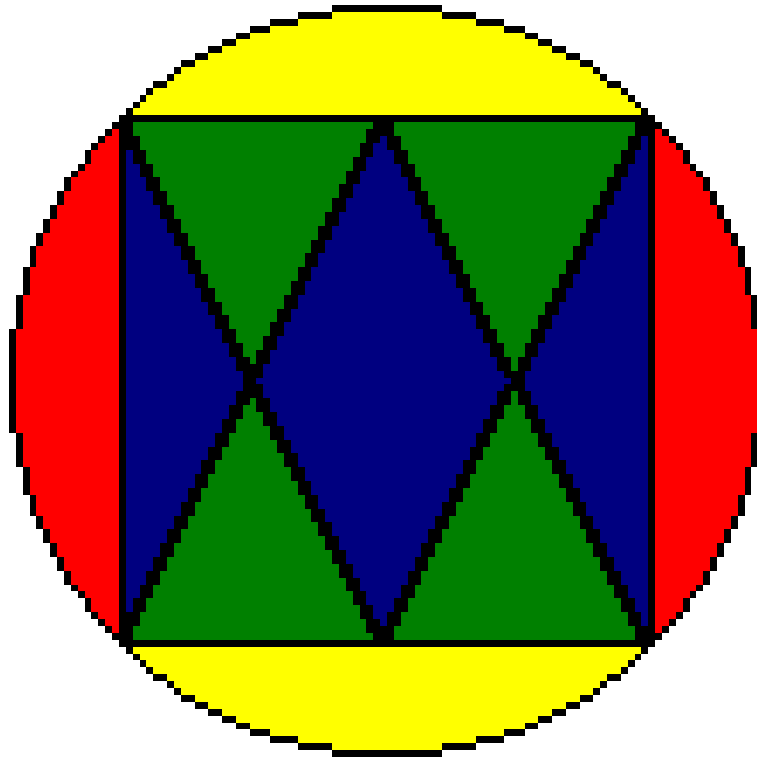


Reward Card System

for



Hela,

the design fashion salon!

By Tim Guhl

Analysis :	4
Background:	4
Problem:	4
The end user:	4
Evaluation criteria:	5
<i>Qualitative:</i>	5
<i>Quantitative:</i>	5
Alternative systems:	5
Design :	7
Tasks and Subtasks:	7
Mail merge tools I will use:	8
<i>Queries:</i>	8
<i>If then else:</i>	8
<i>Fill in:</i>	9
<i>Skip record if:</i>	9
Requirement specification:	9
Software:	9
Hardware:	10
Relate tasks to IT:	10
Implementation :	16
What do I have now?	16
<i>Database fields:</i>	16
<i>Documents:</i>	16
The database:	17
The documents:	18
<i>MM Birthday:</i>	20
<i>MM Information:</i>	22
<i>MM Invitation:</i>	25
<i>MM New Year:</i>	28
<i>MM Sell address:</i>	30
<i>MM Welcome:</i>	32
<i>MM Enter Points:</i>	36
Changes I made during implementation:	37

Testing and Evaluation:	39
Testing plan:	39
Testing:	39
<i>MM Header:</i>	39
<i>MM Birthday:</i>	39
<i>MM Information:</i>	40
<i>MM Invitation:</i>	40
<i>MM New Year:</i>	40
<i>MM Welcome:</i>	40
Evaluation:	41

Analysis:

Background:

Hela is a shop selling clothes to private customers. They want to introduce a reward scheme similar to the one used by Sainsbury's and Tesco's, only the user wants it to create a even more personal User-Customer-Relationship.

The customer can have different levels of cards. Normally he has the normal one and he can be risen to a higher level either by paying a certain amount of money or by creating a certain turnover. Whenever the customer spends a certain sum he earns a voucher. As soon as he earned a certain number of vouchers he will hand them in and receives a price. Details for the amount to spent and number of vouchers necessary vary and will be posted in the shop.

The user wants to use mail merging to send out invitations and greeting cards as well as information about articles.

Problem:

Create a mail merge environment to print out standard documents, that will make the customer think it was a personal letter. Create user guide that will, combined with one major training session (plan), enable the user to set up new documents.

The end user:

The end user is the owner of the shop. He used the computer to do word processing before. He is not very IT-literate otherwise. He did use word and that's why I thought I should be using this was well. He will start a Word course, which will teach him the basic aspects Word soon. Therefore he will be able to open and close documents and format text. He will be familiar with a few aspects of the system.

The user will have employees and he hopes that he will need only a minimum of training sessions to enable them to use the system. He is planning to use one computer to do the printing and one computer connected to the other one via a serial cable which is set up in the shop to enter the reward points while selling.

Evaluation criteria:

Qualitative:

1. Most commonly used documents included. These have to be build up so that the user can change them to fit his needs better.
2. User friendly.
3. User can extend the system with aid of the help-documentation.
4. The customer thinks he is being treated very personally.
5. The customers have to be satisfied.

Quantitative:

1. Minimum of Hardware and Software required because of financial problems of the shop.
2. System has to be capable of holding data of 2500 customers.
3. The printer has to be able to print colour and black and white.
4. The scheme must increase the number of customers so that the growth rate after one year exceeds the normal one by more than 2%.

Alternative systems:

I could have used a template but this would not have been as flexible and it would have required more IT skills by the user. Also it would have needed more work done by the user all the time.

You could use standard documents, which are photocopied, as you need them. The customers would feel special as they get at least more mail than before but it would be obvious that the document had been copied and sent to anyone whose address was known.

You could also have the documents word-processed without a date, a name and an address. Then you fill in the addresses manually and print out a copy with each address. The problem will be that it would be a huge amount of work to fit in all those addresses. Also you would not have the documents specific enough related to the customer.

The last solution is the one I chose to use, and that is the mail merge. Mail merge documents have two components, the document and the database. The data in the database is used (either displayed or used for decisions) by the documents. Combinations of the data and of standard text as well as text entered specifically each time will enable the user to create very customer specific documents. Therefore the user will think the document was written by someone and

only for him. That will make him feel very important and he will think: "The owner of this shop seems to like me. He knows me quite personally. I will go shopping at my friends shop more often than at someone else's shop, even if it is a little bit more expensive." My set-up has to be able to "fool" the customer in exactly this way.

Design:

Tasks and Subtasks:

Task 1: Create standard documents. These are the documents that will come with the program.

They might be quite specialised but they are the foreseeable user requirements.

Subt. a: Have to make the customer think the document was written only to him and no one else.

Subt. b: Print cheap for lower level of card. Can be more costly for gold card.

Subt. c: The customers' member-card so that it only has to be laminated.

Subt. d: Happy birthday: Automatically printed a few days before the birthday to be sent out by post.

Subt. e: Invitation: The customers will be invited to events like a social diners or to a catwalk show.

Different events for certain kind of card only, reservation for certain customers, meal or drinks for gold card only... Special comments for different card status.

Subt. f: Notification of gold card level. If not because of annual turnover then bill £30.

Subt. g: Welcome new applicant.

Subt. h: Information on new products: Depending on card including picture of product (costs of colour print).

Subt. i: Document to forward information of customers to other companies.

Task 2: Create database of names and addresses, birthday, sex. Which extra fields are required obvious as soon as I planned the documents.

Subt. a: Do not use a different software-package if possible.

Subt. b: Must hold: Name, Surname, Address (Street, town, county and postcode), birthday, sex and any extras as discovered during plan of documents.

Subt. c: Easy to enter new or to alter data of existing records.

Task 3: Create a form collecting all the data necessary and still having an attracting visual apparel.

Subt. a: Must get all data necessary for the database.

Subt. b: Must be designed to attract customers.

Subt. c: Must explain what the purpose of the scheme is.

Subt. d: Not too costly to print. (Print on normal printer or printed by a printing office?)

Task 4: Create a user interface:

Subt. a: Add a second way to start Word with the user interface loaded.

Subt. b: Customise toolbars so that the user only has the buttons he needs.

Task 5: Create teaching plan.

Subt. a: Create the plan of teaching session which, combined with the user guide, will enable the user to create new documents.

Subt. b: Enable experienced user (used all mentioned tools... before) to give the teaching session.

Task 6: Create user guide enabling the user to create new documents.

Subt. a: Clear description how to use existing documents and the database.

Subt. b: How to extend the database.

Subt. c: How to enter new or alter customers.

Subt. d: How to create new documents.

Subt. e: How to use tools like if-statements and queries.

Subt. f: Where to get help form.

Task 7: Create a teaching session that will cover all aspects necessary to enable the user to use the program effectively.

Mail merge tools I will use:

Here I will explain which tools I will use to achieve the output. I will use:

- Queries
- If-then-else statements
- Fill ins
- I first wanted to use Skip records if

Queries:

I will use them to select records e.g. for the birthday document. You set up a query in the mail merge set up box. It is the 3rd point after document and data source. The program will only create a document out of the mail merge data if the statement is true.

If then else:

You can find this command when setting up the merge document itself. You click on the command and you will be in a dialog box helping you to set it up. The syntax is easy: **if** statement **then** command or text 1 **else** command or text 2. If the statement is true then

command or text 1 will be fitted into the document, otherwise the command or text 2 will be executed. I will use this to fit in graphics for gold card members only.

Fill in:

Like the if then else this command is fitted into the document while editing it. When you fit it in you will be in a dialog box helping you to set it up. You have to put in a question and a standard text.

As soon as you mail merge the document the program will ask the user to enter some text into the fill in box. If he does not the standard text is entered, otherwise the text the user typed in will be put into every single document.

Skip record if:

This was the old version of the query option. You entered a criterion and if that was true the program went on the next document without making any further changes. I first wanted to use this as I did not know about the query, but when I looked up the syntax I was told by help that it was more effective to use the query. I must agree now that I tried it out.

Requirement specification:

I will need a software holding the data and one to edit the documents in. I also need some link between the document and the database. The easiest way is to have everything in one software package. I could use a table in a word processor, but I think it would be more user friendly if I used the database interface offered by sophisticated software like Word. This interface will automatically save the data in tables and enable the user to retrieve and change the data at any time. Word mail merge will give me all features necessary like e.g. the fill in box required for task 3 h (customer information.).

Software:

User wants me to use one package only because of the costs. It would be appreciated if he could use the software for other tasks as well (e.g. writing to his friends).

Word will enable me to create mail merge documents. I will not need an external database as the database user interface is quite user friendly. This will help the user as good help files are existing, it will keep the links easy and therefore enable other experienced users to help the user and the user can learn more about the software easily. There will be no compatibility

problems. The user can use Word for private matters and to contact providers... It is also highly compatible and widely used enabling the user to ask friends for help if he gets stuck.

Hardware:

The user does not have any hardware but is taking part in a foundation course for skills required to use the computer. He will learn how to use Windows 95 and Word (basic skills like formatting fonts and paragraphs).

To run Windows95 and Word97, which was the software I used to implement this, you do not need much more than a 486 DX2 88 or something similar. You will also not need to much disc space. 200MB for the software and 20MB for the files. You will have noticed that this is far under the systems, which you might be offered in many shops at the moment, so that the user can buy any machine he wants.

He will also need some device to enable him to import pictures of his dresses into the system. He could either use a scanner or he could buy himself a digital camera. As the prices for the second solution seem to be dropping rather rapidly as well as giving better quality and as he could use the camera in his private life as well I would advice the user to buy him one of these.

Relate tasks to IT:

Task 1: I will have to design every single document on paper. You can find the sketches in the appendix.

The header is always the same. Using if-statements I only included the symbol for gold card members and the name of the shop is printed in colour for gold card members only to. Then it has the address of the shop and the address of the customer (mail merged) which is positioned to fit in the envelope window.

Subt. c: Membership card:

The customers card should be designed so that it is printed and then only has to be laminated before handout.

In my design the card was included in the welcome document (Subtask g).

When the user prints that document out he gets the letter and on the second page (which will be coloured paper if the customer is a gold member) he can cut out the member card. The rest of the page goes into the filing cabinet next to the till. Whenever the customer purchases something the earned reward

points will be it is written down on that card.

For the card itself I only need the title, the first name and the surname.

Subt. d: Happy birthday:

This document is based on a query. The query will select the appropriate records of those customers whose birthday is coming up soon. This date is stored in a special format I created. We want to print out the birthday cards for the next two weeks always. Therefore the date is saved in this way: MM.0 and MM.5 for the 1st till 15th and the 16th till 30th/31st respectively. The user only sets the query to the right value, for which he will have a table in the user guide. After mail merging it he can print the documents and they are ready to go in the post.

Depending on card level the customer will be asked to pick up a surprise only or a drink as well. The formula I will use will look like this: If <> normal (therefore gold or normal gold) then surprise and drink else surprise only. As you can see I will need the following data to be stored in the database: Name and address, birthday and card level.

Subt. e: Invitation:

I will use a fill in text box. This has to be filled in one time for all records and it should be used for a description of the event.

After that I included a long **if then else** statement-chain. It tells the customer that if he is a gold card member, depending on his family status, a number of places have been reserved. If he is not a gold card member it tells him how to get a place reserved.

If card level <> normal **then** (must be gold or normal gold then)

if family status = single **then** “1 place”

else

if family status = family **then** “tell us how many places
you would like to book.”

else (family status must be couple) “2 places have been
reserved”

(add to all of them if you add to end of statement) “free drinks will be
served”.

else (Must be Normal card member) “If you want a reservation on this event
tell us and we will try but gold card members are more important” (to make
them want to be a gold card member).

You noticed that I do need all kind of information this time. I need the Name
and Address, card level and the family status.

Subt. f: Notification of gold card level:

A new years card will be sent to all gold card members. A query will select the
appropriate records (card level <> normal). There will be some text and one if
statement. **If** card level = normal gold **then** “If you want to stay a gold card
member, please pay £30.” I will need the Name and Address and card level.

Subt. g: New applicants welcome:

This document is printed as soon as a new member joins the scheme. It consists
of two sections.

Section 1 is a letter to the customer. It consists of a text and one if statement.

This statement asks all customers wanting to pay to be a gold card user to pay
their £30. (**If** card level = normal gold **then** “If you want to stay a gold card
member, please pay £30.”)

Section 2 will be printed on coloured paper if the customer is a gold card
member. This is so that his status is obvious. The page consists of two parts in
the top. The left part being the address and personal details of the user and the
right part being the membership card. You can cut the membership card out
and laminate it. Underneath these two parts you will have some space to make
note of the reward points and any other information he might want to note.

This card will be in a filing cabinet next to the tilt and the member of staff will
write down how many points the user earned.

I will need the database to hold the following fields for this: Name and Address and card level.

Subt. h: Information on new products:

This document is an advert. In order not to annoy the customers I will give them the opportunity to say they do not want to receive information. To print out this document only for people who want to receive the information I will have a query option set up so that the document is only printed out for those people. The rule should look like this: Information Hela = yes

The document will have a fill in box at the beginning. This will help the user to enter a description of the dress. Underneath that I will have a if statement enabling the user to insert a picture of the dress for gold card members. This will only be for gold card members as the printing costs would be too high otherwise. Also the printing would take much longer.

Therefore this document will retrieve information on Name and Address and information sent from Hela.

Subt. i: Document to forward information of customers to other companies:

This will enable the user to make a little bit of extra money out of the system. He can forward information like the address to other companies. They will pay him for the addresses.

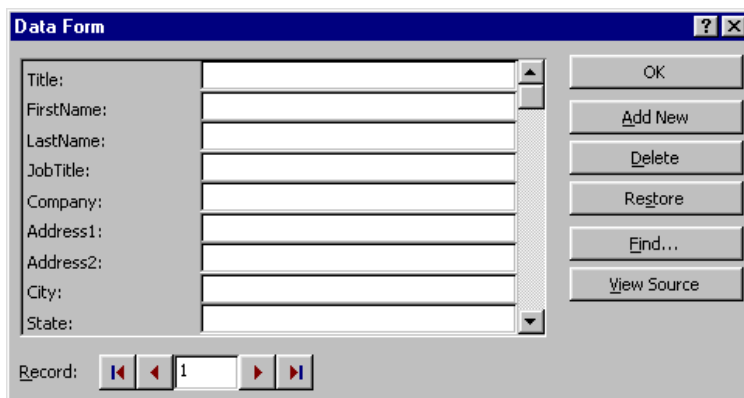
If he just sold the information like this the user would create a conflict with the personal data protection act. Therefore he will have to ask the customer for permission and I will create a query to select only those who agreed to their information to be sold. (information others = yes)

Therefore this document will need the following information. Name and address and information to other companies.

Task 2: Now that I designed the documents I will decide which fields I will have in my database. Please refer to the table below:

Field name	Comment
Title	The title to address the person correctly.
Name	The name for address and card.
Surname	The surname for address, card and to address person.
Street	For the address.
Town	For the address.
Country	For the address.
Postcode	For the address.
Card level	To decide on invitations and colour print.
Birthday	To find out who to send the birthday letter.
Hela information requested	To run query on whether to sent information.
Other information requested	Query on whether to sell address and name to others.
Family status	To decide on how many places to reserve.
Sex	Decided to add later. Can make documents more personal (bring your husband/wife) and ask them to wear certain dress (please in black suit/long skirt)

By using the word data storage device (Tools, Mail Merge, Create Source) I will make it easy for the user to enter and alter data. He can also use the help provided by Word.



You can change this data form to fit your needs. I will only have the fields I want and need. A disadvantage is that you can not

introduce any validation. But the tool is easy to use and the user can refer back to Word help in case of any problems.

Task 3: Pencil sketch

Now I will design the data collection form, which will be available as handouts in the shop. The leaflet must look interesting and make people nosy. It must still get all data

and avoid any false input. Also the form should have a format and order similar to the data entry window to avoid errors while the user copies the data down. These errors are called transcription errors.

First I did this table to plan in general how I can collect the data:

Data required	Input format
Title	<input type="checkbox"/> Mr, <input type="checkbox"/> Mrs, <input type="checkbox"/> Ms.
Name and Surname	Please spell as you want it on your card
Street	
Town	
County	e.g. W. Sussex
Postcode	_____
Sex	<input type="checkbox"/> male, <input type="checkbox"/> female.
Birthday	____.____.19__
Family status	<input type="checkbox"/> single, <input type="checkbox"/> married or long term relationship, <input type="checkbox"/> children.
Product information from Hela	Do you want to receive product information from us?
Product information from other companies	If you do not wish to receive information from other companies, tick here: <input type="checkbox"/>
Card level:	<input type="checkbox"/> Normal <input type="checkbox"/> Gold (£30)

Tasks 4, 5 and 6 will be solved later. It makes more sense to work on these tasks after having done the implementation.

Implementation:

I implemented the system and will now describe it to you. Obviously I did do a few changes during the implementation and although you might notice a few earlier on I will tell you about them in the end.

What do I have now?

To give you an overview and to make the orientation more easy for you I will first list all fields and documents.

Database fields:

- Title (Mr or Mrs)
- Name
- Surname
- Street
- Town
- Country (e.g. East Sussex.)
- Postcode
- Card
- Points (Reward points only for statistical reasons not for mail merge)
- Birthday
- Sex (M or F)
- Family status (Married (two people), single or children (more than two))
- Hela inf (Does the customer want information of Hela)
- Other inf (Does the customer want information of other companies.)

Documents:

- MM Birthday
- MM Enter Points
- MM Header
- MM Information
- MM Invitation
- MM New Year
- MM Sell Address
- MM Welcome

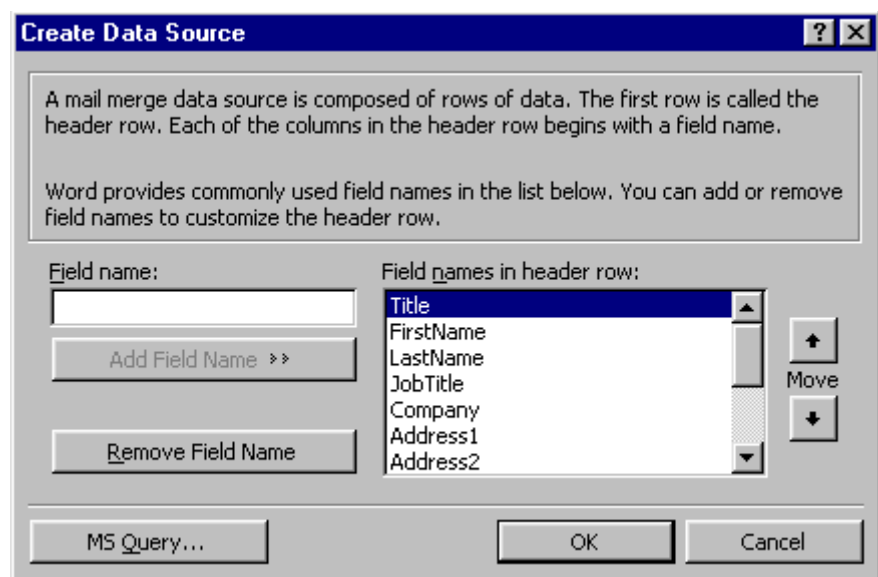
The database:

To be able to create the documents I first had to set up the database. By starting a new document and starting tools mail merge you will end up on a screen leading you through tree steps to create a mail merge document.

Step 1 is to create a document. All my documents (except the one to sell data to other companies) are based on a form letter.

Step 2 is to select/create a data source. Obviously you have to create the data source one time and then you just select that database. To create the database you end up on the following screen:

I removed nearly all the fields and then put in the fields listed above. The program automatically creates a table and saves under your chosen name. If you want to add a field later you just open that file and add a column.



I have one additional field I did not have in the design stage. I added the field points. This will not be used in documents, only to save the reward points in the computer. You could run a query to see who could become a gold card member but this is not the main purpose.

The toolbar:

I also introduced my own toolbar. This will make the system easier to use for the user as it only holds the most important buttons. Later when he is more experienced he can easily open the other toolbars as well and work using all the tasks.

The toolbar looks like this:



The documents:

All document names start with MM for Mail Merge. This is so that you can sort your files very easily. It will also help the user to keep his hard drive tidy. I will always show you several versions of the documents. Most of them will be imported as graphics and will be explained afterwards. I will have as many different versions as necessary to make clear to you what the documents do and how they do it.

[Appendix 2.1](#) The different outputs will be shown as printouts in Appendix 2. The correct number will be given next to the graph in this documentation.

Normal These blue boxes are included to tell you which document you are looking at. It will either say gold or normal. It may look as if it was in the document sometimes but it will always be located over empty space.

The header uses data about card level, title, name and address retrieving it from the database. Using the information on card level the output will either be colour for gold card members or the cheaper version (black and white without a label) for normal card members. To achieve this output I used an If-Then-Else statement. As I have two different descriptions for gold card members (gold or normal gold) I had to set it up the following way: **If** level = normal **then** “the colourful version” **Else** “the colourless version”. I found out that you can also put graphics into the then and else “”. Therefore it was possible to have the logo only for gold card members as well.

The sentence to fit in the logo or not is this:

```

IF MERGFIELD·Card·="Normal"·"Hela"·"Hela"

```

If card = normal then none or black output.

If card = normal then "" else these colour outputs.

```

IF MERGFIELD·Card·="Normal"·""·"

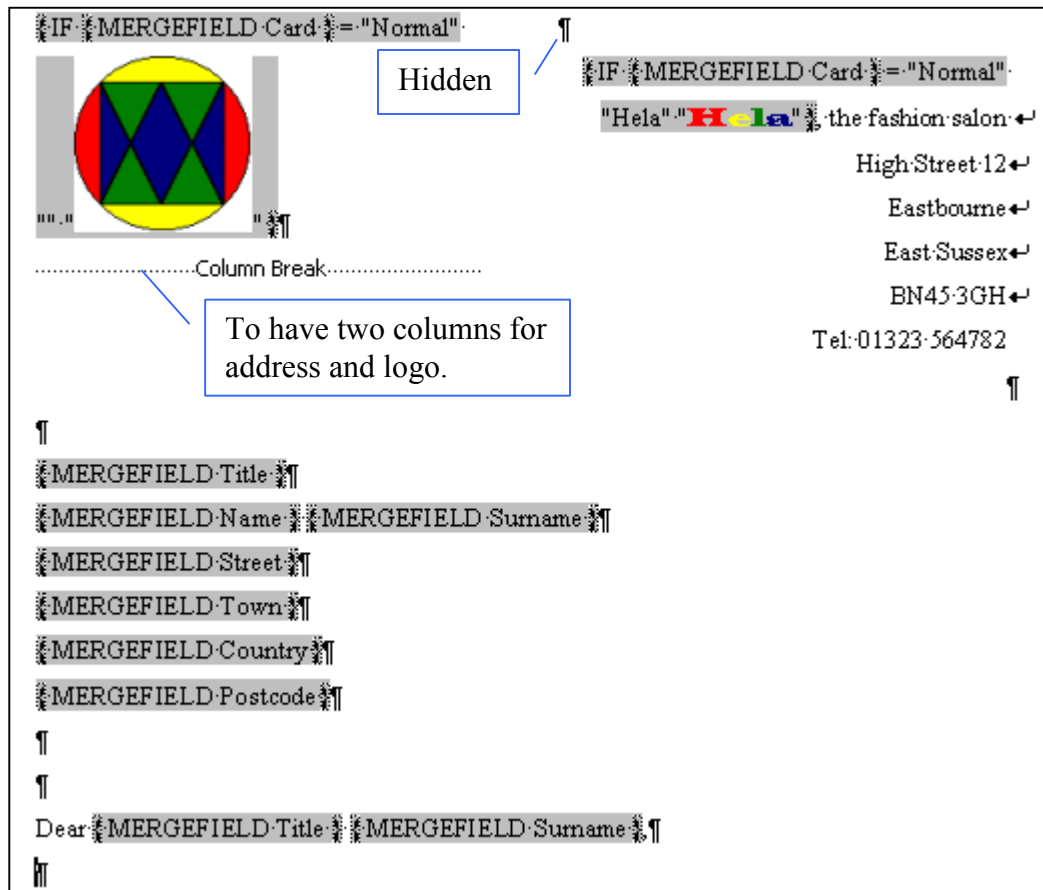
```

The header also contains the address of the shop, which is always the same but will add to customers service.

The next part is the address of the customer. It is located so that you can fit it in the window of an envelope which will save you printing costs and make the sending of the documents easier. It will get all data directly form the database.

After that the customer is addressed personally using his title and his surname. These again are mail merged into the document.

Here I included a screenshot of the whole document showing you the field codes:



MM Birthday:

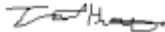
First I would like to show you my output and after that I will explain how I go to this stage.

Hela, the fashion salon
High Street 12
Eastbourne
East Sussex
BN45 3GH
Tel: 01323 564782

Mr
Heimat Hartmann
Highstreet Kensington
London
London
SW24 5GH

Dear Mr Hartmann,

The staff of Hela wishes you a very happy birthday.
If you hand this document in to any member of staff, we will have a little surprise for you.
I am looking forward to continuing the good relationship we have



Normal

Appendix 2.3



Hela, the fashion salon
High Street 12
Eastbourne
East Sussex
BN45 3GH
Tel: 01323 564782

Mrs
Siglinde Fredstein
Ardendale street
Eastbourne
East Sussex
BN27 5HF

Dear Mrs Fredstein,

The staff of Hela wishes you a very happy birthday.
If you hand this document in to any member of staff, we will have a little surprise for you. A
bottle of sparkling wine has been put on ice as well.

I am looking forward to continuing the good relationship we have



Gold

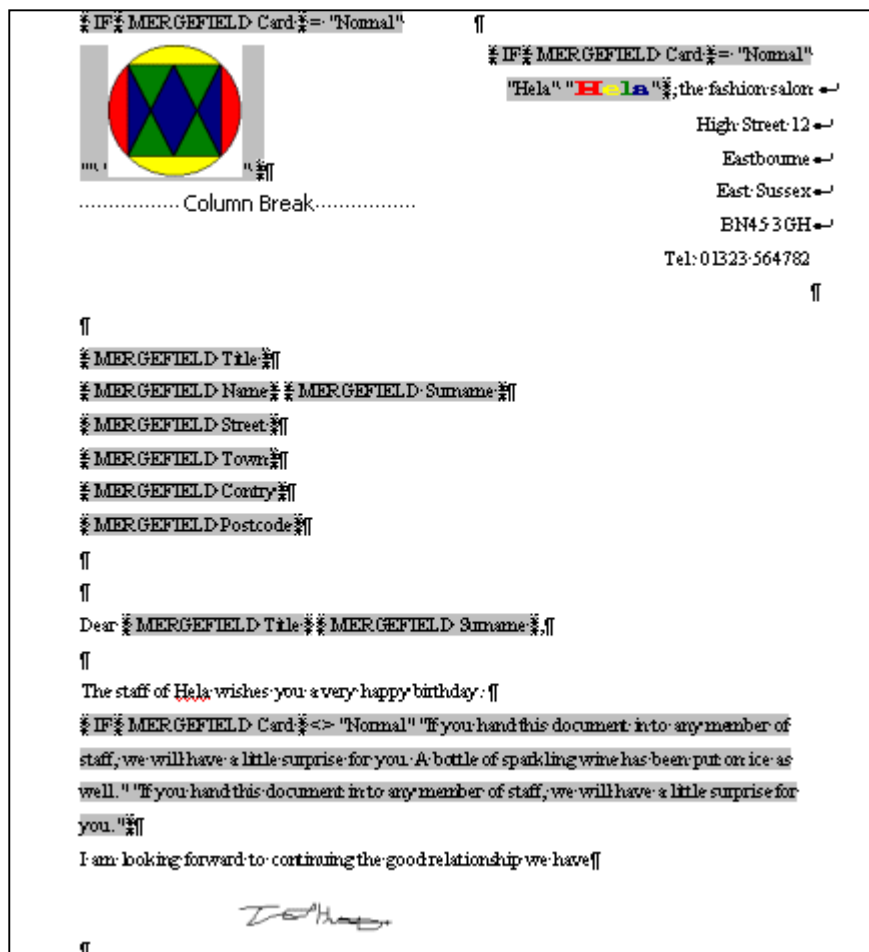
Appendix 2.4

Note difference

How did I make sure this letter is only printed out for people whose birthday it is in the next few days? In the database I have a field called birthday. The birthday of the customers is not saved in the normal way there. My special format works like this. For the first 15 days the value of the field is equal to the number of the month. From then onwards till the end of the month the value is equal to the number plus 0.5 (e.g. Birthday 2nd of February, value in the field: 2; Birthday 18th of February, value in the field 2.5). I then entered the following query in the query option of the document. Birthday equal to "the current number".

How do I want that to work? Well the data is entered when you enter the customers' data. Then on the 27th of the previous month the user inputs the number of the month in the query option. Then he merges the document. On the 12th of the month he will edit the query option again. He will add 0.5 to the current number. After merging the document he will have documents for all those soon to have birthday. He can then send of the letters. It will not matter if the documents arrive there a bit early.

The following graph will show you the field codes enabling me to produce the output:



This document also contains another **if then else** statement as gold card members will get not only a small, valuable present but also glass of sparkling wine. The normal member will only receive the present, which will be smaller in this case. As the customer is show the document the member of staff can see the card level at once. Therefore he knows what kind of present to give the person and whether to open the bottle of sparkling wine or not.

```

IF: MERGEFIELD Card <> "Normal" "If you hand this document in to any member of
staff, we will have a little surprise for you. A bottle of sparkling wine has been put on ice as
well." "If you hand this document in to any member of staff, we will have a little surprise for
you."

```

If-then-else.

MM Information:

First thing I would like to show you my outputs and then later explain how I got them:

```

..... Column Break.....
Hela, the fashion salon
High Street 12
Eastbourne
East Sussex
BN45 3GH
Tel: 01323 564782

Mr
Helmut Hartmann
Highstreet Kensington
London
London
SW24 5GH

Dear Mr Hartmann,

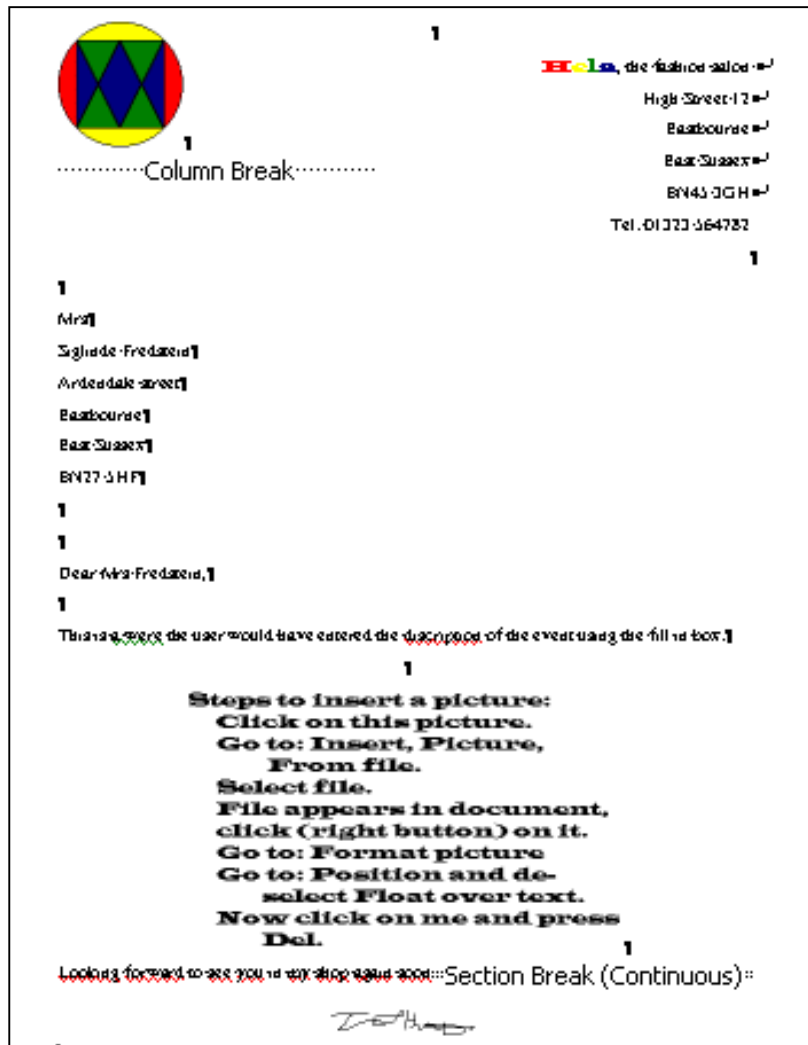
This is where the user would have entered the description of the event using the fill in box.

Looking forward to see you in my shop again soon..... Section Break (Next Page).....

```

Normal

Appendix 2.5



Gold

Appendix 2.6

Now that you have seen both possible outputs I will start to explain to you how I want to achieve them. On little alteration has to be added. The graphic you can see in the gold card output is not the one you would see normally. It is a graphic holding a description for the user how to enter the graphic. He will do this before he mails merges the document. This is the easiest way to make the user proceed in the right way.

Above the graphic you will see a sentence. This sentence will normally be a description of the new article. The user puts it in by filling in a fill in box. This is a very user friendly way to let the user put in his text. If he wants to add a comment to one or two letters only he can mail merge the document into a file first and then edit single document.

After the graphic you have one more standard sentence and the you will have the scanned in signature again.

The code of the fill in box looks like this:

This is the question the user sees when mail merging the document.

~~FILLIN."Please enter the introduction and description here." \d. "The description of the dress can be filled in as soon as you mail merge the document" \o.~~

This is in the document till you fill something in.


Also the graphic is only included in the gold card user document. Therefor it is in a if statement like the logo of the shop in the header.

Here I also enclosed a picture of the field codes:

¶F ¶MEROFIELD Card ¶ - 'Meroid' ¶
 ¶MEROFIELD Card ¶ - 'Meroid' ¶
 ¶Hda' 'H' 'L' ¶ the fobbaa sdaa ¶
 High Sura 12 ¶
 Embauroe ¶
 Emi Sussa ¶
 SM45 3044 ¶
 Tel: 01323 364722 ¶
 ¶
 ¶
 ¶MEROFIELD Title ¶
 ¶MEROFIELD Name: ¶ MEROFIELD Surname ¶
 ¶MEROFIELD Sura ¶
 ¶MEROFIELD Tawa ¶
 ¶MEROFIELD Country ¶
 ¶MEROFIELD Paswade ¶
 ¶
 ¶
 Daa ¶MEROFIELD Title: ¶ MEROFIELD Surname: ¶
 ¶
 ¶FLLY' Please save the introduction and description before! The description of the document can be filled in as soon as you can save the document! ¶
 ¶F ¶MEROFIELD Card ¶ - 'Meroid' ¶
Steps to insert a picture:
Click on this picture.
Go to: Insert, Picture,
From file.
Select file.
File appears in document,
click (right button) on it.
Go to: Format picture
Go to: Position and de-
select Float over text.
Now click on me and press
Del. ¶
 Looking forward to see you in my shop again! ¶
T. Guhl

MM Invitation:

Like every time I will first include snapshots of the output.



Hela, the fashion salon
High Street 12
Eastbourne
East Sussex
BN45 3GH
Tel: 01323 564782

Mrs
Siglinda Fredstein
Ardendale street
Eastbourne
East Sussex
BN27 5HF


Dear Mrs Fredstein,

there will be a band playing at the cathedral.

Two places have been reserved for you and your partner. If you do not plan to attempt the event, could you please cancel the bookings.

Free drinks and snacks will be served.

I am very much looking forward to seeing you on this event.



Gold

Appendix 2.7

Hela, the fashion salon
High Street 12
Eastbourne
East Sussex
BN45 3GH
Tel: 01323 564782

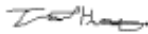
Mr
Helmut Hartmann
Highstreet Kensington
London
London
SW24 5GH

Dear Mr Hartmann,

there will be a band playing at the cathedral.

If you would like to attempt the event please let us know and we will try to reserve places for you. Places will be reserved on first come first serve basis with priority for gold card holders

I am very much looking forward to seeing you on this event.



Normal

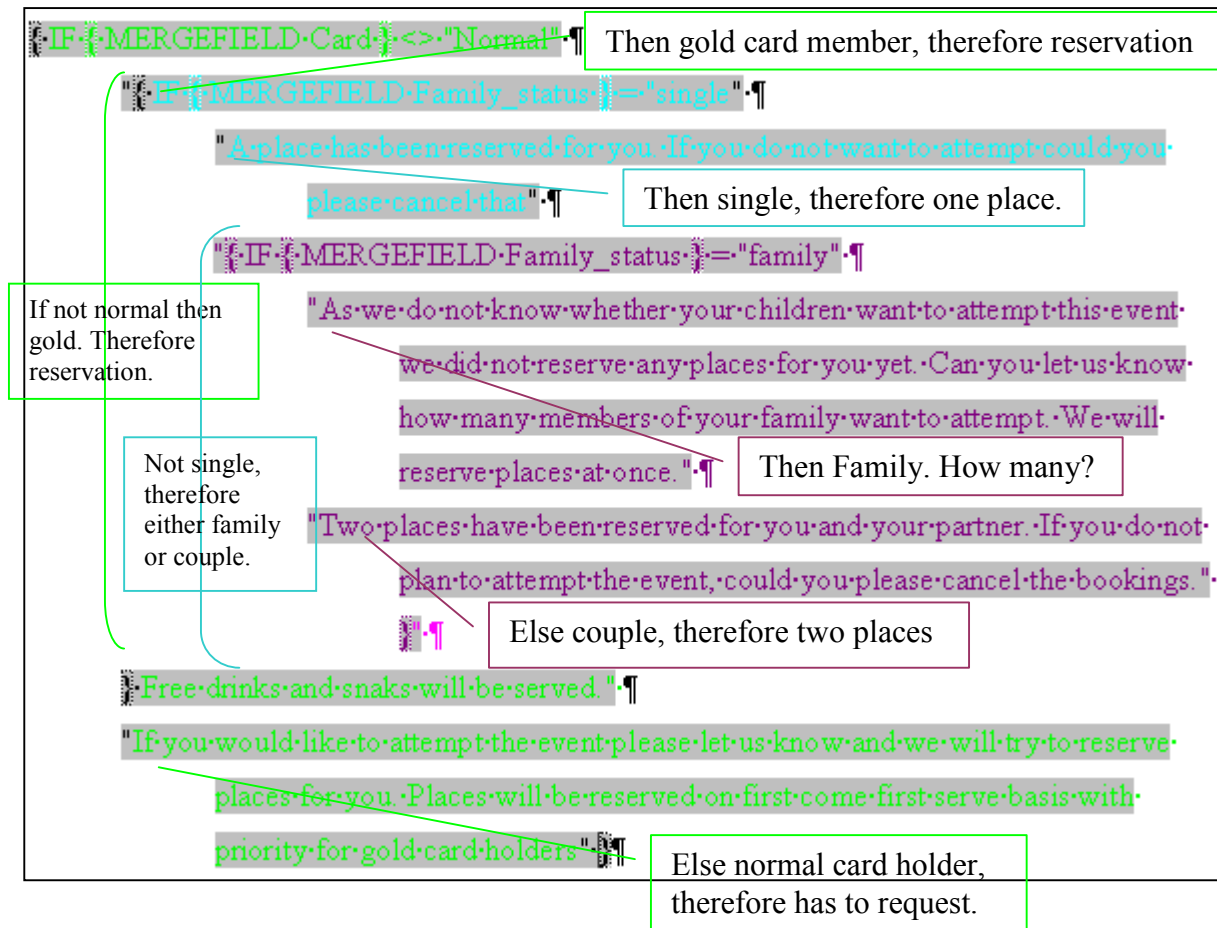
Appendix 2.8

Like the last document this one again uses a fill in box. The sentence “a band will be playing” will normally be a full description written by the user. He will be asked to enter it as soon as he mails merges the program. This time I did not enter a standard text even.

```

FILLIN "Please enter some text describing the event." \o
    
```

The next part is the complicated part with the number of invitations depending on card level and family status: I did colour code it so that it is more obvious and easier to explain.



MM New Year:

And again as normal I will first show you the output.



Hela, the fashion salon
 High Street 12
 Eastbourne
 East Sussex
 BN45 3GH
 Tel: 01323 564782

Mrs
 Siglinde Fredstein
 Ardendale street
 Eastbourne
 East Sussex
 BN27 5HF


Dear Mrs Fredstein,
 A new year has started. By viewing all data from last year I am delighted to tell you that you are registered as a gold card member. I am sure you know what an advantage this is for you.

I am looking forward to another year of co-operation



Gold

Appendix 2.9




Hela, the fashion salon
 High Street 12
 Eastbourne
 East Sussex
 BN45 3GH
 Tel: 01323 564782

Ms
 Anne Marie Trevena
 The Flat
 London
 London
 SE23 5KN

Dear Ms Trevena,
 A new year has started. By viewing all data from last year I am delighted to tell you that you are registered as a gold card member. I am sure you know what an advantage this is for you. Sadly in your case this is connected with a small fee of £30. I am absolutely sure you know that you do get much more out of the card than that and you probably wondered already why we give you this option.

However you would be doing me a big favour if you could transfer the amount to my account within the next month. You can also hand in a cheque in the shop if you wish.

I am looking forward to another year of co-operation



Normal Gold

Appendix 2.10

This document will only be sent to gold card members. These are selected by the query option “card not equal to normal”. The output differs depending on whether the customer is a gold card member that does not have to pay or one that decided the advantages where that big that he wanted to be a gold card member even though he has to pay £30.

The document for the “gold” card member (does not have to pay) gives New Year greetings and tells the person he is a gold card member for free for the coming year. I wrote this in a way that is acceptable for both, the customers who just rose to this level and the once who had this status last year as well.

The document for the “normal gold” card member (does have to pay) also gives the New Year greetings and in addition to that asks the customer to pay the £30.

How did I get this output? Well the New Year greetings are just text and therefore appear in every document. So does the sentence telling the customer that he is entered for this scheme.

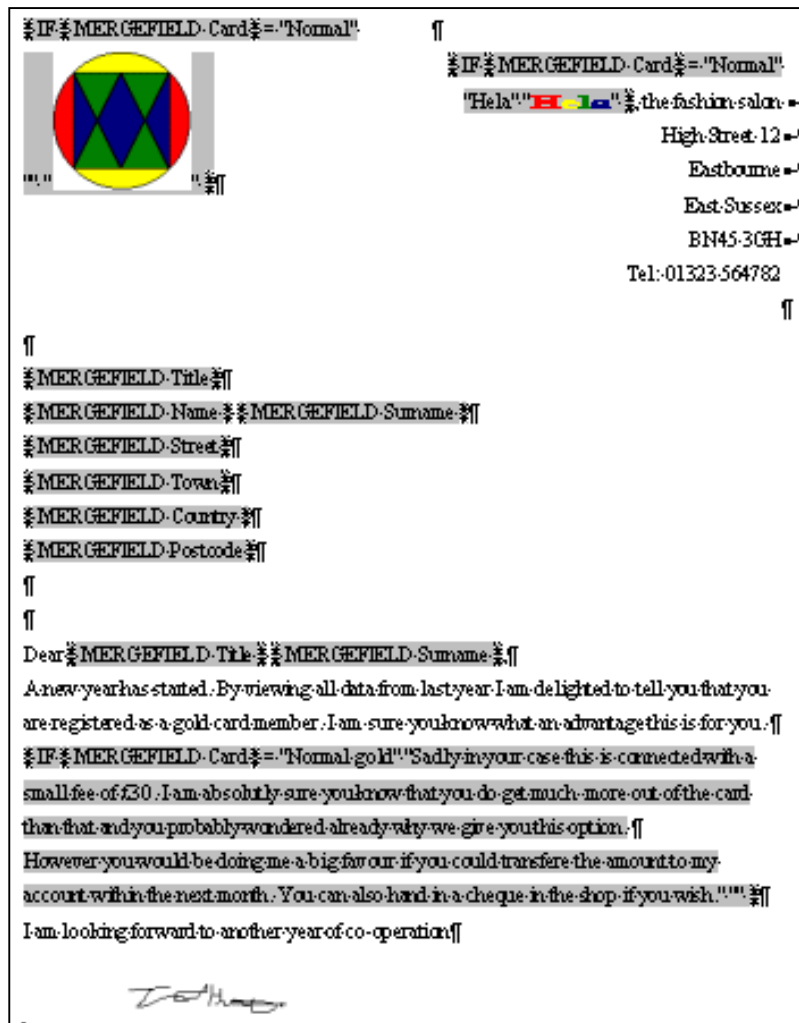
After that I put in a if sentence which only adds the sentence asking the customer for money if the user is a “normal gold” card member.

```

A.new.year.has.started.By.viewing.all.data.from.last.year.I.am.delighted.to.tell.you.that.you
are.registered.as.a.gold.card.member.I.am.sure.you.know.what.an.advantage.this.is.for.you.¶
¶IF MERGEFIELD Card = "Normal gold" ¶ "Sadly.in.your.case.this.is.connected.with.a
small.fee.of.£30.I.am.absolutly.sure.you.know.that.you.do.get.much.more.out.of.the.card
than.that.and.you.probably.wondered.already.why.we.give.you.this.option.¶
However.you.would.be.doing.me.a.big.favour.if.you.could.transfere.the.amount.to.my
account.within.the.next.month.You.can.also.hand.in.a.cheque.in.the.shop.if.you.wish."."."¶

```

And as usual I enclosed the field codes again as well.(next page)



MM Sell address:

And again first the output.

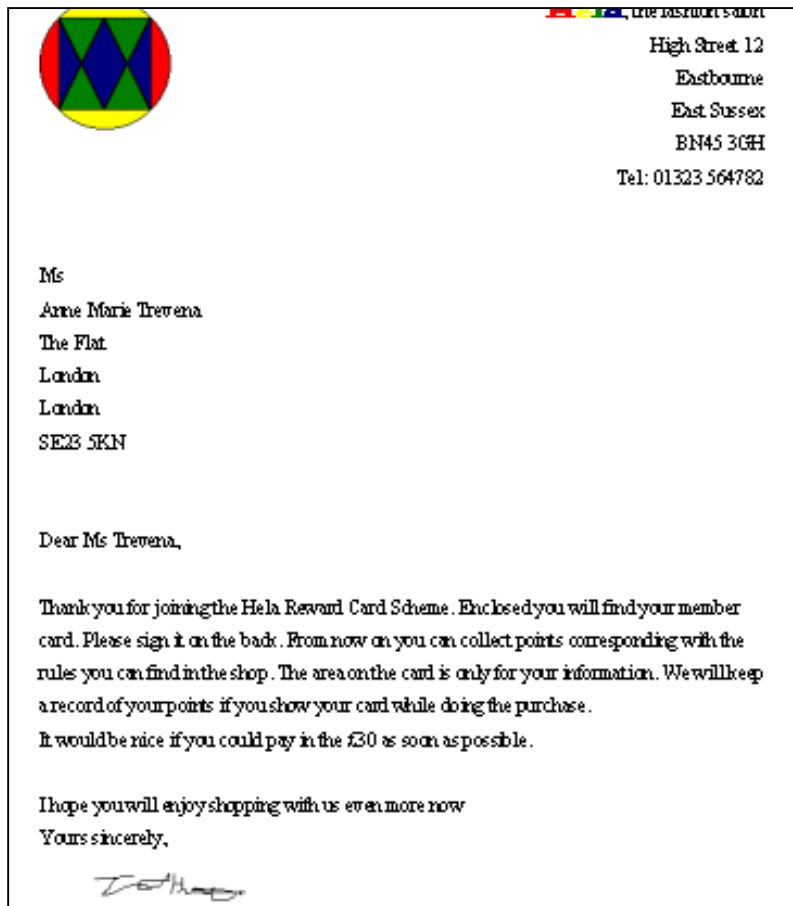
Mr Helmut Hartmann Highstreet Kensington, London London, SW24 5GH. Birthday: 5.5, Sex: m Family status: single Ms Anne Marie Trevena The Flat, London London, SE23 5KN. Birthday: 1, Sex: f Family status: single
--

The entries in this documents are selected by the query “other information = yes”. This document was set up differently. All other documents till know were created by selecting

form letter in the query option. This document was created by selecting Catalog as the document type.

The document gives all data relevant to the other company Hela sells the information to. It will need an explanation of the birthday format to make the other company understand. This can be done by explaining it in the letter that would go with the information. Here I also enclosed a screenshot of the field codes.

```
{MERGEFIELD·Title·} {MERGEFIELD·Name·} {MERGEFIELD·Surname·}
{MERGEFIELD·Street·} {MERGEFIELD·Town·}
{MERGEFIELD·Country·} {MERGEFIELD·Postcode·}
Birthday: {MERGEFIELD·Birthday·}, Sex: {MERGEFIELD·Sex·}
Family·status: {MERGEFIELD·Family_status·}
¶
¶
```

Normal Gold P. 1

Appendix 2.15

I am sure now that you saw this excellent output you would like to know how it was achieved. First though I will tell you why I have two pages for this document. The first page is quite clearly a welcome letter. In the case of the normal gold member it also reminds him that he has to pay for his gold card.

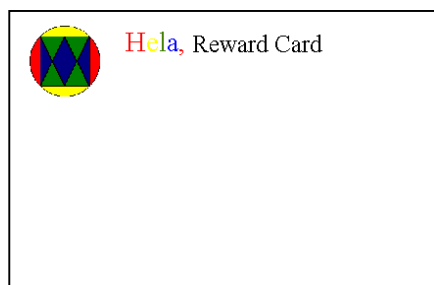
The second page has two purposes. Firstly you can cut out the customer card. You then laminate it and hand it over to the new member. In the case of a gold card member the second page will be printed on coloured paper so that this important person is recognised easily. Next to the customer card you can see nearly all the personal data printed. This document also acts as a backup in case of computer malfunction.

In the section underneath you have a table for the manual reward points collecting. First I only wanted to have this system in which the staff writes down every purchase the customer does. Now I also offer an automated system but the user will have the choice which system to use. The first page consist of MM Header, some text and one if then else statement putting in the request to pay £30 if the member is a normal gold card member.

```
IF MERGEFIELD Card = "Normal-gold"
It-would-be-nice-if-you-could-pay-in-the-£30-as-soon-as-possible."
```

Now I will describe the membership card to you:

As you can see I enlarged the text boxes so that the whole text can be seen.



Putting together several elements produced the output. The first part is this graphic around which I draw a element out of the auto-shapes of word. It is called rounded rectangle. As you could see in the last screenshot I then placed tow text boxes on top of the graph as well. They produced the output of name and customer number.

Next to the reward card I placed another text box containing the following syntax:

```

MERGEFIELD Title MERGEFIELD Name MERGEFIELD Surname
MERGEFIELD Street
MERGEFIELD Town
MERGEFIELD Country
MERGEFIELD Postcode
MERGEFIELD Card
MERGEFIELD Family_status
MERGEFIELD Birthday
    
```

As you can see this textbox will give you a printout of all data you have saved on the customer. So if your computer system fails you can use this card as a backup.

The last part of the page is like a form to fill in by hand.

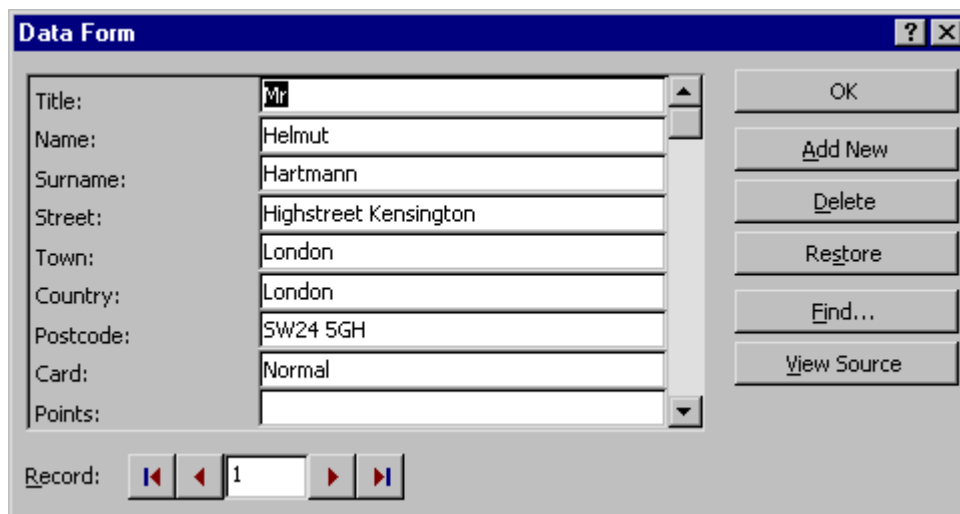
MM Enter Points:

This document does not contain any fields and is not meant to be mail merged. The text the user will see is the text I will show to you in a second. It is a guide how to get in to the data source and edit it easily.

[This document is for the purpose of adding the reward points to the account of the people.¶
To open the data source please press the following keys one after another:¶
Alt - T - R - Tab - Tab - Enter - Enter.¶
This brings you the database window. There you find the customer by keying in the customer number you can find on the button of the Customer card into the little field called Record. Pressing Enter will bring you directly to the record of the customer and you can enter the number of points he has at the moment.¶

This should enable the staff to get into the source and edit the records effectively.

They will then see this screen:



The screenshot shows a 'Data Form' window with a blue title bar. The form contains several text input fields and a set of buttons. The fields are labeled as follows:

Title:	Mr
Name:	Helmut
Surname:	Hartmann
Street:	Highstreet Kensington
Town:	London
Country:	London
Postcode:	SW24 5GH
Card:	Normal
Points:	



At the bottom of the form, there is a 'Record:' label followed by a small input field containing the number '1' and navigation buttons (back, forward, first, last). To the right of the form, there is a vertical stack of buttons: OK, Add New, Delete, Restore, Find..., and View Source.

Changes I made during implementation:

I added the possibility to enter points into the database and the system will support both the former way, manually, or the new way, automatically. The old way might be a little bit simpler for the staff but it will not look as professional to the customers. It will be a hassle to make sure that you have enough forms available and to find out who could become a gold card member for free the coming year. It will be more efficient to look through the records and see a pure number. If it is big enough you will let the customer be a gold member. You could run a query as well.

To make the above easier I had to have a customer number which is equal to the record number so that it is easy to find the customer and mend his point level when he buys and article. Therefore I added the merge field “Merge Record #” to the customer number.

But this also means that if a customer wants to quit the system you can not simply delete his record. This would change the customer number of all customers behind him. Therefore the user will enter “deleted” in the title field I added a query option to all documents. The syntax it “Tile <> deleted”. Only one document is an exception and that is “MM Sell Address”. This will enable the user to earn some money even after the customer quit.

While doing my user guide I discovered the following two buttons:  and . Therefore I could do without the file called MM Enter Points.

It is now easy enough for the user to open any document and by using these buttons get into the data source to alter it. Also it is easy to explain how to get to the query options now.

Teaching session:

These instructions are for a very experienced user, who will have to train to user types.

1. The owner of the shop.
2. The employees of the shop owner.

Shop-owner-training-session:

I will just list what he should be able to do after the session:

1. Open document
2. Save document
3. Save as...
4. He has to understand the difference between Save and Save as...

5. Has to know what to do if asked whether he wants to save. Sometimes it would be better if he did not.
6. Has to understand the question “Database connected to the document. Do you want to save changes to the data in the database?”
7. Understand how mail merging works.
8. How to create a new mail merge document.
9. How to change data in the data source.
10. Insert merge fields.
11. Use if statements
12. Use Queries.
13. Use Fill-in boxes.
14. Be able to use the above while creating a new document.
15. Use printing properly.

The user guide is there to aid the trainer to teach the user. It is also there for the user to remind him how to do things. It is not complete enough to enable the user to use the system. Only after the session with the trainer it will be good enough.

Training the employees:

The employees will only have to do the following:

1. Has to know what mail merge is.
2. Has to be able to open a document.
3. Has to be able to go into the database.
4. Has to be able to find certain customer.
5. Has to be able to change the reward points.
6. Has to be able to close document.
7. Has to be able to know when to save.
8. Has to be able to understand “database is connected to the file. Do you want to save changes in the database?”

Has to be able to quit the program and terminate the computer.

Testing and Evaluation:

After setting the system up you have to make sure that it is working as well. Most of the tasks will have been tested during implementation though so it is very hard to get every detail again later.

Testing plan:

I will look at each document and test the different by setting up example records and seeing whether the output is correct afterwards. I will sometimes assume it something works one time and I set it up the same way later it will always work. (e.g. MM Header uses an if then else statement on graphics and colour writing. If statements like this are used later again. I will not prove every time that they work.

I will have a table with the following columns: Thing to test, how to test and outcome. I might need another column for comments.

Testing:

I will test the documents in the same order as they are described in the implementation.

MM Header:

Thing to test	How to test	Outcome
If statement fitting in the logo. (card=normal, , "logo") and colourful test Hela.	Example data: 3 records, one gold, one normal gold and one normal.	Two with logo one without like it is supposed to.
Merge title, name, address...	Check with the above data records.	The address has to be the same as in the data source (not mixed up or confused). No problem.
Dear Mr Flos,	Check with the above data records.	The right person is addressed in the right letter.

MM Birthday:

Thing to test	How to test	Outcome
If card <> normal then "wine + present" else "present".	Example data: three records: one gold, one normal, one normal gold.	Gold and normal gold "wine + present" and normal "present".

MM Information:

Thing to test	How to test	Outcome
Fill in box	Is the user asked to fill the box and is it made clear what to enter?	The user is asked for the input and the question should be clear enough.
If card \diamond normal then "picture" else "".	Run with example data like test on logo.	Picture is shown for both gold card members but not for normal card member.
Instructions to insert picture.	Pretend to be the user and play a bit dumb. Do you get the same output?	Yes.

MM Invitation:

Thing to test	How to test	Outcome
Fill in box	Is the user asked to fill the box and is it made clear what to enter?	The user is asked for the input and the question should be clear enough.
If statements to reserve right number of places.	Example data: records: each family status for gold, normal gold and normal card members.	Check whether each person gets the right invitation.

MM New Year:

Thing to test	How to test	Outcome
Is all data correct and not mixed up.	Print out example data from last test.	Comparing data source and printout I can find no difference.
Does the output come on one page (more than one address)	Have enough records and print.	More than one record on each page.

MM Welcome:

Thing to test	How to test	Outcome
If sentence requesting £30 from normal gold card users.	Example data: 3 records, one gold, one normal gold and one gold.	The sentence should only appear in one of the letters.
Does it say "gold" on a gold card members card?	Same as previous.	"Gold" has to show on both, the gold and the normal gold.
Does the card hold all personal details (name and #)	Use the example data above.	Comparing data source and card, ok.
Do all personal details come up next to the card?	Print report and compare data source and printout.	All details are correct.
Is the output of the rest of the document correct?	Print out.	Yes.

Evaluation:

Evaluation criteria:

QUALITATIVE:

1. Most commonly used documents included. These have to be build up so that the user can change them to fit his needs better.

I included all necessary documents the user wanted me to. He will be able to change them after receiving his training session and read his user guide. He can also use Word help and any book he can buy in shops.

2. User friendly.

The system is reasonably user friendly. Mail merging itself is not very easy and therefore it is hard to create a much more user friendly system. This is one of the few points I would try to improve if I had more time and if I had to do the system again.

The user was satisfied and thought he can manage to deal with the problems (Appendix, questionnaire).

3. User can extend the system with aid of the help-documentation.

The combination of the teaching session, my user guide and the Word documentation should enable the user to extend the system.

4. The customer thinks he is being treated very personally.

I dropped enough personal hints that the user will think the shop owner addressed him personally. E.g. he will not know that the computer figured out how many reserved seats he wants.

5. The customers have to be satisfied.

Sadly I had no time to launch a questionnaire to ask the customers this question. But it must have seemed reasonable to them as every 3rd or 4th person shopping there in the first two weeks took a leaflet.

QUANTITATIVE:

1. Minimum of Hardware and Software required because of financial problems of the shop.

If the user decides to use the manual system and probably otherwise as well he only needs one computer. He will need one printer and I would advice him to buy a ink printer. Then he can print black and white relatively cheap and he can achieve good results printing

colour. He will only need Word as software.

At the system is not that complex you can use a 486 DX4 100 or something similar. These computers are amazingly cheap at the moment.

2. System has to be capable of holding data of 2500 customers.

The database used in word is capable of holding many more records than that.

3. The printer has to be able to print colour and black and white.

As I said in 1., a ink printer will be solving the task perfectly.

4. The scheme must increase the number of customers so that the growth rate after one year exceeds the normal one by more than 2%.

I could not test this jet either, as a year is not over yet.

Tasks and subtasks:

Task 1: Create standard documents.

I created several standard documents.

Subt. a: Have to make the customer think the document was written only to him and no one else.

The documents contain personal comments making the user think it was written only for him.

Subt. b: Print cheap for lower level of card. Can be more costly for gold card.

I only included colour print and print of pictures for Gold card members.

Subt. c: The customers' member-card so that it only has to be laminated.

Included in "MM Welcome".

Subt. d: Happy birthday: Automatically printed a few days before the birthday to be sent out by post.

The user can automatically select the people whose birthdays are due to soon.

Subt. e: Invitation: The customers will be invited to events like social diners or to a catwalk show.

Different events for certain kind of card only, reservation for certain customers, meal or drinks for gold card only... Special comments for different card status.

The user can introduce the different events using the fill in box. If the event is only for gold card users he will be able to use a query. If he wants to add special comments for different card users he can do this in the if then else statement chain.

Subt. f: Notification of gold card level. If not because of annual turnover then bill £30.
This is a half-manual process (user has to select people) but the letter goes out automatically. Bill is written automatically.

Subt. g: Welcome new applicant.

“MM Welcomë

Subt. h: Information on new products: Depending on card including picture of product (costs of colour print).

“MM Information”

Subt. i: Document to forward information of customers to other companies.

“MM Sell Address”

Task 2: Create database of names and addresses, birthday, sex.

Subt. a: Do not use a different software-package if possible.

Only used Word.

Subt. b: Must hold: ...

Holds all necessary data.

Subt. c: Easy to enter new or to alter data of existing records.

Not too hard and loads of help available.

Task 3: Create a form collecting all the data necessary and still having an attracting visual apparel.

Have designed such a thing.

Subt. a: Must get all data necessary for the database.

Gets all data necessary.

Subt. b: Must be designed to attract customers.

It's nice and colourful.

Subt. c: Must explain what the purpose of the scheme is.

Does more or less. There will be posters up, but I did not have any influence on that.

Subt. d: Not too costly to print.

Print made cheaper by using ink jet and by only using colourful graphics for gold card users. The user refused to go to a printing office as this would steal his independence.

Task 4: Create a user interface:

Subt. a: Add a second way to start Word with the user interface loaded.

I have a second icon for Word Reward Card on the desktop. This will start word with my special toolbar loaded.

Subt. b: Customise toolbars so that the user only has the buttons he needs.

I have a customised toolbar.

Task 5: Create teaching plan.

Subt. a: Create the plan of teaching session which, combined with the user guide, will enable the user to create new documents.

Done.

Subt. b: Enable experienced user (used all mentioned tools... before) to give the teaching session.

Done.

Task 6: Create user guide enabling the user to create new documents.

Subt. a: Clear description how to use existing documents and the database.

Done.

Subt. b: How to extend the database.

Done.

Subt. c: How to enter new or alter customers.

Done.

Subt. d: How to create new documents.

Done.

Subt. e: How to use tools like if-statements and queries.

Done. Word help needed as well.

Subt. f: Where to get help form.

Done.

Task 7: Create a teaching session that will cover all aspects necessary to enable the user to use the program effectively.

Done.

Overall:

I think I created quite a nice program. It meets all the evaluation criteria as far as I can see at this state and it fulfils all the tasks. It could be a little bit user-friendlier, but the user pushed me for time as he wanted to introduce the system this New Year.

What I would do differently? I think I would not invest so much into doing the actual documents but I would try to teach the user as much as possible so that he can do it himself. I would probably implement the thing with him next to me and explain everything to him. Never the less I am satisfied with what I produced and so is the user (see questionnaire in Appendix).