
if(CBClasseDiMeritoI.Text = '1H')
  then begin
    Coefficiente:=XMLNode.ChildNodes['Classe1H'].Text;
    Totale:=Totale*StrtoFloat(Coefficiente);
  end;
if(CBClasseDiMeritoI.Text = '1G')
  then begin
    Coefficiente:=XMLNode.ChildNodes['Classe1G'].Text;
    Totale:=Totale*StrtoFloat(Coefficiente);
  end;
if(CBClasseDiMeritoI.Text = '1F')
  then begin

```

```

    Coefficiente:=XMLNode.ChildNodes['Classe1F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '1E')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe1E'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '1D')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe1D'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '1C')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe1C'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '1B')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe1B'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '1A')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe1A'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '1')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe1'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '2')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe2'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '3')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe3'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '4')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe4'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '5')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe5'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '6')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe6'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '7')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe7'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '8')
then begin

```

```

    Coefficiente:=XMLNode.ChildNodes['Classe8'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '9')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe9'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '10')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe10'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '11')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe11'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '12')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe12'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '13')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe13'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '14')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe14'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '15')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe15'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '16')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe16'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '17')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe17'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoI.Text = '18')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe18'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBCavalliFiscaliI.Text = 'fino a 8')
then begin
    Coefficiente:=XMLNode.ChildNodes['CF0_8'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBCavalliFiscaliI.Text = 'da 8 a 10')
then begin
    Coefficiente:=XMLNode.ChildNodes['CF8_10'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBCavalliFiscaliI.Text = 'da 10 a 12')
then begin

```

```

    Coefficiente:=XMLNode.ChildNodes['CF10_12'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBCavalliFiscaliI.Text = 'da 12 a 14')
then begin
    Coefficiente:=XMLNode.ChildNodes['CF12_14'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBCavalliFiscaliI.Text = 'da 14 a 16')
then begin
    Coefficiente:=XMLNode.ChildNodes['CF14_16'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBCavalliFiscaliI.Text = 'da 16 a 18')
then begin
    Coefficiente:=XMLNode.ChildNodes['CF16_18'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBCavalliFiscaliI.Text = 'da 18 a 20')
then begin
    Coefficiente:=XMLNode.ChildNodes['CF18_20'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBCavalliFiscaliI.Text = 'oltre 20')
then begin
    Coefficiente:=XMLNode.ChildNodes['CF20_inf'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '1')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona1'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '2')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona2'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '3')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona3'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '4')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona4'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '5')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona5'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '6')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona6'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '7')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona7'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '8')
then begin

```

```

    Coefficiente:=XMLNode.ChildNodes['Zona8'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '9')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona9'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '10')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona10'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '11')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona11'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '12')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona12'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '13')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona13'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '14')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona14'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '15')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona15'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '16')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona16'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '17')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona17'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '18')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona18'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '19')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona19'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '20')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona20'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '21')
then begin

```

```

    Coefficiente:=XMLNode.ChildNodes['Zona21'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaTerritorialeI.Text = '22')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona22'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
end;

procedure TForm1.SETTORE_I_II_AN_AC;
begin
if(CBALimentazioneI.Text = 'benzina')
then begin
    Coefficiente:=XMLNode.ChildNodes['Benzina'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBAlimentazioneI.Text = 'gasolio')and(CBCavalliFiscaliI.ItemIndex<=4))
then begin
    Coefficiente:=XMLNode.ChildNodes['GasolioFino16'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBAlimentazioneI.Text = 'gasolio')and(CBCavalliFiscaliI.ItemIndex>4)
and(CBCavalliFiscaliI.ItemIndex<=6))
then begin
    Coefficiente:=XMLNode.ChildNodes['Gasolio16_20'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBAlimentazioneI.Text = 'gasolio')and(CBCavalliFiscaliI.ItemIndex=7))
then begin
    Coefficiente:=XMLNode.ChildNodes['GasolioOltre20'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBALimentazioneI.Text = 'gas')
then begin
    Coefficiente:=XMLNode.ChildNodes['Gas'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBALimentazioneI.Text = 'elettrica')
then begin
    Coefficiente:=XMLNode.ChildNodes['Elettrica'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBALimentazioneI.Text = 'ibrida')
then begin
    Coefficiente:=XMLNode.ChildNodes['Ibrida'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '18-19')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age18-19F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '20-21')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age20-21F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '22-23')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age22-23F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '24-25')and(RGSessoI.ItemIndex = 0))
then begin

```

```

    Coefficiente:=XMLNode.ChildNodes['Age24-25F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '26-30')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age26-30F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '31-35')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age31-35F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '36-40')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age36-40F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '41-45')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age41-45F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '46-50')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age46-50F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '51-55')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age51-55F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '56-60')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age56-60F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '60-75')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age60-75F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '75+')and(RGSessoI.ItemIndex = 0))
then begin
    Coefficiente:=XMLNode.ChildNodes['AgeOver75F'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;

if((CBEtaProprietarioI.Text = '18-19')and(RGSessoI.ItemIndex = 1))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age18-19M'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '20-21')and(RGSessoI.ItemIndex = 1))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age20-21M'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '22-23')and(RGSessoI.ItemIndex = 1))
then begin
    Coefficiente:=XMLNode.ChildNodes['Age22-23M'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '24-25')and(RGSessoI.ItemIndex = 1))

```

```

then begin
  Coefficiente:=XMLNode.ChildNodes['Age24-25M'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '26-30')and(RGSessoI.ItemIndex = 1))
then begin
  Coefficiente:=XMLNode.ChildNodes['Age26-30M'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '31-35')and(RGSessoI.ItemIndex = 1))
then begin
  Coefficiente:=XMLNode.ChildNodes['Age31-35M'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '36-40')and(RGSessoI.ItemIndex = 1))
then begin
  Coefficiente:=XMLNode.ChildNodes['Age36-40M'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '41-45')and(RGSessoI.ItemIndex = 1))
then begin
  Coefficiente:=XMLNode.ChildNodes['Age41-45M'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '46-50')and(RGSessoI.ItemIndex = 1))
then begin
  Coefficiente:=XMLNode.ChildNodes['Age46-50M'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '51-55')and(RGSessoI.ItemIndex = 1))
then begin
  Coefficiente:=XMLNode.ChildNodes['Age51-55M'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '56-60')and(RGSessoI.ItemIndex = 1))
then begin
  Coefficiente:=XMLNode.ChildNodes['Age56-60M'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '60-75')and(RGSessoI.ItemIndex = 1))
then begin
  Coefficiente:=XMLNode.ChildNodes['Age60-75M'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if((CBEtaProprietarioI.Text = '75+')and(RGSessoI.ItemIndex = 1))
then begin
  Coefficiente:=XMLNode.ChildNodes['AgeOver75M'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;

if(CBRivalsaI.Text = 'fino a 1.500.000 €')
then begin
  Coefficiente:=XMLNode.ChildNodes['RivalsaFino1500k'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBRivalsaI.Text = 'nessuna rivalsa')
then begin
  Coefficiente:=XMLNode.ChildNodes['NoRivalsa'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBRivalsaI.Text = 'intestataro < 26 anni')
then begin
  Coefficiente:=XMLNode.ChildNodes['Inferiore26'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;

```

```

if(CBProvenienzaI.Text = 'evoluzione bonus')
then begin
  Coefficiente:=XMLNode.ChildNodes['EvBonus'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProvenienzaI.Text = 'evoluzione malus')
then begin
  Coefficiente:=XMLNode.ChildNodes['EvMalus'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if (RGPlurisinistrataI.ItemIndex = 0)
then begin
  Coefficiente:=XMLNode.ChildNodes['Plurisinistrata'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if (RGPlurisinistrataI.ItemIndex = 1)
then begin
  Coefficiente:=XMLNode.ChildNodes['NoPlurisinistrata'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
end;

procedure TForm1.SETTORE_I_II_AC;
begin
if(CBMassimaleI_AC.Text = '775.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AC_775000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AC.Text = '1.100.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AC_1100000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AC.Text = '1.600.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AC_1600000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AC.Text = '2.100.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AC_2100000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AC.Text = '2.600.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AC_2600000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AC.Text = '3.700.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AC_3700000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AC.Text = '10.000.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AC_10000000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AC.Text = '26.000.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AC_26000000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
end;

```

```

if(CBMassimaleI_AC.Text = '50.000.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AC_50000000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProfessioneI.Text = 'agente di commercio')
then begin
  Coefficiente:=XMLNode.ChildNodes['AgenteDiCommercio'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProfessioneI.Text = 'agricoltore')
then begin
  Coefficiente:=XMLNode.ChildNodes['Agricoltore'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProfessioneI.Text = 'artigiano')
then begin
  Coefficiente:=XMLNode.ChildNodes['Artigiano'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProfessioneI.Text = 'commerciante')
then begin
  Coefficiente:=XMLNode.ChildNodes['Commerciante'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProfessioneI.Text = 'dimostratore scientif.')
then begin
  Coefficiente:=XMLNode.ChildNodes['DimostratoreScientifico'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProfessioneI.Text = 'dirigente')
then begin
  Coefficiente:=XMLNode.ChildNodes['Dirigente'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProfessioneI.Text = 'lav. autonomo')
then begin
  Coefficiente:=XMLNode.ChildNodes['LavoratoreAutonomo'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProfessioneI.Text = 'libero profess.')
then begin
  Coefficiente:=XMLNode.ChildNodes['LiberoProfessionista'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProfessioneI.Text = 'medico')
then begin
  Coefficiente:=XMLNode.ChildNodes['Medico'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProfessioneI.Text = 'rappresentante')
then begin
  Coefficiente:=XMLNode.ChildNodes['Rappresentante'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBProfessioneI.Text = 'altro')
then begin
  Coefficiente:=XMLNode.ChildNodes['AltreProfessioni'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaVeicoloI.Text = 'fino ad 1 anno')
then begin
  Coefficiente:=XMLNode.ChildNodes['Age1'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;

```

```

if(CBEtaVeicoloI.Text = 'da 2 a 4 anni')
then begin
  Coefficiente:=XMLNode.ChildNodes['Age2-4'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaVeicoloI.Text = '5 anni')
then begin
  Coefficiente:=XMLNode.ChildNodes['Age5'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaVeicoloI.Text = 'da 6 a 10 anni')
then begin
  Coefficiente:=XMLNode.ChildNodes['Age6-10'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaVeicoloI.Text = 'da 11 a 14 anni')
then begin
  Coefficiente:=XMLNode.ChildNodes['Age11-14'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaVeicoloI.Text = 'oltre 14 anni')
then begin
  Coefficiente:=XMLNode.ChildNodes['AgeOltre14'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(RGGuidaEsclusivaI.ItemIndex = 0)
then begin
  Coefficiente:=XMLNode.ChildNodes['GuidaEsclusiva'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(RGGuidaEsclusivaI.ItemIndex = 1)
then begin
  Coefficiente:=XMLNode.ChildNodes['NoGuidaEsclusiva'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
end;

procedure TForm1.SETTORE_I_II_AN_FR;
begin
if(CBMassimaleI_AN.Text = '775.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AN_775000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AN.Text = '3.000.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AN_3000000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AN.Text = '6.000.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AN_6000000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AN.Text = '10.000.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AN_10000000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AN.Text = '26.000.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['AN_26000000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleI_AN.Text = '50.000.000')

```

```

then begin
  Coefficiente:=XMLNode.ChildNodes['AN_5000000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
end;

procedure TForm1.SETTORE_I_II_FR;
begin
if((CBFranchigiaI.Text = '500 €')and(RGFranchigiaI.ItemIndex=0))
  then begin
    Coefficiente:=XMLNode.ChildNodes['FR500LPL'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
if((CBFranchigiaI.Text = '750 €')and(RGFranchigiaI.ItemIndex=0))
  then begin
    Coefficiente:=XMLNode.ChildNodes['FR750LPL'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
if((CBFranchigiaI.Text = '500 €')and(RGFranchigiaI.ItemIndex=1))
  then begin
    Coefficiente:=XMLNode.ChildNodes['FR500ALTRO'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
if((CBFranchigiaI.Text = '750 €')and(RGFranchigiaI.ItemIndex=1))
  then begin
    Coefficiente:=XMLNode.ChildNodes['FR750ALTRO'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
end;

procedure TForm1.SETTORE_I_II_AN;
begin
if(CBEtaVeicoloII.Text = 'fino ad 1 anno')
  then begin
    Coefficiente:=XMLNode.ChildNodes['nAge1'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
if(CBEtaVeicoloII.Text = 'da 1 a 4 anni')
  then begin
    Coefficiente:=XMLNode.ChildNodes['nAge2-4'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
if(CBEtaVeicoloII.Text = 'da 4 a 5 anni')
  then begin
    Coefficiente:=XMLNode.ChildNodes['nAge5'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
if(CBEtaVeicoloII.Text = 'da 6 a 10 anni')
  then begin
    Coefficiente:=XMLNode.ChildNodes['nAge6-10'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
if(CBEtaVeicoloII.Text = 'da 11 a 14 anni')
  then begin
    Coefficiente:=XMLNode.ChildNodes['nAge11-14'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
if(CBEtaVeicoloII.Text = 'oltre 14 anni')
  then begin
    Coefficiente:=XMLNode.ChildNodes['nAgeOltre14'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
if(CBEtaProfessioneII.Text = 'dipendente')
  then begin
    Coefficiente:=XMLNode.ChildNodes['dipendente'].Text;

```

```

    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaProfessioneII.Text = 'artigiano/commerciante')
then begin
    Coefficiente:=XMLNode.ChildNodes['artigiano_commerciante'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaProfessioneII.Text = 'libero_professionista')
then begin
    Coefficiente:=XMLNode.ChildNodes['nAgeOltre14'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaProfessioneII.Text = 'pensionato/casalinga')
then begin
    Coefficiente:=XMLNode.ChildNodes['pensionato_casalinga'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaProfessioneII.Text = 'studente')
then begin
    Coefficiente:=XMLNode.ChildNodes['studente'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaProfessioneII.Text = 'disoccupato')
then begin
    Coefficiente:=XMLNode.ChildNodes['disoccupato'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBAnniPatenteI.Text = 'fino ad 1 anno/senza patente/minorenne')
then begin
    Coefficiente:=XMLNode.ChildNodes['PAT1'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBAnniPatenteI.Text = 'da 2 a 3 anni')
then begin
    Coefficiente:=XMLNode.ChildNodes['PAT2_3'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBAnniPatenteI.Text = 'da 4 a 5 anni')
then begin
    Coefficiente:=XMLNode.ChildNodes['PAT4_5'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBAnniPatenteI.Text = 'oltre 5 anni')
then begin
    Coefficiente:=XMLNode.ChildNodes['PAT5_inf'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(RGRiservataI.ItemIndex = 0)
then begin
    Coefficiente:=XMLNode.ChildNodes['GuidaRiservata'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(RGRiservataI.ItemIndex = 1)
then begin
    Coefficiente:=XMLNode.ChildNodes['NoGuidaRiservata'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBRischioI.Text = 'plurisinistrata')
then begin
    Coefficiente:=XMLNode.ChildNodes['RPlurisinistrata'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBRischioI.Text = 'no plurisinistrata')
then begin
    Coefficiente:=XMLNode.ChildNodes['RNOPlurisinistrata'].Text;

```

```

    Totale:=Totale*StrToFloat(Coefficiente);
end;
end;

procedure TForm1.BTOTALE;
begin
LTOTALE.Caption:=FloatToStr(SimpleRoundTo(Totale,-2));
XMLNode := XMLDati.DocumentElement.ChildNodes[0];
LLORDO.Caption:=FloatToStr
(SimpleRoundTo(Totale*StrToFloat(XMLNode.ChildNodes['Lordo'].Text),-2));
end;

procedure TForm1.RESETPAUTO;
begin
PAUTO_AC.Visible:=False;
PAUTO_AN.Visible:=False;
PAUTO_AN_AC.Visible:=False;
PAUTO_AN_FR.Visible:=False;
PAUTO_FR.Visible:=False;
end;

procedure TForm1.RESETPMOTO;
begin
PMOTO_P.Visible:=False;
PMOTO_AN_P.Visible:=False;
PMOTO_AN_AC.Visible:=False;
PMOTO_AC.Visible:=False;
end;

procedure TForm1.BCALCOLA_I_IIClick(Sender: TObject);
begin
XMLNode := XMLDati.DocumentElement.ChildNodes[1];
PREMIOBASE_;
FRAZIONAMENTO_;
XMLNode := XMLDati.DocumentElement.ChildNodes[1];
if (Settore = 'Settore I e II - Affari Nuovo')
then begin
SETTORE_I_II_ALL;
SETTORE_I_II_AN;
SETTORE_I_II_AN_AC;
SETTORE_I_II_AN_FR;
end;
if (Settore = 'Settore I e II - Affari in Corso')
then begin
SETTORE_I_II_ALL;
SETTORE_I_II_AN_AC;
SETTORE_I_II_AC;
end;
if (Settore = 'Settore I e II - Franchigia - Affari Nuovo')
then begin
SETTORE_I_II_ALL;
SETTORE_I_II_FR;
SETTORE_I_II_AN_FR;
end;
BTOTALE;
end;

procedure TForm1.AffariNuovo1Click(Sender: TObject);
begin
BRESET;
PAUTO.Visible:=True;
RESETPAUTO;
PAUTO_AN.Visible:=True;
PAUTO_AN_AC.Visible:=True;
PAUTO_AN_FR.Visible:=True;

```

```

    PAUTO_AN_FR.Top:=312;
    Settore:='Settore I e II - Affari Nuovo';
end;

procedure TForm1.AffariinCorso1Click(Sender: TObject);
begin
    BRESET;
    PAUTO.Visible:=True;
    RESETPAUTO;
    PAUTO_AC.Visible:=True;
    PAUTO_AN_AC.Visible:=True;
    Settore:='Settore I e II - Affari in Corso';
end;

procedure TForm1.BMFranchigiaAffariNuovi1Click(Sender: TObject);
begin
    BRESET;
    PAUTO.Visible:=True;
    RESETPAUTO;
    PAUTO_FR.Visible:=True;
    PAUTO_AN_FR.Visible:=True;
    PAUTO_AN_FR.Top:=170;
    Settore:='Settore I e II - Franchigia - Affari Nuovo';
end;

procedure TForm1.FranchigiaFissaAffariNuovo1Click(Sender: TObject);
begin
    BRESET;
    PMOTO.Visible:=True;
    RESETPMOTO;
    PMOTO_AN_P.Visible:=TRUE;
    PMOTO_AN_AC.Visible:=TRUE;
    Settore := 'Settore V - Franchigia Fissa Affari Nuovo'
end;

procedure TForm1.FranchigiaFissaAffariinCorso1Click(Sender: TObject);
begin
    BRESET;
    PMOTO.Visible:=True;
    RESETPMOTO;
    PMOTO_AC.Visible:=TRUE;
    PMOTO_AN_AC.Visible:=TRUE;
    PMOTO_AC.Top:=120;
    Settore := 'Settore V - Franchigia Fissa Affari in Corso'
end;

procedure TForm1.NoClaimPejus1Click(Sender: TObject);
begin
    BRESET;
    PMOTO.Visible:=True;
    RESETPMOTO;
    PMOTO_P.Visible:=TRUE;
    PMOTO_AN_P.Visible:=TRUE;
    PMOTO_AN_P.Top:=56;
    PMOTO_P.Top:=103;
    Settore := 'Settore V - No Claim Pejus'
end;

procedure TForm1.BCALCOLA_VMClick(Sender: TObject);
begin
XMLNode := XMLDati.DocumentElement.ChildNodes[2];
PREMIOBASE_;
FRAZIONAMENTO_;
XMLNode := XMLDati.DocumentElement.ChildNodes[2];

```

```

if (Settore = 'Settore V - Franchigia Fissa Affari Nuovo')
then begin
SETTORE_VM_ALL;
SETTORE_VM_AN_P;
SETTORE_VM_AN_AC;
end;
if (Settore = 'Settore V - Franchigia Fissa Affari in Corso')
then begin
SETTORE_VM_ALL;
SETTORE_VM_AN_AC;
SETTORE_VM_AC;
end;
if (Settore = 'Settore V - No Claim Pejus')
then begin
SETTORE_VM_ALL;
SETTORE_VM_P;
SETTORE_VM_AN_P;
end;
BTOTALE;
end;

procedure TForm1.SETTORE_VM_ALL;
begin
if(CBCilindrataVM.Text = '51-399')
then begin
Coefficiente:=XMLNode.ChildNodes['C51_399'].Text;
Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBCilindrataVM.Text = '400-599')
then begin
Coefficiente:=XMLNode.ChildNodes['C400_599'].Text;
Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBCilindrataVM.Text = '600-899')
then begin
Coefficiente:=XMLNode.ChildNodes['C600_899'].Text;
Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBCilindrataVM.Text = '900+')
then begin
Coefficiente:=XMLNode.ChildNodes['C900'].Text;
Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBTipoVeicoloVM.Text = 'Motocicletta')
then begin
Coefficiente:=XMLNode.ChildNodes['Motocicletta'].Text;
Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBTipoVeicoloVM.Text = 'Scooter')
then begin
Coefficiente:=XMLNode.ChildNodes['Scooter'].Text;
Totale:=Totale*StrToFloat(Coefficiente);
end;
end;

procedure TForm1.SETTORE_VM_AN_P;
begin
if(CBMassimaleVM_AN_P.Text = '775.000')
then begin
Coefficiente:=XMLNode.ChildNodes['VMANP_775000'].Text;
Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleVM_AN_P.Text = '3.000.000')
then begin
Coefficiente:=XMLNode.ChildNodes['VMANP_3000000'].Text;

```

```

    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleVM_AN_P.Text = '6.000.000')
then begin
    Coefficiente:=XMLNode.ChildNodes['VMANP_6000000'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleVM_AN_P.Text = '10.000.000')
then begin
    Coefficiente:=XMLNode.ChildNodes['VMANP_10000000'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
end;

```

```

procedure TForm1.SETTORE_VM_AN_AC;
begin
if(CBZonaVM_AN.Text = '1')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona1'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '2')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona2'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '3')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona3'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '4')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona4'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '5')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona5'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '6')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona6'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '7')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona7'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '8')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona8'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '9')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona9'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '10')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona10'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;

```

```

end;
if(CBZonaVM_AN.Text = '11')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona11'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '12')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona12'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '13')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona13'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '14')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona14'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '15')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona15'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '16')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona16'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_AN.Text = '17')
then begin
    Coefficiente:=XMLNode.ChildNodes['Zona17'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBFranchigiaVM_AN.Text = '0 €')
then begin
    Coefficiente:=XMLNode.ChildNodes['VMF0'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBFranchigiaVM_AN.Text = '258 €')
then begin
    Coefficiente:=XMLNode.ChildNodes['VMF258'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBFranchigiaVM_AN.Text = '516 €')
then begin
    Coefficiente:=XMLNode.ChildNodes['VMF516'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
end;

procedure TForm1.SETTORE_VM_AC;
begin
if(CBMassimaleVM_AC.Text = '775.000')
then begin
    Coefficiente:=XMLNode.ChildNodes['VMAC_775000'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleVM_AC.Text = '1.100.000')
then begin
    Coefficiente:=XMLNode.ChildNodes['VMAC_1100000'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
end;

```

```

if(CBMassimaleVM_AC.Text = '1.600.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['VMAC_1600000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleVM_AC.Text = '2.100.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['VMAC_2100000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleVM_AC.Text = '2.600.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['VMAC_2600000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleVM_AC.Text = '3.700.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['VMAC_3700000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBMassimaleVM_AC.Text = '5.200.000')
then begin
  Coefficiente:=XMLNode.ChildNodes['VMAC_5200000'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;

end;

procedure TForm1.SETTORE_VM_P;
begin
if(CBClasseDiMeritoVM.Text = '1C')
then begin
  Coefficiente:=XMLNode.ChildNodes['Classe1C'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoVM.Text = '1B')
then begin
  Coefficiente:=XMLNode.ChildNodes['Classe1B'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoVM.Text = '1A')
then begin
  Coefficiente:=XMLNode.ChildNodes['Classe1A'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoVM.Text = '1')
then begin
  Coefficiente:=XMLNode.ChildNodes['Classe1'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoVM.Text = '2')
then begin
  Coefficiente:=XMLNode.ChildNodes['Classe2'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoVM.Text = '3')
then begin
  Coefficiente:=XMLNode.ChildNodes['Classe3'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoVM.Text = '4')
then begin
  Coefficiente:=XMLNode.ChildNodes['Classe4'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
end;

```

```

if(CBClasseDiMeritoVM.Text = '5')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe5'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoVM.Text = '6')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe6'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoVM.Text = '7')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe7'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoVM.Text = '8')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe8'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBClasseDiMeritoVM.Text = '9')
then begin
    Coefficiente:=XMLNode.ChildNodes['Classe9'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '1')
then begin
    Coefficiente:=XMLNode.ChildNodes['PZona1'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '2')
then begin
    Coefficiente:=XMLNode.ChildNodes['PZona2'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '3')
then begin
    Coefficiente:=XMLNode.ChildNodes['PZona3'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '4')
then begin
    Coefficiente:=XMLNode.ChildNodes['PZona4'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '5')
then begin
    Coefficiente:=XMLNode.ChildNodes['PZona5'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '6')
then begin
    Coefficiente:=XMLNode.ChildNodes['PZona6'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '7')
then begin
    Coefficiente:=XMLNode.ChildNodes['PZona7'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '8')
then begin
    Coefficiente:=XMLNode.ChildNodes['PZona8'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
end;

```

```

if(CBZonaVM_P.Text = '9')
then begin
  Coefficiente:=XMLNode.ChildNodes['PZona9'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '10')
then begin
  Coefficiente:=XMLNode.ChildNodes['PZona10'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '11')
then begin
  Coefficiente:=XMLNode.ChildNodes['PZona11'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '12')
then begin
  Coefficiente:=XMLNode.ChildNodes['PZona12'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '13')
then begin
  Coefficiente:=XMLNode.ChildNodes['PZona13'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBZonaVM_P.Text = '14')
then begin
  Coefficiente:=XMLNode.ChildNodes['PZona14'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBAnniPatenteVM.Text = 'fino ad 1 anno/senza patente/minorenne')
then begin
  Coefficiente:=XMLNode.ChildNodes['PAT1'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBAnniPatenteVM.Text = 'da 2 a 3 anni')
then begin
  Coefficiente:=XMLNode.ChildNodes['PAT2_3'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBAnniPatenteVM.Text = 'da 4 a 5 anni')
then begin
  Coefficiente:=XMLNode.ChildNodes['PAT4_5'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBAnniPatenteVM.Text = 'oltre 5 anni')
then begin
  Coefficiente:=XMLNode.ChildNodes['PAT5_inf'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaAssicuratoVM.Text = 'fino a 22')
then begin
  Coefficiente:=XMLNode.ChildNodes['EAFino22'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaAssicuratoVM.Text = '23-29')
then begin
  Coefficiente:=XMLNode.ChildNodes['EA23_29'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;
if(CBEtaAssicuratoVM.Text = '30-49')
then begin
  Coefficiente:=XMLNode.ChildNodes['EA30_49'].Text;
  Totale:=Totale*StrToFloat(Coefficiente);
end;

```

```
if(CBEtaAssicuratoVM.Text = '50+')
  then begin
    Coefficiente:=XMLNode.ChildNodes['EA50'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
if(RG2ROUTEVM.ItemIndex = 0)
  then begin
    Coefficiente:=XMLNode.ChildNodes['B2R'].Text;
    Totale:=Totale*StrToFloat(Coefficiente);
  end;
end;

end.
```