|  |  |
| --- | --- |
|  | Pay-and-Tax |
|  | Version 2017-18 release 5  By  Virendra Bajpai  Commandant  10 Battalion BSF  🖃[virendra.bajpai@gmail.com](mailto:virendra.bajpai@gmail.com) (preferred mode)  🕿+919953295030, +919485145921 (in case of emergency) |

|  |
| --- |
| Pay-and-Tax |
| A system that helps in disbursement of pay and allowances to personnel of the unit with speed, accuracy and transparency, while saving the manpower utilised as pay-clerks. Its tax subsystem allows to deduct TDS with accuracy implementing tax rules. |

Contents

[Preface 5](#_Toc487280490)

[Acknowledgements 7](#_Toc487280491)

[Introduction 8](#_Toc487280492)

[Brief Functioning of the System 13](#_Toc487280493)

[User’s Manual 16](#_Toc487280494)

[Using the Package 16](#_Toc487280495)

[Objects in the Database 18](#_Toc487280496)

[Relationships 19](#_Toc487280497)

[Independent Tables 20](#_Toc487280498)

[Permanent Tables 20](#_Toc487280499)

[Semi-Permanent Tables 26](#_Toc487280500)

[Temporary Tables 27](#_Toc487280501)

[Masters Level 1 28](#_Toc487280502)

[Ranks 28](#_Toc487280503)

[Subunits 28](#_Toc487280504)

[Units 29](#_Toc487280505)

[Masters Level 2 30](#_Toc487280506)

[Amount Types Pay 30](#_Toc487280507)

[Amount Types Tax 32](#_Toc487280508)

[Funds 34](#_Toc487280509)

[Messes 34](#_Toc487280510)

[Personnel 34](#_Toc487280511)

[Details 45](#_Toc487280512)

[Bills Sent 45](#_Toc487280513)

[Data 47](#_Toc487280514)

[Funding 47](#_Toc487280515)

[Messing 48](#_Toc487280516)

[Pay 49](#_Toc487280517)

[Recoveries 51](#_Toc487280518)

[Tax Data 52](#_Toc487280519)

[Unit Disbursements 52](#_Toc487280520)

[Other Forms 53](#_Toc487280521)

[Accounts 53](#_Toc487280522)

[CA Changed 54](#_Toc487280523)

[DDO Certificates 55](#_Toc487280524)

[RHA Changed 56](#_Toc487280525)

[Separate Emoluments 57](#_Toc487280526)

[Start-up 58](#_Toc487280527)

[Automation Forms 59](#_Toc487280528)

[Bills Received 59](#_Toc487280529)

[Export Trickle File 59](#_Toc487280530)

[Imports 60](#_Toc487280531)

[Import Personnel 63](#_Toc487280532)

[Import Process 64](#_Toc487280533)

[Initialize 64](#_Toc487280534)

[Initialize Tax 65](#_Toc487280535)

[Save Reports to PDF 66](#_Toc487280536)

[Selective Calculation Sheet 67](#_Toc487280537)

[Selective Form 16 67](#_Toc487280538)

[Selective Tax Calculate 67](#_Toc487280539)

[Special Update 68](#_Toc487280540)

[Tax Wizard 69](#_Toc487280541)

[Reports 73](#_Toc487280542)

[Annual Tax 73](#_Toc487280543)

[Bank Letter 74](#_Toc487280544)

[Bank Summary 75](#_Toc487280545)

[Bill Details 76](#_Toc487280546)

[Bill Summary 77](#_Toc487280547)

[Bills Sent 78](#_Toc487280548)

[Calculation Sheet 79](#_Toc487280549)

[DDO Certificates 80](#_Toc487280550)

[DDO Certificates Register 80](#_Toc487280551)

[DDO Intimation to Bank 81](#_Toc487280552)

[Documents Pending 82](#_Toc487280553)

[Form 16 83](#_Toc487280554)

[Messing Recovery (Coy) 84](#_Toc487280555)

[Missing Pay 84](#_Toc487280556)

[Negative Cases 84](#_Toc487280557)

[Out Unit Emails 85](#_Toc487280558)

[Out Unit Letters 85](#_Toc487280559)

[Out Unit Summary 87](#_Toc487280560)

[Payment History 88](#_Toc487280561)

[Pensioner Labels 89](#_Toc487280562)

[Pensioner Letters 90](#_Toc487280563)

[Pensioner Summary 91](#_Toc487280564)

[Recovery List 92](#_Toc487280565)

[Recovery List Separated 92](#_Toc487280566)

[Tax Return \* (Q1-Q4) 93](#_Toc487280567)

[Tax Deduction 93](#_Toc487280568)

[UD List 94](#_Toc487280569)

[Unit Bills 94](#_Toc487280570)

[Unit Bills (Coy) 95](#_Toc487280571)

[Unit Fund Summary 96](#_Toc487280572)

[Unit Pay Summary 96](#_Toc487280573)

[Unit Pay Summary (Coy) 97](#_Toc487280574)

[External Extensions 98](#_Toc487280575)

[Accounts: Password Protected Access Database 98](#_Toc487280576)

[Excel Templates 102](#_Toc487280577)

[Messing 102](#_Toc487280578)

[Pay 102](#_Toc487280579)

[Personnel 102](#_Toc487280580)

[Tax Data 102](#_Toc487280581)

[Summary of the Process 103](#_Toc487280582)

[Onetime Process 103](#_Toc487280583)

[New Users 103](#_Toc487280584)

[Users of older split-database subsystems 104](#_Toc487280585)

[Main Process 104](#_Toc487280586)

[General Guideline 105](#_Toc487280587)

[Revision History 107](#_Toc487280588)

[Version 2017-18 107](#_Toc487280589)

[release 1 107](#_Toc487280590)

[release 2 107](#_Toc487280591)

[release 3 107](#_Toc487280592)

[release 4 107](#_Toc487280593)

[release 5 107](#_Toc487280594)

[A Word from Developer 108](#_Toc487280595)

# Preface

The database application for Disbursements and Tax Deduction at Source was designed to help Accountants and Cashiers of various drawing units, whether small or large, to swiftly, accurately and transparently disburse amounts, due to various individuals of the unit, while recovering and depositing TDS. All applications need to accompany with some documentation for its users, so that they can use the application and reap the benefits of the automation.

1. New users are recommended to read the entire ‘**User’s Manual**’ section of the documentation, however, for the ease of older users ‘**Revision History**’ section has been created towards the end of the documentation where they would be able to quickly spot the changes done since the version they were using. For more details, such older users may like to refer to respective sections that have changed. For older users, while this reduces the requirement of going through the entire documentation, each time a revision or new version is launched, it also enables them to quickly adapt to the changes or enhancements and use the release towards their benefits.
2. The disbursement subsystem was mostly devised out of the discussions between me and my onetime cashier, then **ASI (Min) R S Bhadauriya**, by now, whom I must pass on the credit in guiding me to make it audit-friendly, without whose involvement thorough testing and implementation, it would not have been possible for me as part-time project. ‘Acknowledgements’ section would be used to pass credit for your valuable suggestions in the subsystems merged.
3. Having ventured into making disbursement part, which helped Cashiers in quickly performing the task of disbursement of money to individuals, my attention was sought by **Sh Sanjeev Kumar**, then Second in Command 109 Bn BSF, during my visit to FHQ in the month of Nov-Dec 2013, towards the responsibility of DDOs in respect of Tax Deduction at Source. Income Tax being a ghost topic for me, when I took it during my tenure as software developer, I was slightly reluctant initially, but when I received a letter for penalty from Income Tax Department for late filing of return, I could realise the concern of the fellow officer and, therefore, decided to work on the automation of TDS.
4. The Tax subsystem was initially designed to be used as an out of process monthly TDS calculator for disbursement subsystem allowing DDOs/Accountants to include individual TDS amounts before pay disbursement. It further facilitated preparing ‘Quarterly Income Tax Returns’ for filing and finally printing of ‘Form 16’ for the individuals. The process is required to be run once a month and it took around half an hour to complete on an old dual core laptop that I used to develop it. I had to fall back on revision of table design due to serious performance issues; therefore, you may find it to be faulty in table design especially in terms of stored calculated values. It is as shabby as the Tax System can be.
5. The documentation is intended to provide complete details of functionality and use of the subsystem. While it remains mostly silent on database development part (I treat it beyond the scope of user manual), yet the system hides nothing from developers to keep building on it and making it more useful by augmenting with new functionalities as per their needs.
6. Once data is entered in a version of the application, I assure you that it will be transported to newer releases. However, keeping it backward compatible through many versions may not be possible without much complicating the system. Therefore, I would recommend you keep an eye on <https://github.com/vbaj/ngo-pay-and-tax> so that you are updated with the latest version soon after its release.
7. Feedbacks from the user are important and welcome. **I would recommend you attach a zipped copy of the database with data and maximum possible details you can give in case an error occurs during the process, including screenshots of error followed by screenshot of debug screen.** Suggestions towards additional functionalities are also welcome in which I would request you to give **full details** of the functionality required and suggested design of the reports.
8. Before I go any further on the software manual part, **I must warn you** that in this age of digital data, whatever you do to hide your intentional or unintentional mistakes, those are quite traceable and surely leave your tracks that can be sniffed. **My collaborator** had to face disciplinary action on corruption charges because he thought to be smarter than the actual smartness of the traces, when it came to more than what you control. Don’t do it even if you can read and understand code left open intentionally for developers.

# Acknowledgements

* Control over Account number change was created due to insistence of my collaborator, **R S Bahadauriya**, however, I still feel that the process can be simplified and left to auditors to poke their nose in defaults.
* **Sh Nirman Singh Aujla**, then Commandant 68 Bn BSF for giving valuable advice on DDO Certificates for Bank Loan to individuals and then exercise control over change of account numbers for which undertaking is given by the DDO.
  + Not being used by majority of users
    - Format may be lacking
    - Banks are not providing required data if a loan is sanctioned
    - Individuals are also not interested in providing the details
  + Intended to be dropped in future
* **Sh Sanjeev Kumar**, then Second in Command 109 Bn BSF to wake me up for the responsibilities of DDOs towards TDS that together with fine imposed by Income Tax Department forced me to take on hitherto dreaded GHOST of income tax system that I avoided learning.
* *Waiting for your suggestion*

# Introduction

1. A computer based system needs to have a defined set of ‘Business Rules’ for its design. A change in the governing business rule demands change in the system also that is used to implement it. Obviously, the higher is the frequency of changes in business rules, the higher is the frequency of requirement of the system change implementing it. Though, certain changes in the rules can be foreseen or predicted and may be handled with a careful system design, yet rules can never be forced to conform to the system design. This asks for an endless pursuit for a computer based system to conform to the governing rules whenever they are changed, obviously, without consulting developers of the system how to cope up with. Thus, a centralised application is always preferred by developers whenever the governing rules are frequently changing. Such centralised systems reduce the load on designers of the system to create update patches and to guarantee identical applications running all over the user base always; a scenario best voted for web applications. However, performance, system load, connectivity and security issues have their trade-offs against centrally maintained web applications.
2. Historically, the Pay subsystem in BSF had been central in its nature owing to the underlying modification requirements, which are more frequent than any other system in BSF, in addition to the central availability of the funds at PAD for disbursement. NGO Tax system is relatively new as it took its flight after Sixth CPC when wages were raised without corresponding increase in the exemption limits. It was also coupled with introduction of National Pension Scheme, which contributed to no more than 10% contribution towards savings.
3. In past few years, BSF has seen migration of the system of pay distribution from cash to electronic transactions, wherein the manual system has also gone through corresponding changes. However, the age-old system of pay distribution in BSF, based on ‘Pay-Clerks’, continued with little modifications as far as the paying units were concerned. I believe that it was thought at central level to disburse the amount directly, as they had pursued for account numbers of all NGOs. However, there are constraints and pitfalls in this approach. First, if the disbursement is central, the central DDO must fulfil another connected responsibility of calculating, deducting and depositing tax and then ultimately issuing form 16. It becomes close to unmanageable with the NGO strength of about 2.5 lakh. A highly cumbersome and time taking process. Second, in a remotely deployed battalion, where nearby ATM may become somewhat luxury, essential mess recovery could become problematic and its functioning may get crippled due to lack of funds.
4. In the historical system, sub-unit wise nominal rolls, A-rolls and detachment bills were handed over to pay-clerks, who brought individual messing details with them along with blank nominal rolls in wide sheets. These pay-clerks then filled up salary and messing details against names listed in their nominal roll. Having completed the nominal roll till this point, a CASHWALA used to take out one bill at a time and used to announce regimental numbers, names, etc., and corresponding amounts from bill in front of these pay-clerks, who in turn used to notify and note down the amounts called related to personnel against their names in their pay roll pro-forma. Finally, these amounts were totalled and any recoveries like messing, etc. were subtracted to find out net amount payable to each personnel. The amount was paid in cash.
5. When CVC guidelines were implemented during 2005-06 or so, the transaction of cash slowly stopped and amounts started getting transferred trough banks clearing systems. However, the inherent process continued to remain unchanged except providing data to banks for their automation as it was also the period when units started getting their first lot of glorified typewriters (computers). When some computer trained personnel started getting recruited, some of them started using Microsoft Excel for calculations based on HLOOKUP and VLOOKUP. This eased out much of their work, however, such system could not expand much due to connections and accessibility issues at the lower level.
6. In other units, once the amounts for CBS and cash transactions, etc. were determined, one CASHWALA would enter the CBS part of transaction to the pay roll pro-forma in MS Excel matching account numbers of individuals in the ‘**SBI Trickle Feed**’ system. Thanks to the MS Excel that it can still open legacy version of data contained in ‘**FoxPro**’ ‘.dbf’ files. The bank account number and CBS amounts were then copied and pasted in a copy of prior list prepared in MS Excel for final matching of account numbers to ensure that correct recipient would receive the amount. Of course, even more difficult methods were available for use. Once these account numbers are matched manually with existing lists, by deleting or adding account numbers of various individuals, the data was reverse copied and pasted from Excel to the ‘FoxPro’ dbf file opened in MS Excel. A ‘FoxPro’ application, named ‘Trickle Feed’, that was provided by SBI to generate a (flat) file for their automation is then used to create the desired file, which is provided to bank in a CD or a pen-drive using which the transactions were done in CBS at SBI. This was the situation prevailing in Jammu and Kashmir during 2010; however, banks in many other states had started asking the same data in MS Excel format.
7. As a variation in the process, a few units have adapted to draw specific amount of cash payment to individuals equal to or more than messing. Thus, eradicating the requirement of bank accounts for messes and messing recovery relied upon manual payment by individuals in accordance with the old system or to be exact remaining amount went to individuals having deducted messing. This method also reduces necessary visit to ATM by every individual and felt suitable for battalions especially deployed in remote areas. The pitfall of this system is that if you draw cash in the name of someone, who cannot receive it, you need to deposit it as UD, and, if one goes on leave beyond this period you would end up piling up mess dues.
8. All this process in a battalion is found to consume over a week for 10 pay-clerks each month, during which the precious manpower remains unavailable for its destined use. While analysing the process from the point of view of automation, it was felt that in the manual system, segregation of data used to consume 60% and totalling job used to consume nearly 30% time and efforts of everyone, while remaining 10% job was dedicated towards creation of data file for SBI in ‘Trickle Feed’ system. The totalling of amounts was done row-wise, column-wise for page to finally arrive at final list totals. One of the reasons for this grouping was discovered to be the fact that combined totalling and segregation of amounts was not humanely possible below nine pay-clerks in a battalion (especially amounts segregated on messes running in the unit). Unfortunately, these totals often did not match due to human errors, forcing the process of rechecking of amounts noted and then cumbersome retotalling. I could visualise that calling of pay-clerks for pay disbursement was off-shoot of legacy required for this delegation of efforts.
9. While these ‘Pay-Clerks’ were used to be called for pay on request of unit Cahier, universal crunch of manpower lured ASI/Adjt to utilise them in other duties every now and then. Cash branch used to suffer in terms of pendency of amounts that could not be included in the manual process due to absence of some of the pay-clerks. I started to conceive the system when my cashier then ASI/Min R S Bhadauriya requested me, if I could spare some time to develop a system through which he may be able to minimise the involvement of these pay-clerks. I was at that time unaware of the system developed in Excel and I had never used the formulae enumerated above and my choice went to Access as I had worked with it and it was more capable in the given scenario. I believe that Excel is great analytical tool and quick clean tool to play around with data, however, it lacks the capabilities of a RDBMS. Though, his expectations were rudimentary, he braved to do away with the pay-clerks even with the initial show-case version of the system that I showed him for the first time for his feedback. The first ever pay, without pay-clerks in 182 Bn BSF was, thus, finalised using the initial version during Nov 2010. His enthusiasm towards the system motivated me further and a system, which started with the aim to help cashier branch to minimise involvement of manpower for segregation and totalling, culminated in a pay-clerk-less, accurate, transparent, immediate payment system to individuals.
10. In its present shape, ‘Disbursement’ subsystem in ‘Pay-and-Tax’ is an attempt to cut on this manpower through minor automation done through a stand-alone system developed in MS Access 2010. While the system is designed to achieve accuracy, speed and transparency, can also assist Accounts Branch in preparing bills to be sent to PAD. If this feature is used, a bill can be included instantly in payments to personnel without loss of time as soon as payment for it is received.
11. There are following types of amounts involved in the process of disbursement in general:
    1. One that is payable to an individual (emolument)
    2. One that is recoverable from an individual and is retained in Govt. Fund account as undisbursed amount (recovery/deduction)
    3. One that is recoverable from an individual and is credited against another account/fund (messing/RMA)
    4. One that is recoverable from an individual and is credited in Govt. Fund account of another unit (payments recovered for another unit)
    5. One that is required to be transferred to another fund account of the unit like loan or welfare fund, etc.
    6. One that is required to be transferred to Govt. Fund account of another unit for reasons like return of erroneous credit, etc.
12. While the system handles all above types of transactions, it handles following categories of personnel to make it a complete package:
    1. Personnel who are directly remitted and payment details are printed in groups for local distribution to subunits. This category is intended for personnel posted in the unit under various subunits.
    2. Personnel who are directly remitted and payment details are intimated through letters to individuals. The category is intended for pensioners.
    3. Personnel who are remitted through another unit and payment details are intimated to concerned units through letters/emails for further disbursement to individuals. The category is intended for personnel posted out to or attached with other units.
13. The intended features in the database did not warrant features beyond MS Access 2003 to keep it backward compatible, yet MS Access 2010 did not allow saving the database in older formats. It was felt unwise to uninstall licensed MS Office 2010 Pro Plus just to maintain backward compatibility having known the fact that MS Office 2010 has been installed in all the computers supplied by M/S NIIT under Intranet Prahari Project. Thus, the system remained designed in Microsoft Access 2010 without giving much consideration to backward compatibility issues. Despite using the file format compatible to MS Access 2007, I was tempted to use certain features added in Access 2010. It was later found that the application became incompatible with Office 2007. Thus, the outcome remained limiting its use strictly with MS Office 2010 and above. Though, I have moved on to Office 2016 (Office 365), I am trying hard to keep it compatible to Office 2010. I am also aware of the fact that licensed Office 2013 is also available in BSF, however, of late I have started thinking about piracy and going open source.
14. The basic aim of the system is to achieve desired automation, thus, security as a matter of assumed trust on the part of handlers and further engineering by enthusiasts, has not been catered for while developing the system. I lay the responsibility of security on the user by protecting the database, if required, by creating database passwords. Nothing in the database is concealed beyond the level of hiding of database objects to reduce the clutter of links in the ‘Navigation Pane’. The code view is also left open for user level modifications by developer geeks, if required.
15. I have been working with MS Access since 1999 to be exact, starting with Access 97, and could develop a few applications that lasted a decade and were only superseded to be part of IPP or when people urged to delegate data entry. Yet, during the development of the system I encountered with a few strange problems for which I could not find solution even on MSDN or Microsoft Forums. The strange problems that I encountered are listed below:
    1. The combo-boxes in the system failed in their default ‘Auto Expand’ features at their will, without control of the developer. Even fully typed correct value in the list could not match values existing in the list. Every time the feature failed, the user was forced to select one from the list that needed lot of scrolling. This erratic behaviour (developers would agree) forced me to restrict long list combo-boxes, ‘Regt No’ in this case, on various forms not limiting to list and forced me to use a workaround of not leaving the ‘Regt No’ field and sound an audible alert for user if one is not found in the list. I feel sorry, I could not give you a proper work around as the aim of the system and benefits you may reap from it did not warrant any delay in its launch, while fully debugged class as a proper work around may take some time … popping up a selection form was considered cumbersome by me from data-entry operator point of view. After all, units can send full 9 figures correct for pay changes to the central pay system, which are rejected by the central pay system in case of incorrectness. Thus, my demand to its correctness to 9 figures looked genuine to me, while I provided ‘sorting’ and ‘filtering’ for searching of personnel on all forms on the table independent of toolbar. Frankly, while rewriting the combined documentation I realised the issue is still pending.
    2. A few reports, especially ‘Bank Letter’ report used to crash the system when it was opened without data. I tested all the queries utilised by the report including the embedded query without data. I was surprised to note that all the queries worked fine without crashing the system, yet the report used to crash every time beyond my control. I was, therefore, forced for a workaround of not allowing user to open reports at all (other than in design mode) if data is not available.
16. These were the limitations, but I hope you would bear with me. I believe, many would not have even noticed it adversely had I not confessed here.
17. Historically, the Pay and allied systems in BSF had been central in its nature owing to the underlying modification requirements, which are more frequent than any other system in BSF. Prior to 6th Pay Commission, very small group of NGOs, if any, used to fall in the taxable zone and, therefore, the system of TDS was largely ignored for NGOs. After the hike in salary in 6th Pay Commission without corresponding hike in income tax slabs, slowly personnel from all strata started reaching taxable income. Switchover to National Pension Scheme was another culprit in this aspect limiting deductions to 10% and forcing individuals to make choices in savings. But we don’t generally learn and adapt quickly, thus, department did not think of providing means to DDOs to handle the situation. Resulting in criminal wastage of manpower and efforts at all levels. I was apprised that, in my unit, one man remained engaged throughout last quarter of the financial year for this job in addition to performing some sundry tasks in accounts branch.
18. The tax subsystem of the database is aimed towards providing DDOs a tool that will equip them in deducting income tax, depositing against individual PAN, promptly filing quarterly returns and issuing form 16 to individuals.

# Brief Functioning of the System

1. The system leverages on data import capability of MS Access from other sources and works with two copies of database, one each for Accounts and Cash branches. While Accounts branch may keep on sending bills through ‘Bills Sent’ interface in its copy, Cash branch may continue to enter data in respect of ‘Messing’, direct fund transfer or in respect of any recovery from individual in its copy. Initially, Cash branch may need to enter backlog bills which have been passed and not entered in the copy of database in use by Accounts branch. As and when the processing of pay is completed by IT Wing, Cash branch is required to download an additional report named ‘SALARY FOR IMPORT’, which is available along with other reports of pay processing. It may be noted that this report has been specifically designed by IT Wing for this purpose and is required to be downloaded in ‘MS Excel’ format instead of regular PDF. Any time before the transaction, this downloaded data can be imported into the system using ‘Import Salary’ form, while bills can be imported from the copy of database used by Accounts branch using ‘Import Bills’ form. From the payment vouchers received from PAD, Cash branch may now include received bills into the process by check marking each received bill on ‘Bills Received’ form. In case it is the process that includes ‘Salary’ you would be required to run ‘Tax Wizard’ form and follow the sequence till the finish. And yes! You have done it! Now print the reports, create bank file using ‘Export Trickle File’, export monthly or quarterly reports, write it on CD or pen-drive for bank, TAXWALA and CA, that’s it. The process just explained is an ideal situation when Accounts and Cash branches are working hand in hand. Further elaboration of the process is done in subsequent paragraphs.
2. The Pay subsystem starts with the capturing of data from the process of sending various bills to PAD for payments by Accounts branch. The sanction order is partially automated for print, to reduce my efforts in report designing, while also reducing the complexity of system. This entry at the time of its sending removes delay in payments to individual on their receipt. The system provides copying of data into the process of pay disbursement in respect of any bill so entered by just selecting the bill by click of a button on ‘Bills Received’ form for which payment has been received; thus, the system automatically includes it for payment in the on-going process. Undoing effect of erroneous selection is also catered for through deleting selective erroneous data from the process by the click of the same button on the form. In case amount in any of the bill is retrenched, the modification to this effect can either be done on ‘Bills Sent’ form before selecting the bill for inclusion in process of payment or it can be modified on ‘Pay’ form after its inclusion in the process.
3. Direct data entry on ‘Pay’ form is also allowed to include amounts received from other units or for bills which are not included in the database earlier. This leaves the choice on the user unit whether it wants to utilise the bill sending procedure, partially automated, or wants to leave just backlog bills for entry during pay disbursement and starts sending bills now on through the system, or wants to show even more concern towards early payments by entering backlog of bills already sent to PAD for payment.
4. Any amount included in the process can be marked as ‘UD’ amount. The provision for ‘UD’ is given to not only arrive at bill wise correct totals in ‘Bill Summary’, but also to provide a list of ‘UD’ amounts in the end through a report named ‘UD List’. These ‘UD’ amounts are retained in the database for next process so that whenever these amounts are required to be released may be just unchecked to be included in the process. Recoveries other than ‘Messing’ and ‘RMA’ are listed in ‘Recovery List’.
5. Messing and RMA recoveries are entered through ‘Messing’ form. During discussion with AO auditing 139 Bn BSF at Teliamura in the middle of Dec 2013, it was advised to credit all recoveries for Messing and RMA in Ration Fund from where it could be credited/disbursed to Coy Messes, so that only Ration Fund Cash Book is required during audit of Govt Fund. To account for those cases (withheld) where no amounts are paid to individual, all being ‘UD’, these entries can be marked as ‘Not Recovered’.
6. The records so created for individuals, as mentioned above, need only entering of regimental numbers and corresponding amount and no further for each record.
7. The amounts that are required to be recovered and then disbursed towards another unit are entered on ‘Pay (Other Unit Recovery)’ form.
8. Since GPF bills for individuals are separately made for everyone, ‘Pay (One Amount Type)’ form allows you to enter Bill No along with the record itself.
9. In case some amount is required to be transferred to any of the private funds of the unit other than ‘Ration Fund’, during the transaction, corresponding entries can be included in ‘Funding’ form.
10. In case some amount is required to be transferred to Govt. fund account of another unit, corresponding entries can be made in ‘Unit Disbursements’ form.
11. Having completed all the entries for bills, recoveries, etc. ‘Negative Cases’ report may be opened to check for any case where total recovery from an individual exceeds total payments. If there is no case of such negative payments, reports namely ‘Messing Recovery (Coy)’, ‘Unit Pay Summary (Coy)’ and ‘Unit Bills (Coy)’ can be printed for distribution to subunits for information of personnel and messes running. ‘Ration Recovery Details’ report may also be given to various messes if required, however, list of RMA recoveries also form part of this report, thus, selective pages may be given to unit Ration NCO for record. ‘Unit Pay Summary’ and ‘Unit Bills’ are designed to be kept as record at Cash Branch instead of synonymous reports with ‘(Coy)’ appended in their names, which are meant for subunits.
12. Letters in respect of payments to pensioners can be printed using ‘Pensioner Letters’, while letters to units of posted out/attached with personnel can be printed using ‘Out Unit Letters’ for despatch or ‘Out Unit Emails’ can be used to save the report in PDF format for attachment in email. Labels for addresses of pensioners may be printed through ‘Pensioner Labels’ report to make the job easier for dispatcher. ‘Pensioner Summary’ and ‘Out Unit Summary’ may also be printed and kept for record instead of letters.
13. Finally, ‘Bank Letter’ and ‘Bank Summary’ can be printed for signing by DDO for bank and record respectively. I recommend that you print just single copy of each report and keep ‘Bank Summary’ for record. The data in the format required by SBI is exported using ‘Export Trickle File’ form that can be given to bank on a CD or any other media like pen-drive.
14. Since system date is printed on most of the reports, it is recommended that reports are saved as PDF using ‘Save Reports to PDF’ form, so that these reports can be printed later.
15. This far is the general process of one disbursement after which you need to ‘Initialize’ the system. The brief of this process in terms of steps has been explained under the heading ‘Summary of the Process’.
16. Since the system depends on correct entry of regimental numbers in each record and any other criteria to match to correct individual would ask for additional data entry making system more cumbersome, the key to success of the system is in the correctness of regimental numbers to full 9 figures.
17. As the list of personnel is available in the database itself and a computer based system can generate totals on the fly, flawlessly for everyone and can arrange them in any desired groupings, it is possible to manage the entire pay system of a unit on one computer using one or at most two personnel, one each for calling an entry and another for data entry. During trial, it took lesser time of two personnel than the same for entire team of eleven personnel in the older setup in my battalion despite gross errors in regimental numbers listed in the bills and messing. The time reduced further to about 2-3 days (office hours only) for one person once the correctness of regimental numbers was achieved in the lists. This correctness can be easily enforced and achieved within a unit. After all, units could send several pay changes to IT Wing with all 9 figures in regimental numbers correct. With the possibility of use of import of payments and recoveries using Excel files, the time of process can be further reduced depending on the use of these sheets.
18. The Tax subsystem also leverages on data import/export capability of MS Access from/to other sources. As and when the processing of pay is completed by IT Wing, unit is required to download an additional report named ‘NGO’s Annual Income Tax Sheet’, which is available along with other reports of pay processing. It may be noted that this report has been specifically designed by IT Wing for this purpose and is required to be downloaded in CSV (Comma Separated Values) format instead of regular PDF or Excel format.
19. The process involves providing of data regarding individual’s annual savings, taxable payments through bills and any deductions from salary or allowances as in case of imprisonment or long leaves, etc. Since, in the process beyond this point, most of the work is done through background processes, I have created a form named ‘Tax Wizard’. This form in turn presents you with step by step process using ‘Update PAN’, ‘Import Payments’, ‘Import Recoveries’, ‘Import Tax Sheet CSV’, ‘Calculate Tax’ and ‘Export Tax’ sub-forms. By the end of wizard, you would be able to include monthly tax deduction for everyone. Also, you would be able to export a file for filing return or print actual/tentative Form 16 for individual.

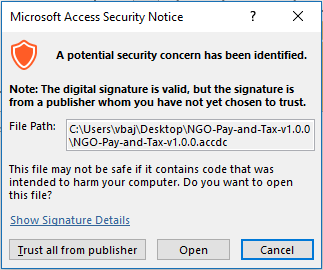
# User’s Manual

## Using the Package

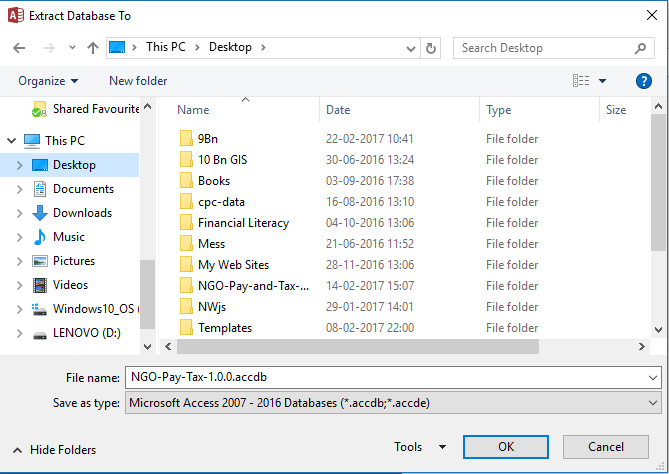
1. When you open the signed package of the database, you may be presented with ‘Insert Smart Card’ dialog box as under.



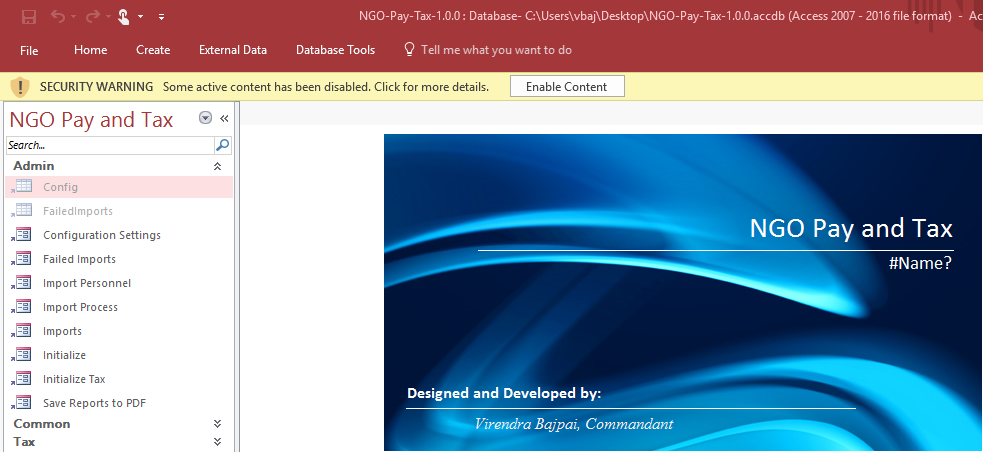
1. Click ‘Cancel’ in the dialog above to proceed to ‘Security Notice’ dialog as shown below.



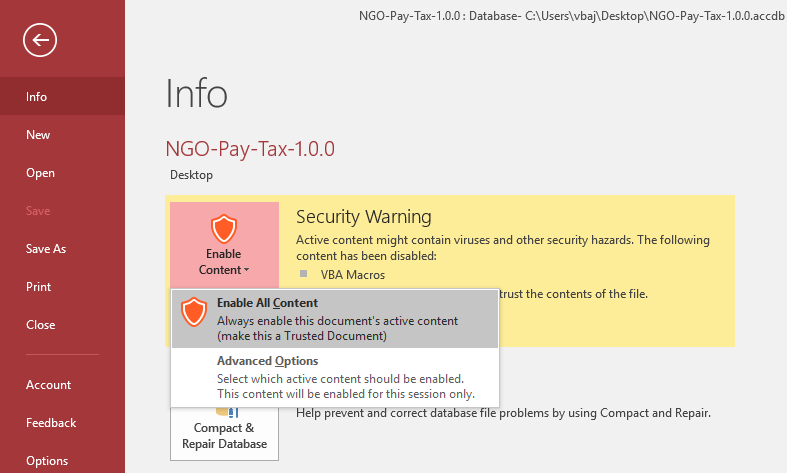
1. Choose ‘Open’ on the dialog above to proceed to the ‘Extract Database To’ dialog as under.



1. Now choose a location and file name to save the database and click ‘OK’. Having specified the location and name of the database you would be presented with following situation each time you create new database from the package.



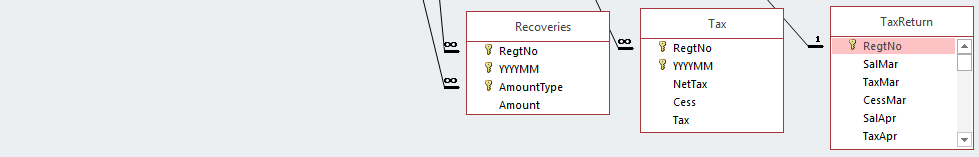
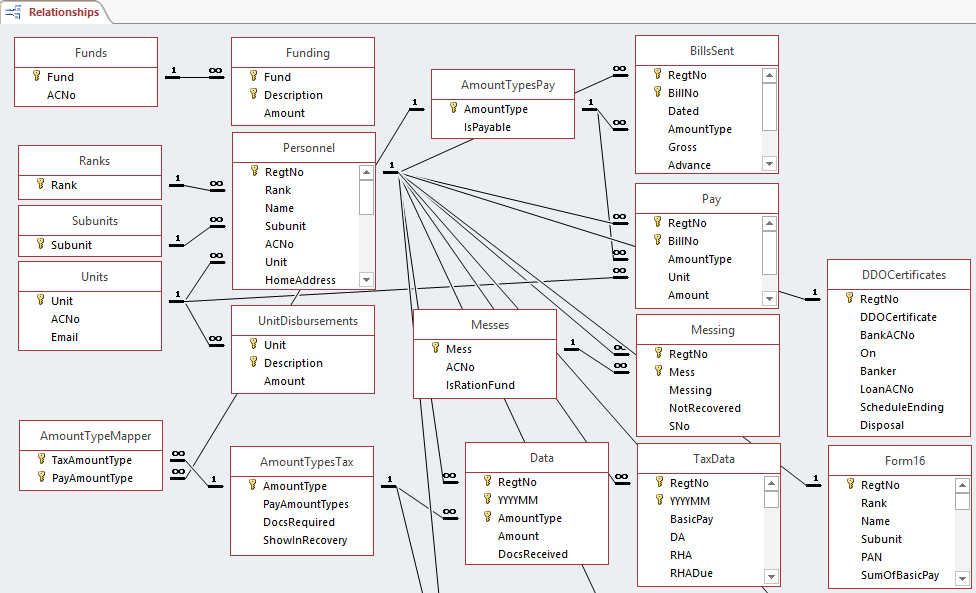
1. In Microsoft Office 2007 onwards, VBA (Visual Basic for Applications) code is by default restricted for security reasons. Since, the database relies on VBA extensively; you would now be required to enable the contents blocked due to VBA code. To do this, switch to ‘File’ tab on top-left of the window and you would be able to easily find ‘Enable Content’ button where you would require clicking on ‘Enable All Contents’ as shown below to start using the database. Alternatively, if ‘Enable Content’ button is visible, clicking the button there itself, which could be the case on larger display screens.



## Objects in the Database

1. 28 tables have been used in the system to hold data required for the process. None of these tables are intended for direct use.
2. 53 queries have been created on these tables for presentation of processed data to user. You should not modify or rename any of these queries as these are used in one or the other report, either directly or indirectly, and their modification may jeopardise the entire system.
3. 60 forms have been created to facilitate the user handle the processes involved.
4. 45 reports have been designed to help user keep record of the process as well as for intimation of payees.
5. 6 code modules and 10 class modules have also been used in the system for common reusable code.
6. This much listing of Access Objects, I felt necessary to provide somewhere in the documentation. If the list is confusing you, I would suggest you forget about it and go-on with the documentation.

## Relationships



1. The figure above depicts relationships between various tables used in the system. You need to have a closer look on the connecting lines among these tables. Tables that are not listed here are ‘Independent’ tables as referred to in the documentation. The tables connected with lines are related with each other having ‘Parent’ table on left and ‘Child’ table on the right. All these relationships are ‘Enforced’ and ‘Cascade Update’. If you do not understand relationships between database tables, you need not bother about it. You just need to look at the connected columns in the figure above and need to know that unless an entry exists in the column of the table shown on left of the relationship, it cannot exist in the connected column in the table at right. For example, you cannot enter ‘Alfa Company’ as a value in ‘Personnel’ table unless you have it entered earlier in ‘Subunits’ table. This is known as ‘Enforced’ relationship.
2. Sorry, I used long names for ‘Subunits’ above and you happen to enter data in ‘Personnel’ table for too many personnel, which lead to lot of space and ink wastage on reports; you wish now to have it as ‘A Coy’ instead of its longer counterpart. No problem, open ‘Subunits’ and change ‘Alfa Company’ to ‘A Coy’, related records in ‘Personnel’ table will be automatically changed correspondingly. This is what is called ‘Cascade Update’ in database relationships.
3. This is true in case of all connected tables in the figure. You can freely modify values in the left table(s) without bothering for existing records in the right table(s). You must note, however, that you cannot change a value to an already existing value in the list, e.g., ‘C Coy’ to ‘A Coy’ if ‘A Coy’ is already entered, as it will create a duplicate record in the table at left, which is not allowed.
4. ‘Cascade Delete’ is another phenomenon in database relationships, which is not used in this database. Since ‘Cascade Delete’ is not used, database will not allow you to delete an entry in the left table(s) unless all corresponding records related to the entry in the connected right table(s) have been deleted or changed to other existing value. Had the relationship been ‘Cascade Delete’, the records in the right table(s) would also have been deleted with the records in left table. This feature was not invoked to avoid accidental loss of data, which could be quite time consuming while trying to recreate it in case the deletion is done mistakenly.
5. This is all about the relationships, simplified, between the tables you need to know as a user. The purposefully arranged figure above and this much description was felt necessary to give you an insight towards why a specific sequence of entry is required in the system. To streamline user’s data-entry sequence, these tables and forms are arranged in following categories or sequence:
   1. Independent Tables,
   2. Masters Level 1,
   3. Masters Level 2,
   4. Details and
   5. Other Forms
6. You may note as a reminder that a value cannot be entered in table(s) listed under ‘Masters Level 2’ unless it is entered in connected table(s) listed under ‘Masters Level 1’. Similarly, data in table(s) mentioned under ‘Details’ can only be entered once the hierarchy of ‘Masters Level(s) 1 and 2’ has been completed. Let me have certain exemptions in saying that to simplify the language spoken by the screenshot image of relationship diagram.
7. Names of forms used in the system are either near synonymous to their underlying table names on which they operate and/or to the purpose they are designed for. In the succeeding paragraphs, these forms are described one by one. Words ‘Table’ and ‘Form’ have been used interchangeably in the description; however, the word ‘Interface’ has been used exclusively for ‘Form’.

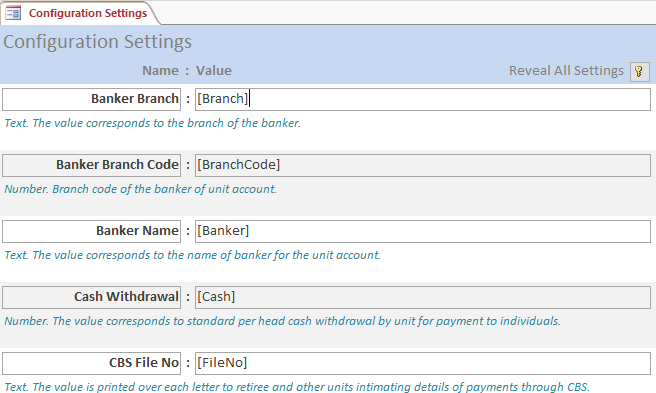
## Independent Tables

1. The tables described in this group are not related with any other table in the database. These tables are either maintained independently, or their data is created through a process to survive across sessions, or their data is stored for temporary purposes.

### Permanent Tables

1. The data in these tables is of permanent nature and persists across processes.

#### Config (table): Configuration Settings (form)



1. To make the system usable beyond one unit, the unit for which it was designed and tested, without asking for changes required in the system design, the database uses a few user defined values that are stored in a table named ‘Config’ handled by ‘Configuration Settings’ form for system customisation. Brief regarding the intended setting is also stored in the table for help to users while looking at data in the form. Having corrected these values suitable to your unit, you should also change the value of variable ‘Setup Done’ to ‘Yes’ so that the ‘Configuration Settings’ form is not automatically opened every time the database is opened. The form can be opened manually to modify data at any time later, if required.
2. Modifying variable names, their deletion or addition has been restricted through the form as the purpose of these values is limited to the value only and accidental deletion or name modification may jeopardise the system. Adding a variable also has no effect unless it is used somewhere in the code, thus, is not catered for through the form.
3. A few variables are not intended to be modified by the users directly. Such variables are hidden on the form using a ‘Hidden’ field (not shown on the form) checked against those variables. These variables are automatically saved during the process to help sustaining a value used in the system. However, ‘Reveal all Settings’ button with key icon on top right of the form to view or modify hidden settings in case it is needed.
4. I feel that the interface is straightforward. You can only modify the values on this form. At present following values are required to be defined by the user. These variables are discussed one by one here.

##### Banker Branch

1. The value is printed in addressing of ‘Bank Letter/Summary’ as well as at the bottom of letters as part of information regarding own banker.

##### Banker Branch Code

1. The value is printed at the bottom of letters as part of information regarding own banker.

##### Banker Name

1. The value is printed in addressing of ‘Bank Letter/Summary’ as well as at the bottom of letters as part of information regarding own banker.

##### Cash Withdrawal

1. If a unit is drawing fixed amount of cash for its personnel, to avoid their mandatory trips to ATMs due to distances involved or due to any other considerations, the fixed amount drawn for each individual may be entered here. If this amount exceeds ‘0’, records of ‘Cash Withdrawal’ are automatically created during the process of ‘Import Salary’ for all personnel for whom ‘Salary’ is drawn. You need to have a list of personnel prepared for whom this amount is not required to be drawn for deleting selective records of ‘Cash Withdrawal’ from ‘Pay’ table. An alternative to this is discussed later under the heading ‘Special Update’ under ‘Other Forms’ for fixed values and another under ‘Import Cash Withdrawal’ for subunit maintained lists.

##### CBS File No

1. This value is printed as a part of letter number in letters generated in the reports. The entry may be modified as per file number in use by user unit for dispatch of such letters of intimation. However, at present separate file numbers are not catered for different reports. Thus, if a unit is using separate files for different categories, user may change this value to appropriate file number before opening corresponding report. In such cases, ‘Save Reports to PDF’ form becomes redundant and you can double click the report to save it as PDF.

##### Contra Account

1. If your bank is asking data in Excel format you need not understand this value and you may safely proceed to next heading.
2. Old users might have encountered with SBI’s ‘Trickle Feed’ system. If you open the final output file created by it in Notepad or any other plain text editor, you would find it to have several lines like the one shown below. Each line in this file, starting with ‘01’, corresponds to each bank account number for which you would have entered amounts in the system with few items added automatically through their programme.



1. First 2 digits correspond to a code indicating whether the amount is credit or debit. Code ‘01’ corresponds to a credit while code ‘00’ corresponds to debit (never used in our case). Code ‘54’ corresponds to debit in the contra account of the bank, which is available only on the last line of the file. The further structure of this line corresponds to each line in the file.
2. Next 17 digits (zeroes padded at left) represent bank’s account number of individuals involved in the transaction. On the last line, this is replaced by the bank’s ‘Contra Account Number’, which can be found at the last line on the Trickle Feed file, where the line starts with code ‘54’ as shown above. It includes 4-digit branch-code (6896 in this case) followed by a figure 3 at the end (who knows the reason of using 3?).
3. Next 16 figures (exceeding thousands of crores of ₹) are equivalent to amounts credited to in the accounts mentioned on the line; however, last 2 zeroes represent paisa in each amount, which must be ‘00’ as no transaction up to the level of paisa is done during disbursements. On the last line, starting with code ‘54’ the amount is replaced with the total amount of transaction involved.
4. Last 66 characters are descriptive in nature and are replicated in all records. Making the ‘Flat File’ contain total 101 characters in each record including trailing spaces in a ‘SBI Trickle Feed’ file.
5. While describing the file structure of ‘Trickle Feed’, I intended to explain what you are required to enter in this value in the ‘Configuration Settings’. The text required to be entered is boxed in the figure shown above from the file you are creating through command ‘TF’ (batch file), i.e., Trickle Feed system of SBI. Leading zeroes are programmatically added during data export thus you need not put these in the value. Though, extra leading zeroes would not hamper the operation, you must take care while noting or copying this value to contain correct figures.

##### DDO Certificate Header

1. The value is printed on top of the DDO’s Certificate issued for a loan from bank. For the sake of aesthetics, I did not use multiline textbox, however, line breaks can be inserted using semicolons (;) in the field.

##### DDO Certificate Life (Hidden)

1. The positive numeric value that corresponds to number of months set for expiry of DDO certificate issued. It is printed on the certificate issued and then the expired records are dropped during initialisation. The value is set to give clear 6 months’ expiry as default.

##### DDO Name

1. The name of present DDO of the unit is stored here and is printed below the signature space on various reports intended to be signed by DDO.

##### Email

1. The email address of the unit that is printed as contact information in the header part of the letters.

##### Employer

1. The value of employer is printed on Form 16 and it needs to be exactly what it is written in the registration of TAN of unit.

##### Financial Year (Hidden)

1. This was required to make peace with the state of Income Tax Rules in vogue. This is also used to calculate the Assessment Year value for ‘Form 16’ report. The value needs to be in the form yyyy-yyyy like 2013-2014. This is also used to populate ‘Year Months’ table required to project last factual record over following remaining months to reach an approximation for entire year. Since it is created through ‘Initialize Tax’ interface, where user has control to change first YYYY part, it is normally kept hidden from user to avoid any accidental modification.

##### Opened Once (Hidden)

1. The Yes/No type of value is used to indicate to the system that the database has been opened once. It is used to decide whether to open ‘Initialize’ form or open ‘Configuration Settings’ form. It is by default set to ‘No’, however, once the ‘Start-up’ form is opened, it checks for its value. If the value is ‘No’, the value is changed to ‘Yes’ and ‘Initialize’ form is presented. Nest time onwards, ‘Start-up’ form will find the value to be ‘Yes’ and then it will check the value of ‘Setup Done’, which if ‘Yes’ it will not do anything, otherwise it will prompt you to complete configuration presenting ‘Configuration Settings’ form.

##### Processed For (Hidden)

1. A numeric value that represents the month number, for which last tax calculation was done. This was required to avoid data corruption for previous processes. Once you have processed the data for say the month of Jun (value 6), you would not be able to recalculate the tax for the month of May (value 5). I felt that by then you would have already deducted the tax for May and its accidental change during the next process may become erroneous. However, you can reprocess Jun as many times as you wish until you process for Jul (value 7).

##### RHA Rebate

1. The system of clubbed allowances under Risk and Hardship Allowances was found to be a difficult preposition. Since codes are not available at tax database end and it would be precisely futile if I tried and recreated the logic by which the value of RHA was arrived at in central system, I thought it prudent to provide the user a means through which he can specify total monthly rebate allowed on allowances included in RHA. This value would eventually be multiplied by 12 or number of RHA values while giving annual rebate. If your unit does not have a value in RHA, this value would be ignored or may be given 0 as value.

##### SDA/CA Rebate

1. On the similar grounds, as to RHA, it was difficult to segregate various CA values for exemption, therefore, the value is asked from user and rebate given. If your unit does not have a value in SDA/CA, this value would be ignored or may be given 0 as value.

##### Separate Emolument

1. While designing a report with dynamically changing columns is to some extent is possible through certain workarounds, but designing a report with dynamically changing number of columns cannot been made possible unless the report is created entirely through code. Though it was possible through code, was difficult and did not worth the efforts as such a report would easily and soon exceed the width of paper while wasting lot of white space. This forced me to club certain amounts in the summary reports, namely ‘Unit Pay Summary’ and ‘Unit Pay Summary (Coy). The clubbed amounts were then printed in separate reports, namely ‘Unit Bills’ and ‘Unit Bills (Coy)’.
2. During the workshop conducted for Pay subsystem at IT Lab of Frontier HQ, Jammu on 6th and 7th Jan 2011, it was proposed to have a user defined column printed separately on these summary reports, which is excluded from other reports designed to give details of clubbed bills and recoveries. This was earlier done in case of ‘Detachment’. Since the proposed change was considered highly valuable in terms of stationary saving it was incorporated in the system.
3. The value of this variable should invariably be defined by the user to suite his need in terms of longest bill included in the process. The value must exactly match the ‘Amount Type’ mentioned in the ‘Pay’ table. You must note that the amount so separated is checked for being an emolument, thus, you would not be able to separate a recovery this way. Further, the column heading must also be mentioned against the variable ‘Separate Emolument Title’ described next.

##### Separate Emolument Title

1. As described above, the system provides facility of separating one of the ‘Amount Type’ from bills to save stationary as it is not printed in detailed report. While this material-saving-feature allows freedom to user, it also rests responsibility on the user to depict enough details in the header of the column for information of the payee. The text given for this variable is printed as column title of the separated emolument. You are requested to keep it short enough to adjust within the width of column as wrapped text will not be visible because expanding the textbox vertically has not been allowed for aesthetics.

##### Setup Done

1. Unless the value is ‘Yes’ (case insensitive), the system interprets that you have not finished editing the values in ‘Configuration Settings’ table as per your requirements, thus, this form is opened each time (second time onwards) you open the database to prompt you to fill in the values. Once you have changed these values as per your requirements, change this value to ‘Yes’ to avoid system bothering you each time you open the database. However, ‘Configuration Settings’ form can be opened anytime later if you wish to change a value.

##### Telephone

1. The value is printed in the header part of the letters as contact information.

##### Unit

1. The value of ‘Unit’ is printed below signature space in various reports where signatures are expected to be appended.

##### Unit Account Number

1. The value is printed in contents part of ‘Bank Letter/Summary’ as well as at the bottom of letters as part of information regarding own banker.

##### Unit Address

1. The value is printed on top of letters generated in reports. For aesthetics, I did not provide line breaks in the values on this form; however, the effect of line break can be introduced by inserting a semicolon (;) in the address, preferably, with no spaces around. For example, if you enter ‘HQ X Bn BSF;Name of Road;Name of Distt (State);Pin-123456’ it will be printed as under:

HQ X Bn BSF

Name of Road

Name of Distt (State)

Pin-123456

##### Unit TAN

1. The value is printed on ‘Form 16’ report.

##### Version (Hidden)

1. The value is shown during the start-up of the system. Though, it is not in any use at present other than showing it on start-up, but it may be used inside code to keep it compatible with data import from older versions. Thus, changing this value may jeopardise the system at some later stage.

##### Year Month (Hidden)

1. It stores the ‘Year Month’ value in ‘YYYYMM’ format and is used in various reports to show the Mon-Year values in the title.

#### Payment History (table)

1. It was felt necessary to keep a record of payments and recoveries affected through ‘Pay’ table to a user for a user-defined period, say on annual basis to satisfy queries of individuals regarding payments/recoveries at a later stage. The table is, therefore, added to keep complete date wise record of payments and recoveries individually till the previous process. While ‘Payment History’ report is saved as PDF during ‘Save Reports as PDF’ process, it is once again saved as PDF when the data is cleared during ‘Initialize Tax’ form. This makes the copy of report exported as the main backup of payments during the period following the last clear action before this, i.e., for the period of taxation from Mar to Feb.

#### Tax (table)

1. The data in this table is populated during execution of ‘Calculate Tax’ form under ‘Tax Wizard’. This table holds actual monthly TDS data. I felt that this data did not warrant any manual changes, therefore, no interface was designed for this table. During the process of recalculation of tax, data for the given month is first removed and then repopulated. This is the reason why it is not allowed to recalculated tax for previous months.

#### Year Months (table)

1. The table is populated during ‘Initialize Tax’ and is used to project last pay record over entire remaining period to reach an approximated gross salary for calculation of tax. Since, the data is auto generated and is not intended to be modified, the table does not have corresponding interface.

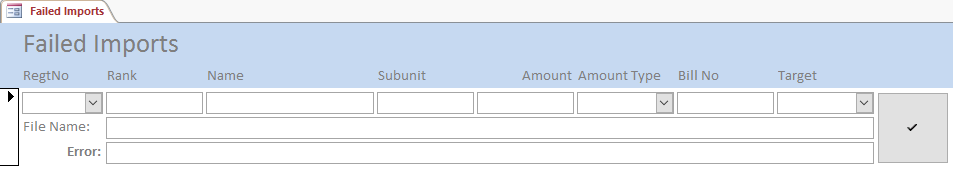
### Semi-Permanent Tables

1. The tables described here are of semi-permanent nature and data in the table persists only in the process or data in the table is removed first during a process and then is recreated by the process.

#### Amount Type Mapper (table)

1. The table is populated during execution of ‘Initialize’ form. This is designed to direct the system to fetch certain payments in Pay subsystem made through ‘Pay’ interfaces. See ‘Amount Types \*’ tables for [more details](#_Amount_Types_Pay). Since the data is auto generated and is not intended to be modified, the table does not have a corresponding interface.

#### Failed Imports

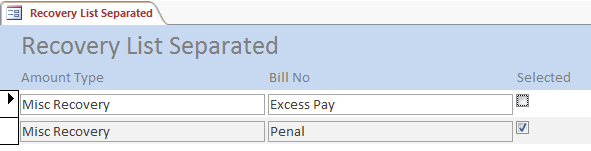


1. The database was designed to enable user to take an Excel file from template as input on ‘Imports’ form and target the data towards ‘Bills Sent’, ‘Pay’ or ‘Messing’ tables, however, it was felt that in certain situations, records could fail during import in the process due to wrong ‘Regt No’, ‘Amount Type’ or ‘Mess’ etc. This table holds such records for the users to correct them and then include them in the process by clicking the check button in front of the record. If the record is uploaded successfully, it is removed from the table.

#### Form16 (table)

1. This table is populated during the execution of ‘Calculate Tax’ form under ‘Tax Wizard’. The data in this table is used in ‘Form 16’ and ‘Calculation Sheet’ reports. Since the data is auto generated and is not intended to be modified, the table does not have a corresponding interface.

#### Recovery List Separated



1. This table is designed to be used in conjunction with synonymous report. It is used towards the end of process and its data is not ported across the process. Having completed the data entry in the ‘Pay’ table, this table presents you with a list of all ‘Amount Type’, ‘Bill No’ entries in ‘Pay’ table that are marked on recovery side. Bills that are selected using the check-mark, are included in the ‘Recovery List Separated’ report. You are advised to complete this selection before actual printing of the report.

#### Tax Return (table)

1. During development of the Tax subsystem, the system encountered serious performance issues at Tax return related reports. Therefore, data related to reports was stored in the table, which is repopulated during execution of ‘Calculate Tax’ form under ‘Tax Wizard’. The table is created to address performance issues and to quickly query data for quarterly return filing. Since, the data is auto generated and is not intended to be modified, the table does not have corresponding interface.

### Temporary Tables

1. The tables described here are of temporary nature and data in the table does not persist beyond the connected form operation.

#### Accounts (Table)

1. It is a kind of temporary table, data in which is populated and deleted through code by the ‘Accounts’ form. The description of its usage is deferred till ‘Other Forms’ so that basics of the system are clear to the user before its details.

#### Special Update (Table)

1. To facilitate faster batch update similar amounts on a group of individuals, this table temporarily holds list of ‘Personnel’ for their selection through check-mark using ‘Special Update’ form. The description of the form on the table has been deferred till ‘Other Forms’ to allow the user to first know about the targeted tables handled by the form.

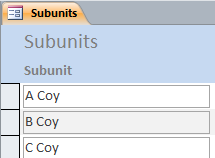
## Masters Level 1

### Ranks



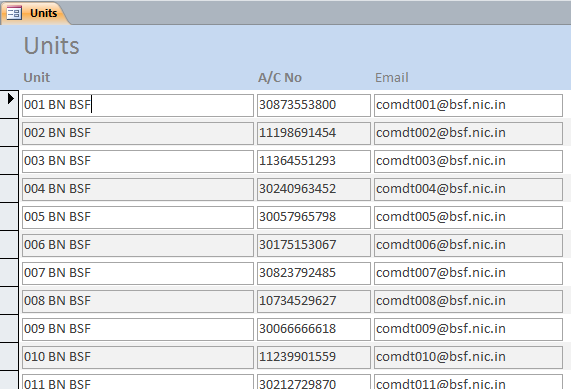
1. The list of ranks in use in the unit have been stored in the ‘Ranks’ table. The rank is not used for any purpose other than as a title of an individual.
2. To include pensioners in the system, ranks like ‘Sh’, ‘Smt’, etc., may also be included. This would allow ‘Regt No’ to continue to hold the original identity of the record, while title can change as per the custom. It is suggested that having changed the ‘Rank’ to ‘Sh’, the name may be appended with ‘(Ex RANK)’ for a pensioner. Further, if a dependent of the pensioner becomes the recipient, ‘Rank’ may be appropriately changed, name can be changed to dependents’ name and retiree’s name preceded by relationship of dependent (W/O or the likes) may appropriately be added with ‘Late’ as per the custom which should find part in ‘Address’ column.

### Subunits



1. The subunits of the user unit can be stored in the table named ‘Subunits’.
2. As in the case of a battalion, its companies are deployed geographically apart, and it was felt necessary to intimate all its personnel regarding what all they have received and what has been recovered, therefore, a few reports have been designed in the system to be distributed to these subunits. In cases of vast span of deployment, this data finds its importance, whereas smaller and compact units like headquarters may like to ignore this table. However, I would suggest you group all personnel in one group, in case of small and compact unit, as the subunits are checked against incomplete records at the end of ‘Imports’ when ‘Salaries’ is selected for import and leaving it blank would mix it up with other incomplete details.

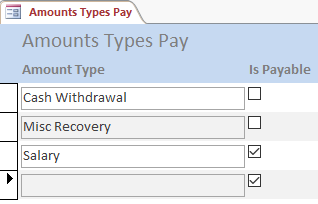
### Units



1. The table is created to hold unit names, their official bank account numbers and email addresses so that amounts received in respect of posted out/attached personnel using ‘Pay’ table or any other transfer of fund using ‘Unit Disbursements’ can be credited in the accounts of concerned units followed by letters/emails of intimation.
2. To avoid maintaining up-to-date addresses of units in Cash branch, which is not its basic job, I have left the responsibility of correct dispatch address on dispatcher of the unit and, thus, have not catered for addresses of units in the table.

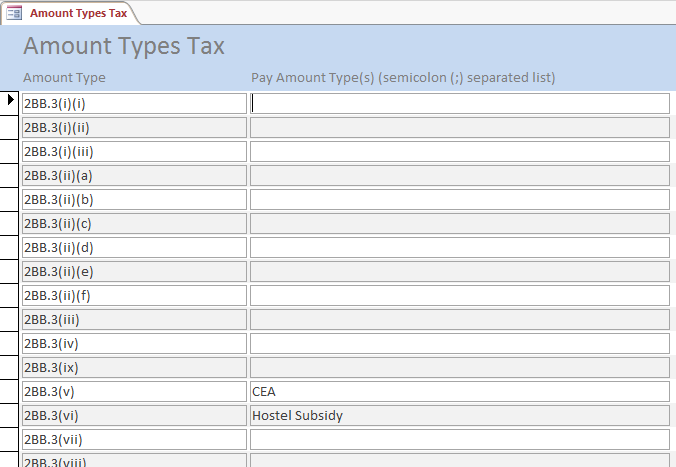
## Masters Level 2

### Amount Types Pay



1. ‘Amount Types Pay’ table is one of the most important table in the Pay subsystem, as most of the system design revolves around it. The table has only two columns namely, ‘Amount Type’ as text and ‘Is Payable’ as check mark. ‘Amount Type’ column is created to hold descriptive name of an amount entity involved in the system like ‘Salary’, ‘Detachment’, ‘Cash Withdrawal’, ‘GPF’, ‘LTC’ to name a few. The ‘Is Payable’ column is created to indicate to the system regarding the sign of an amount during calculations, i.e., if the field is checked the amount is added (+) towards calculated amount for payment to individual, otherwise it is subtracted (-) from amount payable to individual. While creating a record, it is checked (+) by default, but care must be taken to uncheck it if the amount is a kind of deduction from the individual. This type of consideration was necessary to handle ‘Cash Withdrawal’, a system being followed by a few units and it can be easily extended to other kinds of recoveries if required.
2. The list is only illustrative and user may add many more values including the payments pertaining to pensioners or even private funds. However, since the debit account of unit is given to the bank while presenting the ‘cheque’ and only one account can be given per transaction, care must be taken to segregate funds while starting a process. For example, you may have entered a few private fund disbursement values for transaction and you happen to include ‘MRC’ bill in the process, the results would be what you would not want. Please don’t blame me for this; I have already warned you, isn’t it? However, I would suggest you make as many separate copies of the database as required, one for each fund.
3. A few ‘Amount Type’ values have special considerations in the system that need to be explained here.
   1. ‘**Cash Withdrawal**’ is considered as a kind of recovery during CBS by the system and is retained in the Govt. fund account. This is withdrawn in cash by presenting a separate cheque. Records for this amount for each individual, included in salary, are automatically created during ‘Imports’ when ‘Salaries’ is selected from the value stored against the variable ‘Cash Withdrawal’ in ‘Configuration Settings’ given that the value was greater than ‘0’ (zero). The process of ‘Imports’ with ‘Salaries’ has been described later. A unit may need to have the list of personnel prepared for whom this amount is NOT required to be drawn during the transaction. Records of this ‘Cash Withdrawal’ for personnel in the list prepared may be selectively deleted from the ‘Pay’ table. The amount has been given a separate place in reports destined to be given to sub-units. I must make it clear here (I had a long argument with my cashier on this) that to do any job for a list, the list needs to be an ‘Action List’, i.e., a job each is done for each item in the list. If you want to perform a job on items that are not included in the list, you must first create the ‘Action List’. Thus, deleting records from ‘Pay’ table corresponding to personnel for whom ‘Cash’ is not drawn, using a list of personnel for whom ‘Cash’ is drawn, would be extremely difficult. Alas! You cannot mark a record on ‘Pay’ form on computer screen; the system does not cater for it. However, ‘0’ as a value can be placed against ‘Cash Withdrawal’ variable in ‘Configuration Settings’ to avoid creation of automatic entry during the import process and then list of ‘Cash Withdrawal’ cases can be manually created on ‘Pay’ form or may be imported using ‘Imports’ form with ‘Cach Withdrawals’ selected or as an alternative, records of fixed value ‘Cash Withdrawal’ may be entered through ‘Special Update’ form, which is described later.
   2. ‘**Salary**’ is another important component that also finds separate place in the reports designed for subunits. Since, records for ‘Salary’ are automatically created in ‘Pay’ table; this record would also be automatically created during ‘Imports’ with ‘Salaries’ process if it is not found.
   3. ‘**Misc Recovery**’ is the component used as the main head for any recovery and is automatically created during ‘Imports’ with ‘Salaries’, if not found.
4. Except amount types mentioned above and a user defined emolument, all other payable amounts are clubbed under ‘Bills’, while all recoveries except ‘Cash Withdrawal’ and ‘Messing’ (described later) are clubbed under ‘Recovery’ column in ‘Unit Pay Summary’ and ‘Unit Pay Summary (Coy)’. Other types can be modified, added or deleted as per requirement of the user.
5. To keep the list short, I would suggest users to use ‘Bill No’ field to include additional description if required especially in case of ‘Arrears’. For example, if a bill number 333 pertains to Arrear of ‘Pay’, the bill number in the system can be entered as ‘333 (Pay Arr)’.

### Amount Types Tax



1. The table has two columns shown, namely, ‘Amount Type’ and ‘Pay Amount Types’ both contain text data. ‘Amount Type’ column is created to hold descriptive name of an amount entity involved in the Tax subsystem, hardcoded in the ‘Tax Calculator’ class module. The ‘Pay Amount Types’ column is created to direct the system, as to which all ‘Amount Types’ in Pay subsystem are needed to be imported into the Tax subsystem during Tax processing and be mapped to which ‘Amount Type’ in Tax subsystem. **In case multiple amount types in pay are required to be mapped to single amount type in tax, you may separate them using semicolon (;).** To simplify querying, ‘Amount Type Mapper’ table is repopulated during the process.
2. As the ‘Amount Types’ are hardcoded in the system, adding any other type will not have any effect. Therefore, addition of records has not provided on the form. Similarly, deletion or modification of type will cause loss of effect; therefore, this facility is also restricted. The only modification you are allowed is to modify contents of semicolon separated ‘Pay Amount Types’ column.
3. There are two more columns in the table that are hidden on the interface, namely, ‘Docs Required’ and ‘Show in Recovery’. These need some explanation that is as under:

#### Docs Required

1. This check-mark at ‘Amount Types Tax’ table (pre-decided in system design) along with ‘Docs Received’ check-mark on ‘Data’ form determines which all records are there in ‘Data’ for which individuals have not submitted required documents and, thus, such records are shown on ‘Documents Pending’ report made subunit-wise.
2. It is suggested that undertakings from individuals be taken regarding their plan for savings and other tax exemptions in the beginning of the financial year and while incorporating them on ‘Data’ form, leave ‘Docs Received’ as unchecked. This check-mark can be placed as and when document with respect to the planned saving is received. However, ‘Documents Pending’ report may be sent to subunits 3-4 months before closure of tax year (closing on Feb) as a warning that records for which documents are not received by a ceiling date (should be specified say 31st Dec) will be deleted and tax will be deducted accordingly. This would ensure smoother hike in TDS and may help in avoiding negative pay situations.

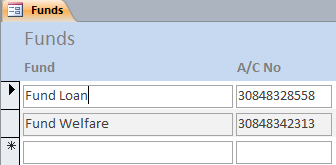
#### Show in Recovery

1. A table quite like ‘Data’ with the name ‘Recoveries’ has been created to include corresponding recoveries. However, many ‘Amount Types’ in Tax subsystem are not required in this table. Thus, the list is restricted using the combo-box instead of any other method. You must also note that ‘Pay Amount Types’ are only imported to Data table, hence, recoveries must not be mentioned in ‘Amount Types Tax’ table in that column.

#### Handling of various Amount Types

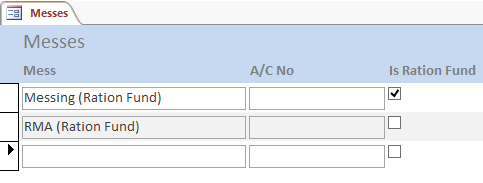
1. **2BB\***: These correspond to compensatory allowances. Corresponding rebates are calculated on 12 months flat if a value is found in ‘Data’ table. However, ‘CEA’ and ‘Hostel Subsidy’ are also part of this rule respectively 2BB.3 (v) and (vi).
2. **Donation\***: These correspond to donations for which 100% and 50% exemption is allowed.
3. **Edu Loan Interest**: Full amount is taken, therefore, use only allowable value here.
4. **Entertainment Allowance**: Limited to ₹5000 or 20% of Annual Basic, whichever is less.
5. **HBA\***: Rebate of ₹150000 is allowed for HBA(i) and limited within ₹150000 for HBA(p) in GPF group.
6. **HRA**: No rebate is allowed if ‘Own House’ is checked in ‘Personnel’ or ‘Rent Paid’ is not available in ‘Data’ or ‘Recoveries’. If given, it is governed by ‘HRA City Class’ selected in ‘Personnel’.
7. **Infrastructure Bond**: It is given additional ₹20000 rebate under Sec 80 over and above ₹150000 in GPF group. However, not all such bonds are admissible, therefore, include only those which qualify.
8. **Medal Allowance**: You may not use it. It is here to complete the logic.
9. **Medical Insurance**: Limited to ₹15000. Includes CGHS.
10. **Medical Treatment**: Full amount is used, therefore, use only allowable amount.
11. **Profession Tax**: In some states this is charged and, therefore, it is deducted from gross salary.
12. **Treatment Disabled Dependent**: The rebate is governed by ‘Special Treatment’ in ‘Personnel’.

### Funds



1. The table named ‘Funds’ has been created to store bank account numbers of various private funds of unit, which are expected to receive amounts while pay is being disbursed. Accordingly, the entries in this table may be done at user end. In case, a unit does not intend to transfer any amount to its other accounts through CBS during pay, they may ignore it. The table should not contain ‘Ration Fund’ as it is catered for its funding through various messes and RMA from individuals.

### Messes



1. The table named ‘Messes’ has been created to store bank account numbers of various messes operating in the unit. Accordingly, the entries in this table may be done at user end. Unit’s ‘RMA (Ration Fund)’ also finds place in this table for crediting RMA recoveries from individuals. The same account should be repeated under ‘Messing (Ration Fund)’ with a check mark in ‘Is Ration Fund’ column so that amounts recovered from individuals against all ‘Mess’ are clubbed into it in ‘Bank Letter \*’ reports. You must ensure that only ‘Messing (Ration Fund)’ account is checked as ‘Is Ration Fund’.
2. The change from crediting messing recoveries into accounts of various messes to credit in the account of Ration Fund was an advice by the audit team of Tripura Frontier. I also found it appropriate in the light of the issue of not having account numbers for messes and, thus, resulting in undesired retention of the recoveries in Govt. Fund Account.

### Personnel

1. The ‘Personnel’ table forms part of pivot in the system as the system revolves around it for entire set of transactions. To reduce my efforts in explaining and user to comprehend, the columns of the table are described first. The interfaces working on the table are described subsequently, wherein specific purpose of the interface is given. The columns of the table are as under:

#### Columns common to both subsystems

1. ‘**Regt No**’ forms the liking key of pivotal value for various amount types in ‘Bills Sent’, ‘Pay’ and ‘Messing’ tables during any Pay subsystem transaction as well as with ‘Data’ and ‘Recoveries’ tables of Tax subsystem. The entry needs to match full 9 figures identically as mentioned in central pay system to correctly match ‘Salary’ entries created from data downloaded from pay processing every month during ‘Imports’ with ‘Salaries’ selected as well as during import of ‘Tax Data’ under ‘Tax Wizard’. This column cannot be left blank as it is the only linking parameter for records in tables mentioned above.
2. ‘**Rank**’ of an individual, as mentioned earlier, forms only title part for an individual. It can either take a value from the list provided by ‘Ranks’ table or can be left blank. While entering, the data using the forms on ‘Personnel’, any new ‘Rank’ entry can be added in in the ‘Ranks’ table by just entering a new value and telling the system to add the new entry on popup, thus, it somewhat makes ‘Ranks’ form redundant. However, this entity is checked for incomplete records shown after each ‘Imports’ process, thus must be entered to distinguish from other incomplete records.
3. ‘**Name**’ being natural identifier for an individual, is required to be entered necessarily and correctly. The entry is required by the system and cannot be left blank. However, it is not matched with the name listed in central pay system, thus, may vary from it. During discussions, it transpired and one should agree that the name of individual should match the record of Bank as well as his PAN. In case of any anomaly in the names is service record, bank and PAN, individual should be encouraged to rectify the name.
4. ‘**Subunit**’ forms part of segregation of individuals into various groups as mentioned earlier under ‘Subunits’ heading. The entry can be chosen out of the list or can be left blank. However, this entity is checked for incomplete records of personnel posted in the unit, therefore, even if segregation of personnel in subunits is not required, it is suggested to group them all in one subunit. For personnel marked as ‘Deleted’, ‘Out’ or ‘Retired’, this entry loses any value during processing. ‘Form 16’ and ‘Documents Pending’ reports of Tax subsystem are also grouped on this field to bunch them in subunits.

#### Columns used in Pay Subsystem

1. ‘**A/C No**’ is designed to store bank account numbers of the individuals. Leaving it blank could accommodate automatic record creation during ‘Imports’ with ‘Salaries’ selected for regimental numbers for which matching entry is not found in the table. However, it must be noted that the records of posted personnel and pensioners, which do not have bank account number listed, are not included in the bank data; therefore, all such records in personnel require this entry completed essentially before transaction. A unique index is also created on this field to avoid erroneous duplicate account number entry in the table. One must carefully and correctly enter these values for everyone to avoid erroneous CBS transactions, which would be extremely hard to identify and rectify.
2. ‘**Unit**’ field is designed to store the unit to which an individual is posted out or attached to having marked its ‘Is Out’ field. This field is checked for completeness of a record that is marked as ‘Out’ on ‘Personnel Out Incomplete’ form. If you do not use ‘Is Out’ in conjunction with ‘Unit’ as described here, all payments will go to individual directly and you must compile letters to the payee manually for intimation, which may lead to omissions inviting avoidable correspondence. I have not yet created a unit wise report in tax subsystem to help send them partial ‘Form 16’, but individual partial Form 16 can be printed and sent to concerned units along with movement orders so that they can include TDS data deducted by user unit.
3. ‘**Home Address**’ field is designed to keep home address of an individual for dispatch of letters, once he gets retired. This field is checked for completeness of record that is marked as ‘Retired’ on ‘Personnel Retired’ form.
4. ‘**Is Deleted**’, shown as ‘Del’ was created to mark records of personnel for deletion without physically deleting them; the legacy of version 1.0 of Pay subsystem continues. The records marked for deletion are not included in any transaction. Related records of ‘Pay’ table are automatically marked as ‘UD’, while that in ‘Messing’ table are marked as ‘Not Recovered’. This enables users to withhold erroneous payments to someone who deserted, absented without leave or is overstaying leave granted, etc. The records marked for deletion are not included in any transaction in pay. Thus, any tax recovery posted in the database would not have any effect. Therefore, I decided to forego any tax calculation for such personnel. However, if it is unchecked in Pay subsystem in later, it will be included and tax arrears will be evenly distributed across remaining months including current month. Also, any ‘UD’ in ‘Pay’ table will also be released. Care must be taken before releasing ‘UD’ if entire amount is not payable.
5. ‘**Is Out**’, shown as ‘Out’ is created to mark an individual as posted out/attached with another unit, i.e., his pay is drawn by another DDO. The ‘Out Unit Letters’ report prints letters of details of amounts paid to individuals through their DDO of posted to/attached with units having deposited their payments into accounts of concerned units. As described under ‘Unit’ field, ‘Is Out’ needs to be used in conjunction with ‘Unit’ field. As the salary of individual is not drawn in parent unit, his tax deduction is also not possible; therefore, the field is looked at to avoid any tax calculation for checked ones.
6. ‘**Is Retired**’, shown as ‘Ret’ is created to mark an individual as pensioner. Transactions in respect of these personnel are included in ‘Pensioner Letters’ report, which prints individual letters of intimation addressed at their ‘Home Address’ field. As described under ‘Home Address’ field, it needs to be used in conjunction with ‘Home Address’ field. The field is also looked at to avoid any tax calculation for checked ones.
7. ‘**S No**’. During the trial of the system, need was felt to keep entered list in the order of entry, i.e., in the sequence on paper for ease of comparison during error spotting or completing partially entered list. The list otherwise used to get sorted on ‘Regt No’ automatically being part of ‘Primary Key’. This automatic sorting made comparison of long lists formidably difficult. The ‘S No’ field is created to solve the issue as this auto incrementing number correctly maintains the original order of entry of records in the database. You should note that the order of data-entry is maintained and not the physical list, thus, I should not be blamed if a list is not entered in its physical order. For similar needs, the concept has been borrowed in this table too from ‘Messing’ and ‘Pay’ tables, which needed this functionality originally.

#### Columns Used in Tax Subsystem

1. ‘**Is Retiring**’, shown as ‘Retiring’ is created to mark an individual as retiring soon and tax is not calculated for checked ones. Suppose a person is proceeding on retirement in the month of say May of the current financial year. Having received only 3 months’ salary, he may not fall in the zone of tax; however, the system would project his salaries for entire period of 12 months deducting undue tax. I recommend checking ‘Is Retiring’ for such persons and calculate and deduct any tax due manually, if needed.
2. ‘**PAN**’ is created to hold PAN Card number of individual. It is essential for filing of return hence it is used in quarterly tax returns and ‘Form 16’ report. In addition, as per tax laws, tax is deducted at higher rates (20%) for the one who does not have it. If you want to give relaxation to those ‘APPLIED FOR’, you would need to enter a fake PAN, however, this cannot go a long way as it will be caught at first tax return and, therefore, I would discourage this method. Instead, you may arrange an agent who may facilitate possession of a valid PAN.
3. The values of ‘**Special Treatment**’ play important role in certain kind of tax exemptions to individual. ‘None’ is the default value and no special treatment is given to such personnel. As of now following options are available:
4. None
5. Disabled
6. Severely Disabled
7. Disabled Dependent
8. Severely Disabled Dependent
9. Exempted U/S 10(26)
10. ‘**CEA**’ plays its role in exemption on ‘2bb.3(v)/CEA’. If amount is present, exemption is given for specified number of children or at least one child. Earlier versions used ‘Children’ field in this table and 0, 1 and 2 were only allowed values. This could handle majority of cases, however, to provide relief in case of twins or more at second birth, I thought it prudent to replace it with ‘CEA’ and split low majority cases in ‘Hostel Subsidy’, which is explained next. The value contained in ‘Children’ will be ported to ‘CEA’, so be careful to move value to ‘Hostel Subsidy’ if someone is affected.
11. ‘**Hostel Subsidy**’ plays its role in exemption on ‘2BB.3(vi)/Hostel Subsidy’. If amount is present, exemption is given for specified number of children or at least one child
12. ‘**Own House**’ check mark is straightforward. No exemption is given for ‘Rent Paid’ if it is checked.
13. ‘**HRA City Class**’ holds either of the two values ‘A’ or ‘Other’ with ‘Other’ as default value. Exemptions are accordingly adjusted.
14. ‘**RHA Rebate**’ (not shown on Tax interfaces of Personnel) is provided to specify annual rebate admissible to the individual. You may notice that monthly amount with the same name is also stored in ‘Configuration Settings’, which is applied on all individuals unless a value is mentioned here. While applying the monthly value from ‘Configuration Settings’, number of times RHA is received by an individual is taken as number of months for which the rebate is to be extended. This proposition works for personnel, who either receive RHA for one place for the entire year or they join/leave such place midway from/to a place where RHA was not applicable. However, for those who join a place with RHA from another with a different RHA, this system would not suffice. Thus, this field was created for everyone, which is preferred over common value. Now you may understand my dilemma if I had to ask you to give me monthly value, thus, I asked you to provide total annual rebate without dividing it by 12. The total rebate is then extended to specific individual without counting for how many number of times it was received.
15. ‘**SDA/CA Rebate**’ (not shown on Tax interfaces of Personnel) is provided to specify annual rebate admissible to the individual. The reasons for this field are like the explanation given for ‘RHA Rebate’.
16. ‘**TPT Rebate**’ (not shown on the form) is provided to specify annual rebate admissible to the individual. The reasons for this field are like the explanation given for ‘RHA Rebate’.
17. Values once entered are carried forward to next database through ‘Initialize’ or ‘Import Process’ interfaces. As an exception, ‘Initialize Tax’ does resets values stored in ‘\*Rebate’ columns 0 (zero) as is considered unwanted for next Tax regime year.
18. The interfaces on the ‘Personnel’ table are described next, however, sorting and filtering of data on all these interfaces have common effects, which are described before the interfaces on the table.

#### Sorting Data

1. The interfaces on ‘Personnel’ provide sorting (smallest to largest or ‘ascending’) on any column through clicking on any column heading. The sort is on/off toggle. For example, you may sort the list on ‘Regt No’ column (or any other) by clicking on its heading, this will change the background colour of the heading to indicate the sort order, however, clicking it again would restore the background colour and sorting would not be undone. If data is not sorted on any shown column it would retain original order of entry using ‘S No’.

#### Filtering Data

1. The interfaces on ‘Personnel’ also provide text-boxes and combo-boxes in the header part of the form for filtering of records through partial entries to help searching of an individual in long lists.
2. Earlier versions of separate subsystems used to use ‘wildcards’ in the background code, which I considered easier, however, later I wished to give you complete control on this filtering. Thus, I removed internal ‘wildcard’ in the background code; therefore, you need to understand the ‘wildcards’ in SQL. Please don’t worry, it is very simple.

##### Wild Cards

1. ‘Wildcards’ are something like ‘x’, ‘y’, ‘z’, etc. in the Algebraic Mathematics that you may be able to recall easily. Yet, I will not ask for the solutions to the equation so don’t be afraid. Have you ever played ‘Rummy’ in cards’ game? The wildcard is something close to the ‘Joker’ in the game. Sorry, I used three identifiers in the sentence describing Algebraic Mathematics; simple SQL supported in MS Access VBA has only two ‘wildcards’. One is ‘\_’ (underscore character), while the other is ‘%’ (percent character). Some other SQL databases use ‘?’ in place of ‘\_’ and ‘\*’ in place of ‘%’ (this is the actual case in MS Access SQL). Yet, I find it closer to ‘Rummy’ game of cards. Once you understand these two characters you will be able to have full control on the data filtering. You only need to fire the question; solution would be given by the system … so you are the paper setter or ‘Examiner’. You need not bother about the solution to the question … the solution would always be right … SQL assures it for you. It is like asking ‘Google’ … what ‘KISS’ is … to result in ‘keep it simple and stupid.’

###### ‘\_’ (Underscore) Wildcard

1. The ‘\_’ (underscore wildcard) represents one and exactly one character in the data, i.e., ‘\_at’ will match ‘b**at**’, ‘c**at**’, ‘e**at**’, ‘f**at**’, ‘h**at’**, ‘m**at**’, ‘r**at**’, ‘s**at**’, etc., but would not match ‘at’ or ‘brat’ as the underscore must be replaced by one and exactly one character. Similarly, ‘\_\_\_’ (three underscores) would match any three and exactly three letter words like ‘cat’, ‘but’, mud’ or the likes. Further, ‘ba\_’ would match ‘**ba**r’ (people enjoy), ‘**ba**t’, but not ‘**ba**re’.

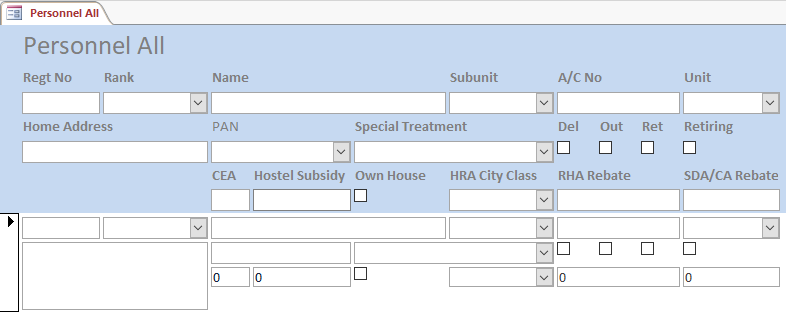
###### ‘%’ (Per cent) Wildcard

1. The ‘%’ (percent wildcard) represents none or any number of characters that can be replaced for the ‘%’ sign. For example, ‘%Lakhan%’ may represent ‘Ram **Lakhan** Singh’, ‘**Lakhan** Sharma’, or ‘Mera **Lakhan**’, or may be ‘My Name is **Lakhan**’, but not something containing ‘**La**a**khan**’. Similarly, ‘%at%’ may match ‘**at**’, ‘b**at**’, ‘br**at**’, ‘d**at**e’, ‘**at**tention’, ‘conc**at**enation’, etc.

#### Interfaces on ‘Personnel’

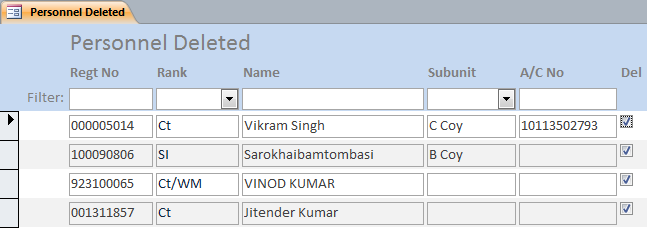
1. The ‘Personnel’ table is handled through ten different forms, which are described below. Name of all these forms start with ‘Personnel’ word to indicate their linkage with the table, followed by the purpose of use of the form. The main aim of having various forms on the same table is to segregate same data in six different lists for convenience, i.e., all personnel, personnel marked as deleted, personnel posted in unit with incomplete interface, personnel retired with incomplete interface, personnel posted out/attached with incomplete interface and personnel taxed with incomplete interface. Except for ‘All’ and ‘Deleted’ other three lists have their own validations for incomplete records, thus, additional four forms are created to present only incomplete records on corresponding interfaces.

##### Personnel All



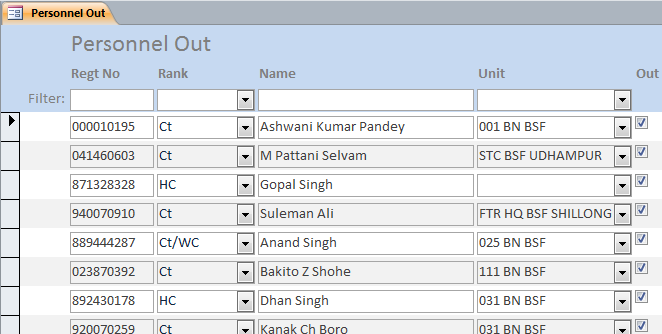
1. This interface is provided to show all the columns and all personnel listed in ‘Personnel’ table in one place. Once having segregated the lists from the same table, it was felt many times necessary to see a complete list while regimental number of someone was required to be searched. Therefore, the interface was created at a later stage.
2. As discussed under the ‘Relationships’, records of individuals having corresponding records in ‘Bills Sent’, ‘Messing’ or ‘Pay’ tables cannot be deleted. Such records can be marked for deletion by checking the ‘Del’ mark for the record. The records marked for deletion are not included in the process. Please note that marking a record for deletion would also mark all its corresponding entries as ‘UD’ in ‘Pay’ table and the records would be retained in the system. These records would eventually appear in the ‘UD List’. Records pertaining to ‘Messing’ table are marked as ‘Not Recovered’ and will appear separately in ‘Messing Recovery (Coy)’ and ‘Messing Recovery Due’ reports.
3. Besides marking a record for deletion, individuals may be marked as posted out to or attached with another unit by checking the ‘Out’ field or may marked as retired by checking ‘Ret’ field. The interface can have at most only one check mark checked out of ‘Out’ or ‘Ret’, i.e., these values are mutually exclusive. You must note that, ‘Messing’ recoveries are marked as ‘Not Recovered’ since it is expected to have been cleared by any posted out or retired personnel before relief. The reports for these personnel too do not include in ‘Messing’ table. Thus, any ‘Messing’ if it is required to be recovered from due payments of posted out or retired personnel, it should be recovered as normal recovery in ‘Pay’ table and then can be transferred to the corresponding ‘Mess’ through ‘Funding’ form during a transaction when ‘Messing’ is not part of the transaction.
4. This interface can be used to create records of backlog of posted, posted out or retired personnel. All other interfaces on personnel do not allow you to add records, however, you can modify relevant portion of records over other interfaces.

##### Personnel Deleted



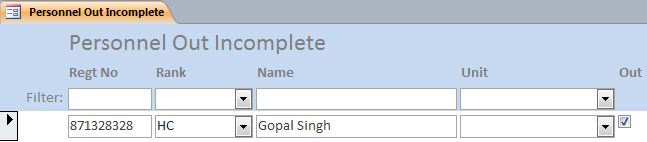
1. As described earlier during relationships, records of personnel cannot be deleted until records in ‘Bills Sent’, ‘Pay’ and ‘Messing’ tables are all deleted with respect to the individual, thus, to indicate soft deletion of a personnel in the application, the ‘Del’ check-mark has been given. The interface shows only those records, which are marked as deleted. The records of personnel so marked are not included in the process of disbursement of amounts by automatically marking amounts in ‘Pay’ as ‘UD’ and amounts in ‘Messing’ as ‘Not Recovered’. This form on the same table allows user to uncheck the ‘Deleted’ mark to eventually restore the record marked for deletion and effects pending payments and recoveries like messing, etc. The form is intended for only this purpose, thus, any other modification in the record is not allowed on this form. Since, the tax is also not calculated for such personnel, it is advisable that after unchecking ‘Del’, ‘Selective Tax Calculate’ process be performed on such individuals if ‘Salary’ part is being paid.

##### Personnel Out



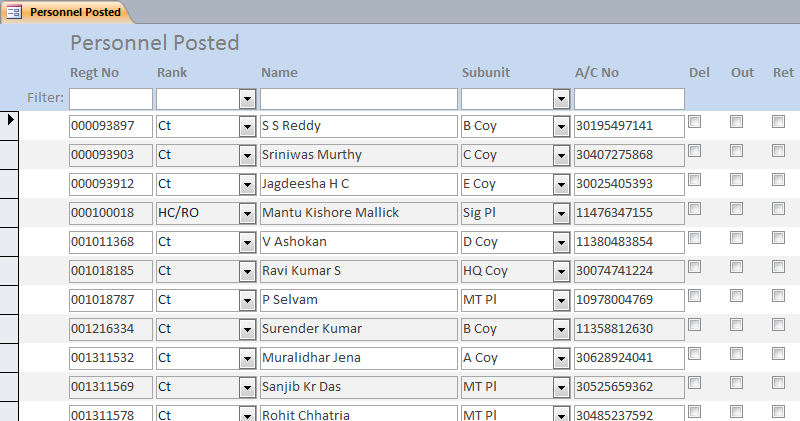
1. Once a record of an individual is marked as posted out to/attached with another unit having checked ‘Out’, it will appear on this form. The record would now require an additional entry of posted to unit, which can be selected from the list provided under ‘Unit’ column. The bank account number of the individual now loses its importance and selected unit’s bank account number will take over for further CBS transactions. You must note that if account number of a unit selected is not available in the ‘Units’ table, the transaction would be ignored during CBS transaction, though, letters of intimation would still be printed.

##### Personnel Out Incomplete



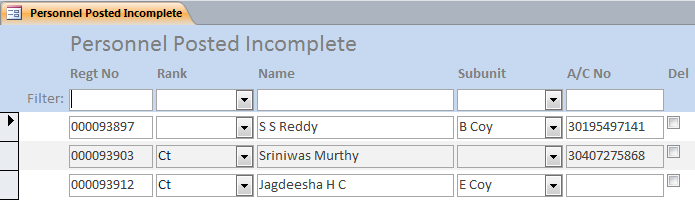
1. The form shows only those records which are marked as posted out and that do not have a unit listed in their record to which they are posted to.

##### Personnel Posted



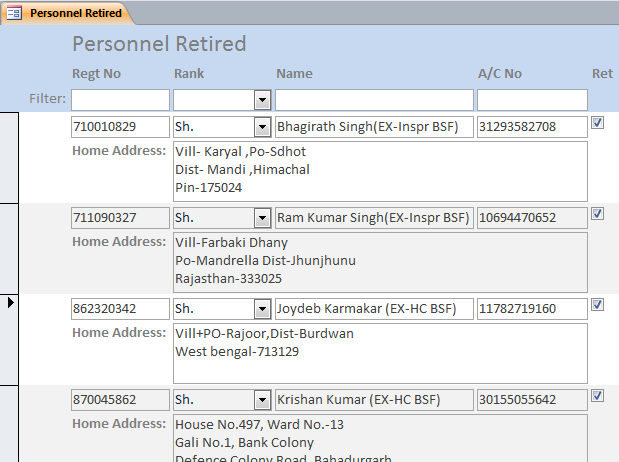
1. The ‘Personnel Posted’ form is the main data-entry form for the table in respect of personnel posted in the unit. It strips out posted out and retired personnel for ease of data-entry of mandatory list of personnel to handle.

##### Personnel Posted Incomplete



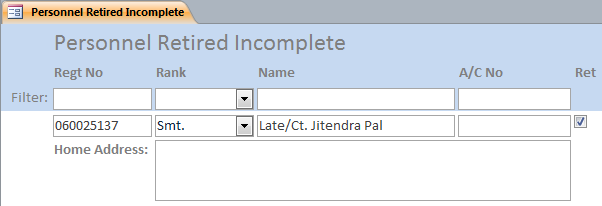
1. This is yet another interface on same ‘Personnel’ table, which filters and shows only those records which are not having one or more of the ‘Rank’, ‘Subunit’ or ‘A/C No’ entered for the records of ‘Personnel’. It is just for ease of the user to complete required data without bothering to find one in a long list of records of all personnel. In case a record is automatically created during ‘Imports’ with ‘Salary’ process, or any of the record is incomplete in respect of the fields mentioned above, the form is automatically opened (once only) by the process to intimate or prompt user to complete the required entries in the records. However, completion can be deferred and the form can be opened any time by the user before finalising the pay. I remind you once again that if ‘A/C No’ is left blank, the record is not included in the CBS transactions.
2. I must remind you that deletion of any record from this form would not be possible without deleting corresponding records from ‘Bills Sent’, ‘Messing’ and ‘Pay’ tables. Thus, if the record is created during import of salary and you incidentally find the personnel with a wrong regimental number present in ‘Personnel’ table; you must cautiously undertake the steps mentioned below:
   1. Copy the correct ‘Regt No’, as per central pay system, from the form ‘Personnel Posted Incomplete’.
   2. Open ‘Pay’ and filter for records by pasting the value in ‘Regt No (Filter)’ and pressing Enter key. Delete the filtered records of salary and/or cash withdrawal if any.
   3. Now, delete the record of personnel for which ‘Regt No’ was copied.
   4. Open ‘Personnel Posted’ form and correct the ‘Regt No’ according to the central pay system noted at step (a).
3. Repeat the process above for each automatically added record, which is otherwise available in the database with wrong ‘Regt No’. Having done this, import salary again.
4. You would be able to device your own methods for tackling such situations, but I could formulate one described above as the least time consuming and the safest. If you find another please do intimate so that it is include in the documentation.

##### Personnel Retired



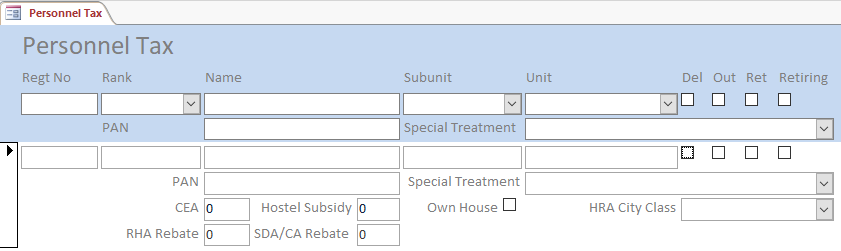
1. Once a record of an individual is marked as retired, it would appear in this list. The interface allows you to enter ‘Home Address’ required for dispatch of intimation letter.

##### Personnel Retired Incomplete



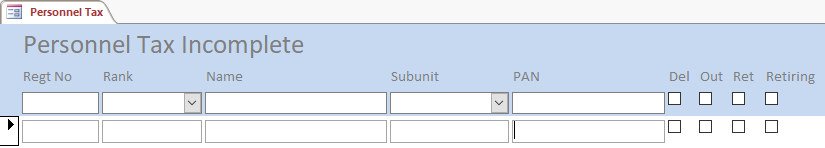
1. This interface is provided to complete ‘Home Address’ entry for personnel marked as retired and filters only incomplete records to this effect for ease of user. The interface otherwise is just a copy of ‘Personnel Retired’.

##### Personnel Tax



1. The form presents personnel data that is relevant to Tax subsystem.

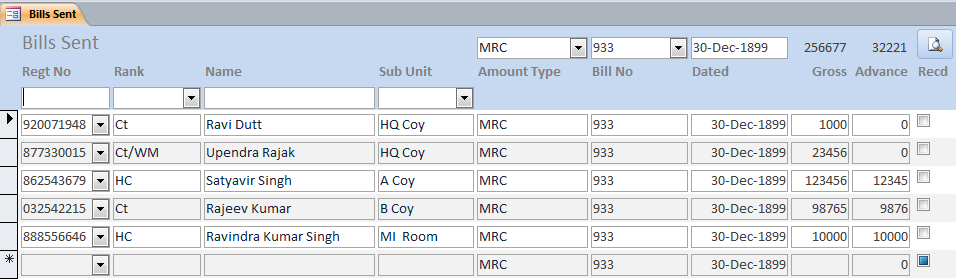
##### Personnel Tax Incomplete



1. It lists only those records, where PAN is missing.

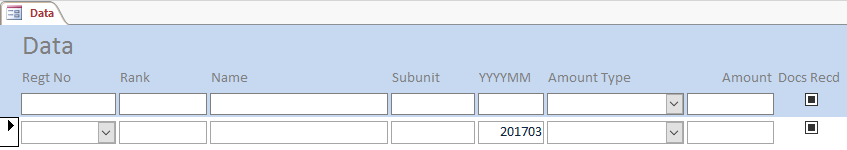
## Details

### Bills Sent



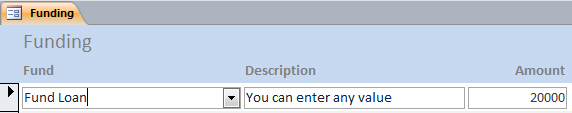
1. ‘Bills Sent’ table is designed to hold records of various bills payable to individuals and sent to PAD based on ‘Is Payable’ field of the ‘Amount Types Pay’ table checked as mentioned under the appropriate heading. This form tries to team up Accounts branch with Cash branch for speedy disposal of received bills. I request the ‘team’ that a near independent copy of the database would not only reduce the job of Accounts branch, but would also enable Cash branch to clear payments speedily in the benefit of troops. Should I remind you that personnel in Accounts branch also form part of men being paid through the Cash branch? The description of the table/form is given below:
2. Having selected an ‘Amount Type’, ‘Bill No’ and ‘Dated’ on top of the form, it would require data-entry of only ‘Regt No’, ‘Gross’ and ‘Advance’ (if any). The ‘Advance’ field can be accessed by clicking in the field to enter a value and not through tabs as I felt that in most of the cases advance is not generally claimed. The button at top-right of the form (Print this Bill) allows you to print partial sanction order (tabular part), which can be easily exported to MS Word for copy and paste operation.
3. Though, ‘Amount Types’ are automatically created in the table during ‘Import Bills’, Accounts and Cash branches are requested to arrive at a common ‘Amount Type’ value before it is added in the database. This is extremely necessary to keep the unnecessary clutter in terms of types of amounts under check in the system.
4. Before entering any record in the table, user is required to choose one of the ‘Amount Type’, choose or type ‘Bill No’ and select a date for ‘Dated’ in the header part of the form for which the amounts are being entered. I would suggest selecting dates from calendar icon that appears at the right of date selector on top of the form to avoid date format confusions in the database that may occur due to time-zone settings of Windows tending to follow ‘mm/dd/yyyy’ format. Though, there is no use of this date in the system beyond printing it on the list to be attached as appendix in the sanction order, yet its correctness would look much better than erroneous prints. These values entered/selected are then replicated in all records entered for individuals. For ease of data entry, these values have not been left editable on this form. However, adding of an ‘Amount Type’ is possible through entering a new value in the ‘Amount Type’ selection combo box on top of the form. As ‘Bill No’ is expected to receive new entries every now and then, it is only creating a list out of the entries existing in the column and it is not linked with anything else, thus, just typing a new value in the text portion of the combo-box would suffice.
5. The form shows ‘Rank’, ‘Name’ and ‘Subunit’ as extra columns from ‘Personnel’ table to decipher regimental number for each record. Thus, modification of these entities has not been catered for at this interface.
6. It needs to be remembered here that for each ‘Bill No’ an individual identified by its ‘Regt No’ can appear only once. While it helps in identifying double claims, amounts may be clubbed together in case more than one claim is required.
7. While a bill is being entered, total of the amounts entered is shown on top of the ‘Amount’ and ‘Advance’ columns in the form. This helps data-entry operator to spot if an error in amounts had occurred during data-entry.
8. The list can be sorted on any column by clicking on its heading. The sort is on/off toggle. If the list is not sorted on any column, it is sorted on hidden ‘S No’ (AutoNumber) column, which maintains the order of entry, for ease of comparison of data-entry during long lists.
9. The list can be filtered further on ‘Regt No’, ‘Rank’, ‘Name’ and ‘Subunit’ through corresponding boxes given on top of the form. For use of wildcards in filtering, kindly see ‘Wildcards’ section of ‘Personnel’.
10. To keep the system stupidly simple, I decided to print only common part of prescribed pro-forma of each sanction order, i.e., a list of tabular details for individual wise amounts in the pro-forma that may be attached as appendix to the actual sanction order mentioning ‘as per list attached’ in it at an appropriate place or may be exported to ‘Word’ for copy and paste operation.
11. The report for selected bill can be printed by clicking the button on the top right of the form.
12. Entering bills in this interface has a few advantages for Accounts and Cash branches, which are mentioned below:
    1. As mentioned above, duplicate claims can be identified to some extent.
    2. Entry in sanction order is reduced to regimental number and amount for each record compared to typing of rank and name additionally.
    3. As the utmost advantage, it provides a system for immediate payment of bills to its recipients. It becomes possible to include a payment in the process in a matter of seconds using ‘Bills Received’ form. The bank file is also created in seconds using ‘Export Trickle File’ form that can be written on CD in less than a minute, while ‘Bank Summary’ can be printed in couple of minutes over an entry level laser printer. Thus, the process can include a bill, arbitrarily long, into the process in a matter of just a few minutes.
13. I was told that separate sanction orders with list are not published in case of detachments, however, since the user will have to finally enter this list sooner or later to disburse these amounts, I would suggest one of the branch to undertake the responsibility of entry of these records at leisure before the closing of the month in their copy of the database, so that payments of detachment are not delayed and pressure during month end is not mounted on the Cash branch. Though, the onus is on Cash branch, you may choose to differ in the name of general benefit, the choice is yours.

### Data



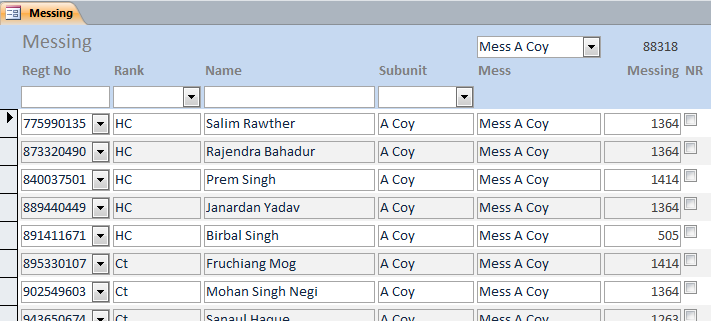
1. This form holds total annual amounts in respect of individuals that are not reported on the report from central tax subsystem, but have bearing over tax. This includes those amount types from Pay that are mentioned in ‘Amount Types Tax’ under column ‘Pay Amount Types’ and thus incorporated into tax subsystem during ‘Initialize’ and year-month of the old process is included as ‘YYYYMM’.
2. Further, you may either enter records manually or you may use ‘Tax Data’ Excel template to create data for individuals and then save it and import the same using ‘Import Payments’ sub form under ‘Tax Wizard’.
3. The checkbox named ‘Docs Recd’ needs to be checked as and when documents are received, otherwise certain entries will be shown in ‘Documents Pending’ report. The mechanism was adopted to enable DDOs to incorporate future savings of individuals on their undertaking before beginning of the tax year or anytime later to give him tax relief. However, towards the beginning of fourth quarter individuals may be reminded for pendency of documents failing which the records may be deleted and the relief will be undone in remaining months. Such deletion of records should be suggestively done before the calculation of tax for Jan so that increased tax is then distributed over two months at least.

### Funding



1. While designing the application, no clear insight could be found to link recoveries to various funds without messing up the simplicity of the system, thus entry of CBS transactions towards these funds has not been automated beyond the level of manual entry of records. This leaves the space for funds to function independently, while the system caters for important, time bound pay process only.
2. As discussed earlier, ‘Ration Fund’ should generally form part of ‘Messes’ for RMA recoveries in a pattern like ‘Messing’ recoveries, unless the process necessarily desires, it should not be included in transactions in this way.
3. The table requires selection of a fund, a little description of the transaction and a positive amount of transaction to be entered per record. You can have more than one transaction for a fund if the descriptions given are different.

### Messing



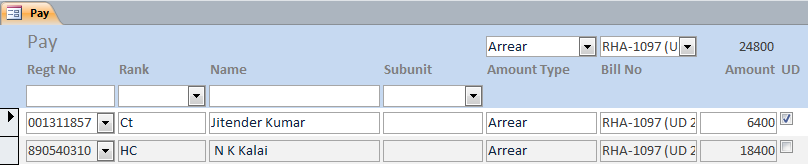
1. ‘Messing’ table is designed to hold records of messing recoveries pertaining to individuals in various ‘Messes’. These amounts are credited in ‘Messing (Ration Fund)’ account mentioned in the ‘Mess’ table. It is worth mentioning here that any other mess mentioned in the table will not find place in CBS transaction through ‘Bank Letter/Summary’, however, it will appear in ‘Unit Fund Summary’.
2. The form shows ‘Rank’, ‘Name’ and ‘Subunit’ as extra columns from ‘Personnel’ table for deciphering the regimental number for each record, thus modification of these entities has not been catered for at this interface.
3. Before entering any record in the table, user is required to choose one of the messes in the header part of the form, for which the amounts are being entered. This value is then replicated in all records entered for individuals. For ease of data entry, this value has not been left editable on this form. However, adding of a mess is possible through entering a new value in the mess selection combo box on top of the form. The Mess ‘Messing (Ration Fund)’ on which ‘Is Ration Fund’ is checked does not appear in the list as direct recoveries in this account is not required.
4. The list can be sorted on any column by clicking on its heading. The sort is on/off toggle. If the list is not sorted on any column, it is sorted on hidden ‘S No’ (AutoNumber) column, which maintains the original order of entry, for ease of comparison of data-entry from long lists.
5. The list can be filtered further on ‘Regt No’, ‘Rank’, ‘Name’ and ‘Subunit’ through corresponding boxes given on top of the form. For use of wildcards in filtering, kindly see ‘Wildcards’ section of ‘Personnel’.
6. Automatic total is also shown on top right side of the form for immediate total matching. Thus, as soon as a list is completed, its total can be matched with the physical list to find out whether there was any error in the amounts entered for ‘Messing’.
7. One thing must be noted here that for any given ‘Mess’ there can be only one entry per individual identified by ‘Regt No’. Thus, in case any ‘Mess’ lists two amounts for one individual, the amounts would be needed to be clubbed under one entry.
8. If a record in ‘Personnel’ table is marked for deletion, the corresponding entries in this table are marked as ‘Not Recovered’, shown as ‘NR’. Such entries can be printed through ‘Messing Recovery Due’ report and be distributed to ‘Messes’ for information. I have left responsibility of accumulation of old dues on the ‘Mess’ for ease of developer.

### Pay

1. ‘Pay’ table is designed to hold records of ‘Salary’ and other bills payable or recoveries pertaining to individuals based on ‘Is Payable’ field of the ‘Amount Types’ table as mentioned before. It also caters for the amounts recovered from individuals, which are payable to other units. The columns of the table are described one by one in the succeeding paragraphs.
2. ‘**Regt No**’ is the main linking field of the table with ‘Personnel’ table. The field cannot be left blank.
3. ‘**Bill No**’ forms part of identification and/or description of a transaction. This field cannot be left blank.
4. ‘**Amount Type**’ is the part of identification of a transaction together with determining the plus or minus of the amount of transaction.
5. ‘**Unit**’ if entered, indicates that the recovery made is payable to the unit selected.
6. ‘**Amount**’ of the transaction is entered here. Please note that it must be a positive number as the logic of its addition or subtraction is based on ‘Amount Type’ and not the sign of the number.
7. ‘**Is UD**’, shown as ‘UD’, can be checked to indicate the system to withhold the transaction. In case one amount is required to be kept as ‘UD’ amount, it is suggested that the amount is entered in the database and its ‘UD’ column is checked to indicate it as ‘UD’. These records are shown separately in ‘Bill Summary’ and a list of UD amounts can be printed through ‘UD List’. During initialization, these amounts are not removed; rather their ‘Bill No’ is appended with the string ‘(UD YYYYMM)’ indicating year-month of the process when the amount was marked as UD. Thus, user can unmark these records to include them in process later without having to recreate these records. In case only partial payment is required to be made, partial amount can be added under original ‘Bill No’, while holding residual value in ‘UD’ record, which now has modified ‘Bill No’. Once the ‘UD’ is finalised, records to this effect can be manually deleted. This manual deletion is not required in case full amount is released.
8. ‘**S No**’ column is created to maintain the order of data-entry as mentioned under ‘Personnel’ table.
9. For the ease of users and to reduce tabbing through fields, I have designed three different interfaces on the table to cater to different needs, which are described next:

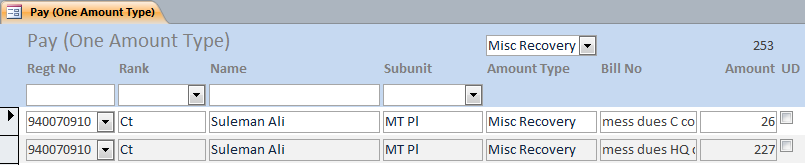
#### Interfaces on Pay

##### Pay



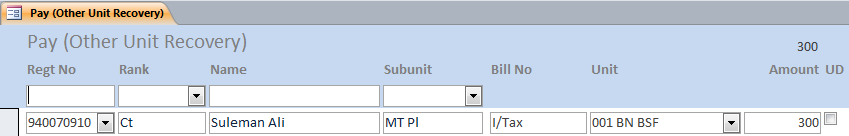
1. This is the main form for data-entry on the table. The form shows ‘Rank’, ‘Name’ and ‘Subunit’ as extra columns from ‘Personnel’ table for the purpose of deciphering the regimental number for each record, thus modification of these entities has not been catered for at this interface.
2. Before entering any record in the table, user is required to choose one of the ‘Amount Type’ and choose or type ‘Bill No’ in the header part of the form for which the amounts are being entered. These values are then replicated in all records entered for individuals. For ease of data entry, these columns have been skipped in the tab order. Addition of an ‘Amount Type’ is possible through entering a new value in the ‘Amount Type’ selection combo box on top of the form. As ‘Bill No’ is expected to receive new entries every now and then, therefore, the list is created out of the entries existing in the column and it is not linked with anything else, thus, just typing a new value in the text portion of combo-box would suffice to create a new entry along with the new record.
3. If a person happens to have been marked for deletion, to avoid data loss, programmatically records of such person are marked as ‘UD’ automatically in ‘Pay’ table and ‘Not Recovered’ in ‘Messing’ table.
4. It needs to be remembered here that for each ‘Bill No’ an individual identified by its ‘Regt No’ can appear only once. While it helps in identifying double entries, amounts may be clubbed together in case the claims are genuine.
5. The list can be sorted on any column by clicking on its heading. The sort is on/off toggle. If the list is not sorted on any column, it is sorted on hidden ‘S No’ (AutoNumber) column, which maintains the original order of entry, for ease of comparison of data-entry from long lists.
6. The list can be filtered further on ‘Regt No’, ‘Rank’, ‘Name’ and ‘Subunit’ through corresponding boxes given on top of the form. For use of wildcards in filtering, kindly see ‘Wildcards’ section of ‘Personnel’.

##### Pay (One Amount Type)



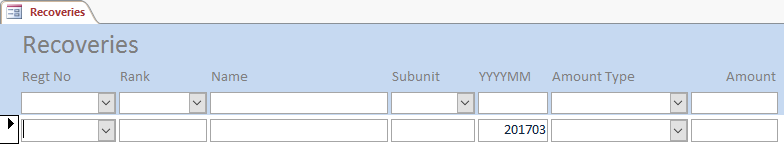
1. As in the case of GPF, each entry has a separate ‘Bill No’. On ‘Pay’ interface, the user is required to type ‘Bill No’ on top of the form and then enter the details. This was felt cumbersome in such case. Thus, this form gives an option of entering ‘Bill No’ manually for each record. Rest of the functionality is similar.

##### Pay (Other Unit Recovery)



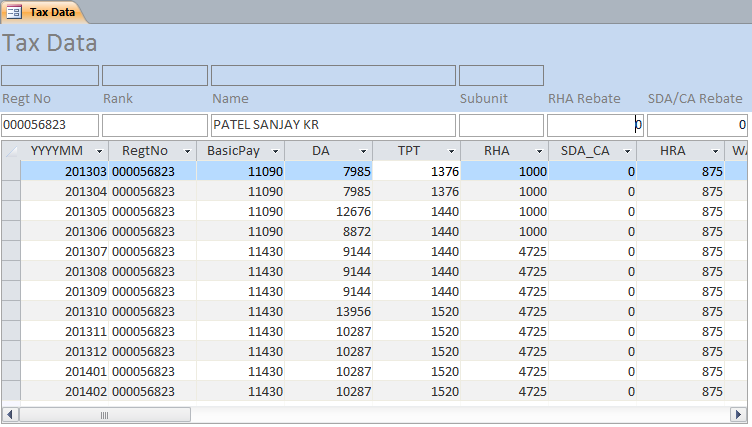
1. The form exposes the ‘Unit’ field to associate ‘Misc Recovery’ as ‘Amount Type’ with the payee unit. Rest of the functionality is same. Here ‘Bill No’ may be used to briefly indicate the reason for recovery.

### Recoveries



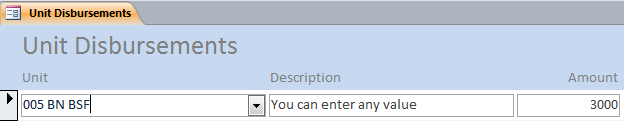
1. ‘Recoveries’ forms part of tax subsystem and has the same structure as ‘Data’. The only difference is that it doesn’t have ‘Docs Recd’ column.
2. There would be situations wherein certain amounts paid to individual are later partly recovered. Such as, part of salary in case of a period treated dies-non, pay fine, recovery of certain allowances during leave, etc. In such situations, corresponding entries may be made here. Sum of these values for given amount type and are subtracted from corresponding amount type during tax calculation.
3. You may either enter records manually or you may use ‘Tax Data’ Excel template to create data for individuals and then save it and import the same using ‘Import Recoveries’ sub form under ‘Tax Wizard’.

### Tax Data



1. The table is appended during execution of ‘Import Tax Sheet CSV’ sub form under ‘Tax Wizard’. It contains data from central pay processing system using ‘NGO’s Annual Income Tax Sheet’ that is downloaded in CSV format.
2. Though, this data is generally not required to be modified, but in certain situations, it may require modification and, therefore, this form is made at a later stage. Since, RHA, SDA\_CA and TPT are linked with presence of individual on duty they may be recovered after calculation in central database, or it may be manually claimed later. However, rebates on these allowances are monthly in nature, making their counts play important role. If these values are entered in ‘Data’ table containing payments or ‘Recoveries’ table, you must provide total annual rebate value manually in individual’s record. This is a case suited for those personnel where these values are for two different areas. However, if the values are same but depicted differently due to recovery/manual claim, it is suggested that you put actual values against corresponding months so that correct rebate values can be arrived at counting values from this table and value of monthly rebate stored in ‘Configuration Settings’ and you are able to avoid entering values in ‘Personnel’ table. You can view complete tax data, however, modification is extended to only ‘TPT’, ‘RHA’ and ‘SDA\_CA’ columns of central records.
3. Sorting and filtering is also provided on this form. You can look at ‘Filtering Data’ under ‘Personnel’ for more details.

### Unit Disbursements

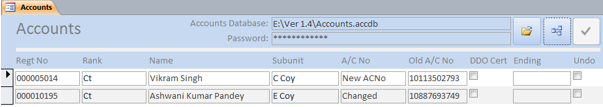


1. The table has been designed to cater for unit to unit fund transfer. The function is like funding. However, it may be required to be used rarely in situations like wrong credits.

## Other Forms

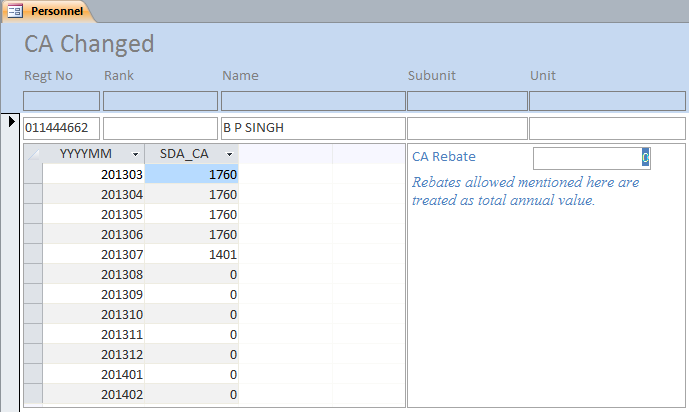
1. The tables and forms in this section are out-of-process and data used in the table is intended to only augment the main process.

### Accounts



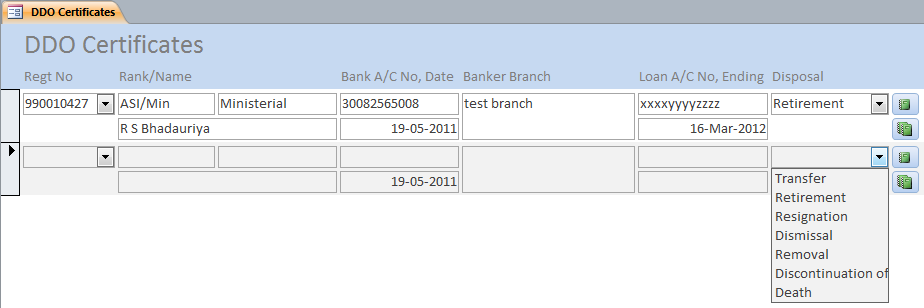
1. I did not try to incorporate security in the system for my assumed trust on the user and my strong feelings towards the fact that security is always restrictive in nature that essentially breaks the flow or freedom. Yet my cashier and teammate, R S Bhadauriya, continued to insist on a feature that at least bank account number of an individual should not get changed without the knowledge of the cashier. His contention was that while human eyes are designed to spot errors in text, it is very hard to train them to spot errors in figures and any erroneous change in account number may be catastrophic. He was right in a sense and I had to yield to his request. Further, in the light of the binding undertaken by DDOs in certificates issued to individuals for bank loans, this control over change was considered even more valuable. Since, this was not feasible without separating tables from forms, which needed too much of work everywhere, a rudimentary work around was provided using an external password protected database and using this table. It is a kind of temporary table that is populated through comparisons and changes of account numbers in the personnel table and those stored in external database. On closing of the form, the data is automatically deleted. As of now, this check is limited to personnel posted in the unit and retirees of the unit. It does not cater for posted out personnel as their payments are intended to be paid through their present unit where they are posted now.
2. Though, the form can be used independently, but it is designed to open automatically before the trickle file can be exported using ‘Export Trickle File’ form and supposed to work in conjugation with it. The method employed is not at all fool proof, rather very easily breakable, yet it provides a strong mechanism to find out and disapprove or approve changes in account numbers of individuals each time before export for actual transaction.
3. The first button on the top of the form with folder icon opens a file open dialog to select the external password protected database, which has been separately created as by the name of ‘Accounts’. Once the process is complete, the setting is saved and you need not browse the database again unless it is renamed or moved somewhere else. On top of the form appropriate ‘Password’ should also to be provided before proceeding any further. Kindly see the ‘Accounts: Password Protected Access Database’ under ‘External Extensions’ section to know how to password protect a database.
4. The second button compares account numbers stored locally in ‘Personnel’ table with that are stored in externally attached database. All new records and those having changed account numbers are then added in the local table named ‘Accounts’. Further, whether a DDO’s certificate has been issued or whether a person is a retiree are shown. If an individual happens to have been issued a DDO certificate or he is a retiree, the ‘Undo’ column is automatically checked for changed account numbers.
5. User is now required to disapprove records by check marking ‘Undo’ in the records that are erroneously changed. Remaining records are considered as approved. However, the form does not allow change of account number of retirees as well as those issued with DDO’s certificates and, therefore, automatically rechecks the ‘Undo’ selector.
6. Having disapproved erroneous changes, the third button on click does the needful and closes the form. If the form was opened using ‘Export Trickle File’, it will now be opened for the needed export. However, if there was no change in the account numbers and no records were added, the form doesn’t wait for approvals and directly opens ‘Export Trickle File’ form.
7. I must inform you that the **‘Accounts’ database once attached with the system** can survive renaming and/or moving across the file system, but it may not survive replacement by another database or manual changes in its data stored. You would also not be able to use older copies of the database in most cases. It is therefore recommended that **the database be kept safely as ‘key’ and manual modifications must not be done in the database**. It is further recommended that the database be decrypted by removing password and handed/taken over between cashiers as keys during changeover along with other keys, which should be re-encrypted by the one taking it over using his own password instead of sharing passwords.

### CA Changed



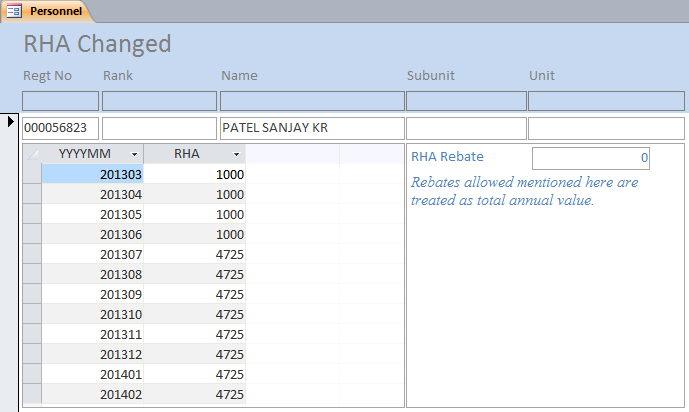
1. You have already read the reasons why ‘RHA Rebate’ and ‘SDA/CA Rebate’ columns have been provided in ‘Personnel’ table under ‘Personnel’ heading. However, I thought that it would be difficult for users to identify such personnel especially in large units like Bn. This form can be helpful in finding such personnel. It filters out only those records of personnel, who have two or more different values excluding 0 in ‘SDA\_CA’ column. It then shows corresponding records in ‘TaxData’ table and offers to provide appropriate value(s) for respective rebates. The form, thus, has only this column editable.

### DDO Certificates



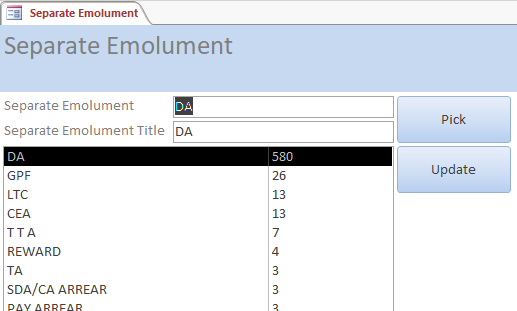
1. This table along with a report with the same name is prepared to issue DDO’s Certificates to individuals for grant of bank loan. While the functionality allows the branch to issue predesigned certificate, it also captures requisite data in respect of the individuals issued with the certificate. The report is printed for an individual by clicking the upper button in front of the record. It prompts for gross salary, which is printed as that of previous month of system date.
2. While issuing the certificate, you are supposed to feed or select ‘Regt No’ and enter ‘Banker Branch’. The branch address need not include SBI as the bank name.
3. Once the bank intimates the unit regarding sanction of loan to individual, unit is supposed to capture ‘Loan A/C No’ and ‘Date of Ending’ of loan recovery.
4. In case of transfer, retirement, etc., the unit is supposed to intimate bank as per undertaking in the certificate issued, which can be done after entering ‘Unit’ and ‘Home Address’ in the ‘Personnel’ table, having checked ‘Out’ or ‘Retired’ in ‘Personnel’ and then selecting one of the values for the ‘Disposal’. Having completed this data, you may click on the lower button in front of the record. It is recommended that you keep a printed copy of the report for record as the record would be removed during next ‘Initialize’.
5. In addition to providing a facility to print certificates and intimation letters, it also provides requisite data for committed undoing of change of account number of an individual during checking of changes in this regard before exporting bank file as described under ‘Accounts’ form.
6. I have seen an infectious tendency in my unit in applying for DDO’s certificate. While this tendency increased the cases of issue of certificate, a DDO needed a little more control over its issue and therefore, maintenance of record was considered necessary. There were cases when men used to get a certificate issued just as a matter of curiosity and then they never applied for loan. Later, on asking for the details of loan, if they were sanctioned any, they replied in negation, by when they had already ‘LOST’ the certificate. This tendency made me to consider an expiry date for the certificate. I decided to give it a clear six months’ validity at the time of issue, kept it for one more month to clear validity and then removed the record during ‘Initialize’ if no bank intimation was received.
7. While the note to this effect on the certificate issued, absolved the issuing DDO from the life-long, open ended commitment of undertaking after genuinely long enough six months, it was always possible to recreate the record if a bank delayed in intimation of sanction of loan or if it was sanctioned on an expired certificate.
8. The life of the certificate is controlled by a hidden ‘Configuration Setting’ named ‘DDO Certificate Life’.

### RHA Changed



1. You have already read the reasons why ‘RHA Rebate’ and ‘SDA/CA Rebate’ columns have been provided in ‘Personnel’ table under the same heading. However, I thought that it would be difficult for users to identify such personnel especially in large units like Bn. This form can be helpful in finding such personnel. It filters out only those records of personnel, who have two or more different values excluding 0 in ‘RHA’ column. It then shows corresponding records in ‘Tax Data’ table and offers to provide appropriate value(s) for respective rebates. The form, thus, has only this column editable.

### Separate Emoluments



1. I finally realised that I dumped this page saving concept deep inside configuration and users tend to ignore it sometimes. Therefore, this form was created to present record-counts in ‘Pay’ table for payable ‘Amount Types’ excluding ‘Salary’ in descending order of counts. When it loads, it reads the values of ‘Separate Emolument’ and ‘Separate Emolument Title’ from ‘Config’ table and displays it in the textboxes provided for this. The top record in the list provided below is automatically selected. When you click on pick button, the value selected in the list is picked for both. The form does not allow editing of ‘Separate Emolument’ for exact match, however, title can be edited by the user. Having done that click on ‘Update’ button to update ‘Config’ and close the form.
2. This is required once per process before ‘Unit Bills\*’ and ‘Unit Pay Summary\*’ reports are printed.

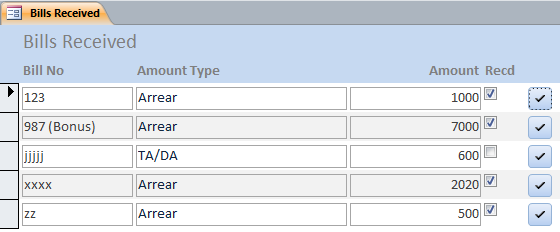
### Start-up



1. Developers would agree that developing a live system, i.e., a system which is in real use while it is being developed, forces anxiety on the developer. This anxiety is greater if developer knows that any unnoticed fault, especially in calculations, would be Omni-present and would be very hard in undoing. As the application deals with outside organisation and features of the application in use with the bank are not known, the fear of unknown is always there on the developer. The situation needed utmost care for developer and even greater confidence of the user as it did not have a trial phase involved.
2. While dedicating the database to the cause of ease and efficiency in the field of disbursement, I thought it would be bad if I do not give credit to my team member, R S Bhadauriya, who not only gave me much needed feedback during development, but also showed unquestioned confidence in still growing system, without which a developer would have certainly felt shaky in working on the live system that was going to change an age-old tested manual system through limited automation. The unquestioned confidence shown by him afforded me the much-needed assurance of the success of the system. For a developer, loss of even a bit of such confidence of the user was highly unaffordable.
3. Thus, I could not avoid desire of showing the credits for the first time. Having shown the credits, the form automatically unloads and opens the ‘Initialize’ form if a new copy of the database is opened. This is the process that you would be using in real usage situations. However, if database is not initialized, it would open ‘Configuration Settings’ form next time onwards. If the settings have been done manually or through ‘Initialize’, it would just unload. The form remains visible for 5 seconds (you may bear this cost for a freeware, open source application); however, you can click on the form anytime to unload it. Any annoyance is regretted. I just wanted you to know who all developed the application for free.

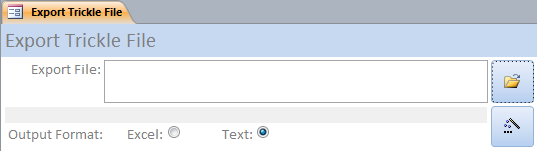
## Automation Forms

### Bills Received



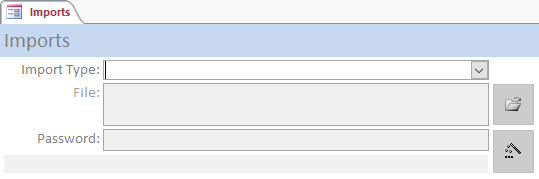
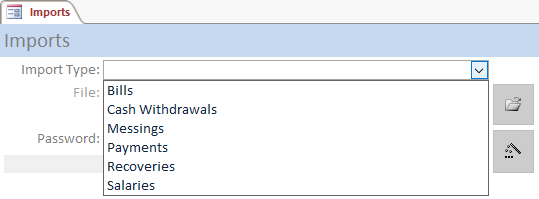
1. This form is created to indicate to the system regarding receipt of a bill through check mark against the bill in the list of all bills in the system. Since, the list is created through a query that cannot be updated, symbolic check button was required to check or uncheck a record, in fact, all records pertaining to the bill in underlying table.
2. As soon as a record on this form is checked, data pertaining to the bill is automatically copied into ‘Pay’ table, thus, including the bill in the process. Erroneous click on the button can be easily undone by clicking it again. As soon as the check mark is removed, related data from the ‘Pay’ table is deleted.
3. Due to this from, working in conjugation with ‘Import Bills’, it becomes possible to include any bill immediately in the process of payment. During the trial, we realised to have received another detachment bill of personnel just a day before the ‘PAY-DAY’. However, it could not be included in the process, since it required entry of a long list and we were not sure enough to have completed that in time, thus, we avoided to venture. The ‘Bills Sent’ form for pre-entry of such bills out of the process and ‘Import Bills’ form for automatic inclusion of such pre-entered bills is the solution to that helplessness.

### Export Trickle File



1. As the final data of CBS transactions, units are required to create a plain text file through SBI Trickle Feed software or are required to give data in Excel format. Since, data required is already available duly processed, the form ‘Export Trickle File’ exports the data in required format. Yet, as described under ‘Accounts’ form heading, this form automatically opens ‘Accounts’ instead of opening itself. Now having approved the account number changes, it is opened for use by ‘Accounts’ form.
2. For user interaction, the interface provides only two buttons. The first button that is shown with a folder icon, if clicked, opens a file save dialog in which user is required to type/select name and location for the text/Excel file to be created. You would also need to choose one of the output formats from ‘Excel’ or ‘Text’. Clicking on the other button would create the file in a matter of a few seconds. A progress bar appearing and finally closing of the form is to be taken as the intimation towards completion of the process.

### Imports

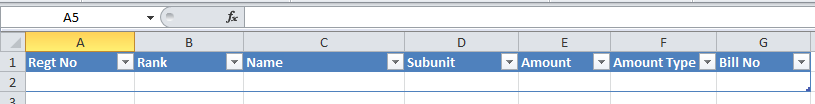
1. Having survived with many ‘Import\*’ interfaces in disconnected subsystems for around eight years, and then while combining them for good reasons, I felt I should reduce some clutter for users and therefore, I combined six imports into this one form, which you need to select at ‘Import Type’.
2. First button with folder icon on this interface, when pressed, opens a file open dialog. User is required to select the copy of Access database/Excel workbook based on template. If the database is password protected, appropriate password may also be given in the text box provided for the purpose. Having provided the file and password (if it was a database), the other button with wizard icon would import the records in the corresponding table. The records that are rejected by the process are stored in the ‘Failed Imports’ table so that user may take corrective steps and include them too.
3. A progress bar and finally closing of the form is to be taken as the intimation towards completion of the process.
4. Right from the start of the design of the system, I was aiming at reusing data available in soft form so that unnecessary data-entry could be avoided. I do understand that systems immeasurably fail if you ask for any uphill data-entry efforts without demonstrating fruits of the efforts reaped. I knew that certain soft copies in some or the form were being maintained by branches, which could be imported into the system. Therefore, I designed a few external Excel templates to convert data into predictable formats using copy and paste and then using corresponding interfaces to import into the system. Now while merging the two ‘Pay’ and ‘Tax’ subsystems, I feel it is a high time to merge certain import procedures.

#### Bills (Import from Access Database)

1. Towards the final stages of the development of initial versions, it was felt that Accounts and Cash branches should use the database on time-sharing basis if only one copy of database is used. While it would be a bit inconvenient to the user, bill sending by Accounts branch would be delayed during the time-bound pay disbursement process undertaken by Cash branch. Therefore, as a work around, this process now allows users to utilise two separate copies of the database, one each for Accounts and Cash branches, working in conjugation.
2. Accounts branch, in its own copy, can keep on sending bills, while cash branch can utilise its copy for data entry of items other than the bills. Before finalisation of the process, at any time, the bills sent by Accounts branch can be inserted in the copy of Cash branch through this interface using ‘Bills’ as ‘Import Type’ and may be selected for payment through ‘Bills Received’ interface.
3. I did not delete records from the copy of database used in Accounts branch during this import assuming a possibility of import at a time when a long list of a bill could be mid-way, whereas, pay process would not wait for its completion being time bound. Thus, I would suggest that after completion of the process, Cash branch should handover a copy of ‘Bill Summary’ report to Accounts branch, using which Accounts branch may check mark bills on ‘Bills Received’ interface and then they may ‘Initialize’ their own database to keep their database in sync with Cash branch. If this synch is not achieved, it would force the copy of database used by Cash branch to unnecessarily hold already paid up bills, which is undesirable.

#### Cash Withdrawals, Payments and Recoveries (Import from ‘Pay’ Excel Template)

1. I decided to design an Excel template to take a feed for payments/recoveries for ‘Pay’ table. I thought that it will allow user to somehow distribute the job of data-entry or prepare otherwise available data for import into the process. Following template was therefore created.



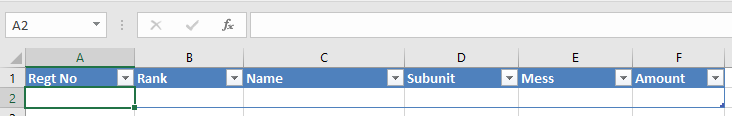
1. The system during import process remains concerned with only ‘Regt No’, ‘Amount’, ‘Amount Type’ and ‘Bill No’ columns. The records that fail during import are stored in ‘Failed Imports’ table. User may add/delete other columns as per his choice. The system, however, needs these four column headers on top row of the first sheet of the file.
2. In case some records fail the import, ‘Failed Payment’ table is automatically opened for correction of data and individual records may be uploaded from there itself.
3. Selecting ‘Payments’ on the ‘Imports’ form is straightforward; however, you need to note a few things here. If ‘Cash Withdrawals’ is selected, the form drops and ‘Amount Type’ in favour of ‘Cash Withdrawal’ and ‘Bill No’ in favour of ‘Cash’. Similarly, in case ‘Recoveries’ is selected, the system drops ‘Amount Type’ in favour of ‘Misc Recovery’, but uses ‘Bill No’ as description of the recovery thing to reduce number of entries in ‘Amount Types Pay’ table.

#### Salaries (Import from SALARY\_FOR\_IMPORT in IPP)

1. One of the aims of any automation is to reduce avoidable workload on the data-entry operator, thus, records of salary are included in the process through an import process. The ‘Imports’ form with ‘Salaries’ selected does the needful for you.
2. As described earlier, a report has been designed by IT Wing especially for this application and has been placed under link ‘SALARY FOR IMPORT’ for download by unit DDOs. This report must be downloaded in MS Excel format. The file so downloaded can be imported in the system.
3. The process performs following steps in its import action:
   1. Imports the data in MS Excel file into a temporary table created during the process, which is named ‘Imports’.
   2. All records in ‘Pay’ table having ‘Amount Type’ and ‘Bill No’ equal to ‘Salary’ are deleted, allowing retention of ‘UD’ records of ‘Salary’. If ‘Cash Withdrawal’ value in ‘Configuration Settings’ table happens to be greater than 0, records having ‘Amount Type’ equal to ‘Cash Withdrawal’ and ‘Bill No’ equal to ‘Cash’ are also deleted during this step.
   3. In case match for any ‘Regt No’ in temporary table containing salaries, is not found in the ‘Personnel’ table, a record with ‘Regt No’ and ‘Name’ is created in the ‘Personnel’ table.
   4. The data imported in temporary table is then transferred to ‘Pay’ table. During this process if ‘Cash Withdrawal’ value happens to be more than 0 in ‘Configuration Settings’, corresponding records of ‘Cash Withdrawal’ are also created in ‘Pay’ table, one record of ‘Cash Withdrawal’ for each record of ‘Salary’.
   5. The ‘Imports’ table is deleted.
   6. In case a record of personnel is found to be incomplete in respect of posted personnel, the form ‘Personnel Posted Incomplete’ is opened to prompt user to complete the records.

#### Messings (Import from ‘Messing’ Excel Template)

1. I decided to design an Excel template to take a feed for recoveries for ‘Messing’ table. I thought that it will allow user to somehow distribute the job of data-entry for import into the process. Following template was therefore created.



1. The template is straightforward; only three columns namely, ‘Regt No’, ‘Mess’ and ‘Amount’ are used in the system. I decided to include ‘Mess’ as part of columns of template instead of file name used earlier, so that characters that could not be used due to naming limitations under OS filesystem earlier could now be used.

### Import Personnel



1. I understand the difficulties that are encountered during initial data capturing in master tables. This difficulty together with inertia or fear of unknown leads a user to avoid using the automation. I am aware that creating tabular data in Excel is generally considered easier than on a form of Access, where limited copy-paste facility is generally available. I therefore decided to standardise the procedure of importing initial data with respect to ‘Personnel’ table using an Excel sheet in the following format.



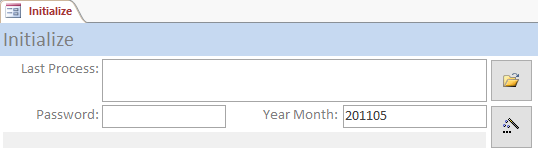
1. This caters for data related to posted personnel only. Data in respect of retired or posted out personnel is advised to be entered using ‘Personnel\*’ interfaces. I feel that you would not require this Excel template and form again unless you are going to receive bulk of manpower coming to you every now and then like in case of STCs.
2. Since ‘Ranks’ and ‘Subunits’ are also created using this file during the import, I would recommend you use these values carefully to reduce any unnecessary clutter in these tables. I must remind you that ‘LC’, ‘RC’, ‘Ct’ and ‘Const’ are entirely different values in the system, while all are used for constables.
3. The first button, with folder icon, on the form shown above, opens a file open dialog to enable user to select the file. The second button, with wizard icon, does the import for you.

### Import Process



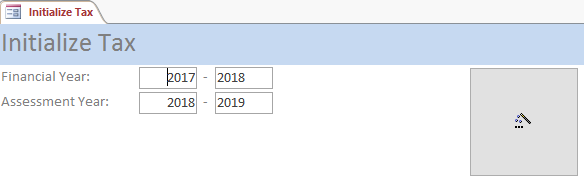
1. During the process of development, it was desired many times, to include new set of features that were incorporated after the feedback from the user, while the process was still on. Every time a change was made, I had to manually import data from various tables. After the table design stabilised, I decided to automate this process for the user so that he can utilise updates on the system even before a process is complete, which was otherwise designed to be included at the end of the process through ‘Initialize’.
2. The form has similar interface, in which the first button with folder icon on it would open a file open dialog where you would be required to select database in use for the process. If the database happens to be password protected, appropriate password may also be given in the text box provided for it. The other button with wizard icon, when clicked, imports the data in the new modified application. You may now continue with the process further in the new database.

### Initialize



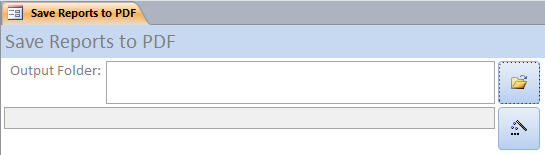
1. To keep the size of database under control and to avoid ceiling limit of AutoNumber ever achieved in ‘S No’ fields used to maintain original order of list entered, a process starts from creating a blank database at a user decided location and name. Having opened and enabled contents of this new copy, the user would be presented with this form as the first activity.
2. First button with folder icon, when pressed, opens a file open dialog. User is required to select the database he worked during previous process (completed). It does not allow selecting the database itself which is running the form, i.e., self-initialisation is not allowed to avoid any loss of data. If the database happens to be password protected, appropriate password would also be required in the text box provided. User is then required to enter appropriate value in ‘Year Month’ space; default is presented to user using the system date. Having done this, the second button with wizard icon is required to be pressed so that following actions are performed:
   1. Imports data contained in ‘Configuration Settings’, ‘Payment History’, ‘Amount Types Pay’, ‘Amount Types Tax’, ‘Funds’, ‘Messes’, ‘Ranks’, ‘Subunits’, ‘Units’, ‘Amount Type Mapper’, ‘Personnel’, ‘Bills Sent’, ‘DDO Certificates’, ‘Pay’, ‘Data’, ‘Form 16’, ‘Recoveries’, ‘Tax’, ‘Tax Data’, ‘Tax Return’ and ‘Year Months’ tables from the selected database.
   2. Moves data from ‘Pay’ table to ‘Payment History’ leaving records marked as ‘UD’. As of now date of payment is taken from the system date when the process is performed. It is, therefore, suggested that the process be done immediately after successful transaction at bank or at least on the same date.
   3. Updates remaining ‘UD’ records to change their ‘Bill No’ appending ‘(UD yyyymm)’ using ‘yyyymm’ of older database while sparing still older ‘UD’ records.
   4. Deletes records from ‘Bills Sent’ table, which have been included in the process, i.e., the records that are found checked.
   5. Saves the value for year-month in the ‘Configuration Settings’ table for use in various reports.
3. This import and clean-up process as discussed above, while made life simpler for the developer, also allowed to keep the lists manageably short and helped keeping the database size smaller for responsiveness. For simplicity of the system, the database can contain only one process at any given time. It also provided a mechanism to incorporate updated versions of the database that may be released in future.
4. Closing of the form is to be taken as the intimation towards completion of the process.

### Initialize Tax



1. For the first time and thereafter on completion of pay of Feb this process is required. I mean to say that you need it once for each tax year (Mar to Feb). Here the form resets the current database for Tax subsystem to new Financial/Assessment Year. Since, data in Tax subsystem and payment history is deleted, care must be taken before hitting the wizard button. However, before deletion of these records, it saves reports in PDF format in specified folder. See ‘Save Reports to PDF’ for more details.
2. You may also notice ‘Financial Year’ and corresponding ‘Assessment Year’ on top of the form. Here you can modify only the first textbox, rest of the values are correctly calculated. By default, the financial year is made from system date, however, you may change it to your requirements if need be.

### Save Reports to PDF

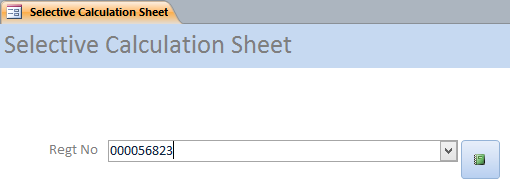


1. The reports in the database can be printed on normal printers as hard copy outputs for record. To keep a digital record of the process, each report has been designed to be saved as popular PDF just by double-clicking the page portion of the report in any view other than design. It was felt that this digital record would ask user to open each report for double clicking on it to save as PDF. Each time the report was saved, user interaction was needed. The form automates the process for user leaving certain reports that are not considered part of main process. It makes PDF of following reports:

|  |  |
| --- | --- |
| * 1. Annual Tax   2. Bank Summary   3. Bank Letter   4. Bill Details   5. Bill Summary   6. Calculation Sheet   7. DDO Certificates Register   8. Form 16   9. Messing Recovery (Coy)   10. Out Unit Emails   11. Out Unit Letters   12. Out Unit Summary   13. Payment History   14. Pensioner Letters | * 1. Pensioner Summary   2. Recovery List   3. Recovery List Separated   4. Tax Return Q1\_2   5. Tax Return Q2\_2   6. Tax Return Q3\_2   7. Tax Return Q4\_2   8. UD List   9. Unit Bills   10. Unit Bills (Coy)   11. Unit Fund Summary   12. Unit Pay Summary"   13. Unit Pay Summary (Coy) |

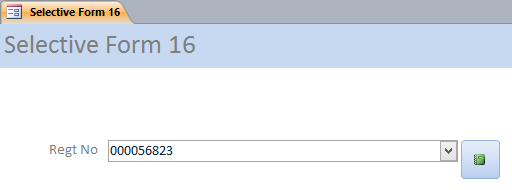
1. The interface is like many other forms. The first button with folder icon, when clicked, opens a folder select dialogue. Using the dialogue, you may choose an existing folder or create a new one at a desired location. Once a folder is selected, clicking of the button with wizard icon saves each report with its name at desired path as PDF. If any report happens to be without data, it is notified and the same is not saved.
2. A progress bar and closing of the form may be taken as the indication for completion. ‘Cheque No’ and ‘dated’ prompt would appear for both bank reports and I would suggest you to correctly provide the data for correct outputs. If it is not provided, the reports will have blank spaces reserved for it where you could write by hand if you happen to print it.

### Selective Calculation Sheet



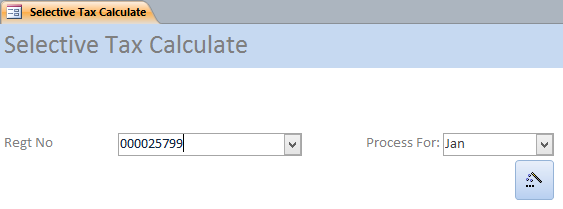
1. At times, it is felt to directly open record of a person. This form helps in such situations and opens corresponding record in the ‘Calculation Sheet’ report.

### Selective Form 16



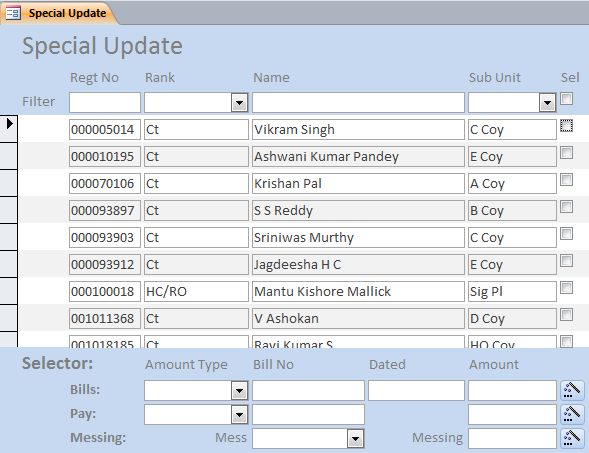
1. Similarly, this form opens selective record of an individual in ‘Form 16’ report.

### Selective Tax Calculate



1. In the event some data is required to be changed in ‘Data’ or ‘Recoveries’ after the calculation of the tax, I thought it prudent to recalculate tax selectively for an individual instead of embarking into time consuming complete reprocess. This form does the trick for you in seconds and opens the corresponding ‘Calculation Sheet’ also to verify the results.
2. Nooooooooo! I made it for testing, but later found it a useful method as a by-product.

### Special Update

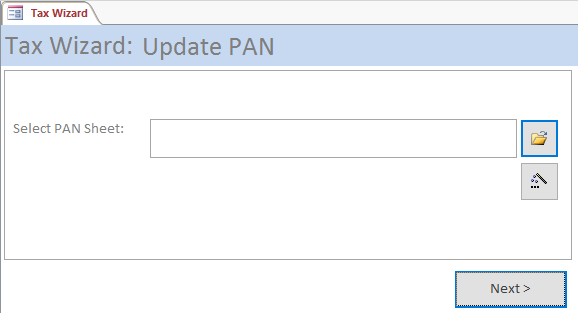


1. During development of the database, I felt that in certain situations it may be desired that same amount is inserted for a list of personnel in ‘Bills Sent’, ‘Pay’ or ‘Messing’ tables. I also felt that selecting personnel from a list by check marking may be easier than entering full 9 figures for each personnel in the long list, thus, a special form named ‘Special Update’ is created. The name legacy continues from days prior to my days in IT Wing. It uses a temporary table named ‘Special Update’, data in which is created on opening the form and deleted on closing it. To show the list of personnel and their selection, filtering and sorting on the data is provided.
2. At the top of the form a check mark for ‘Select All’ is also given to select or unselect all filtered personnel that are shown in the list retaining any other personnel already selected, thus, helping a user to select or unselect group of personnel by just checking one mark instead of too many, applying multiple filters one by one.
3. The form provides three rows below the list of personnel, named ‘Bills’, ‘Pay’ and ‘Messing’. Having selected personnel from the list, appropriate row may be completed according to the target for automatic data creation before clicking on the button in that row. The records would be created for marked personnel in a matter of seconds. The selection list is cleared having created records for one of these tables and the form becomes ready for next special update.
4. For example, if a user wants to insert ₹ 3,500/- as ‘Cash Withdrawal’ from a list of personnel drawing this cash, he may select their records and on ‘Pay’ row in the bottom, enter appropriate values and then click on the wizard button in the row.
5. The form would also find place in situations of RMA increase as you would be needing to recover more advance from all affected personnel.

### Tax Wizard

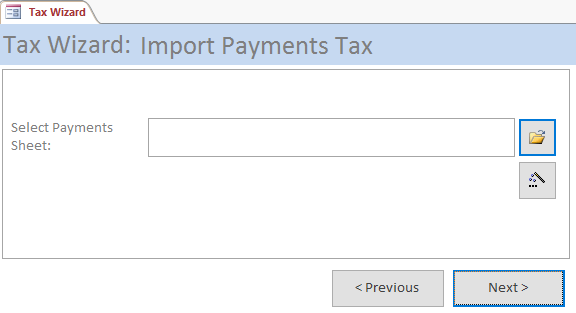
1. Every time a pay process is over, you need to download tax data from IPP. Keep any payments and recoveries handy that you want to import and then run the wizard form. It will guide you through step by step process as per following sequence:

#### Update PAN



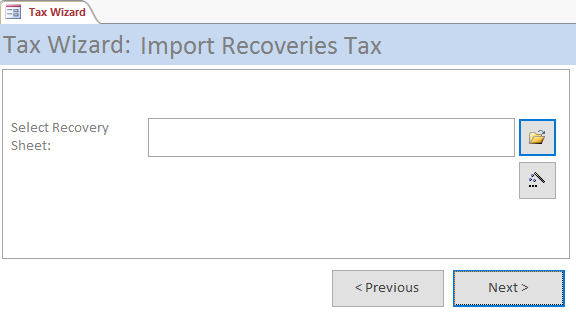
1. First, it offers you to provide an Excel sheet in externally provided template, wherein PAN details are given. Once you import this sheet, corresponding PAN numbers of individuals will be updated. However, in case no PAN update is required, you may press ‘Next’.

#### Import Payments Tax



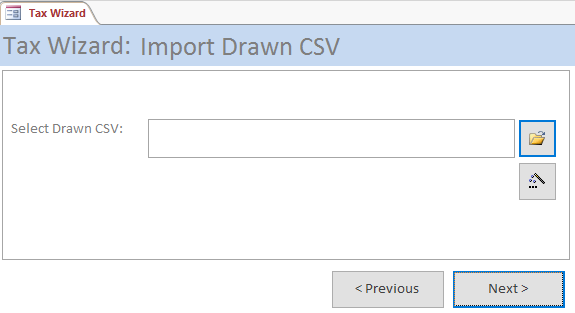
1. The wizard now offers you to include external Excel sheet form ‘Tax Data’ template for import of tax related data in ‘Data’ table. You can include more such files, or move forward to next stage.

#### Import Recoveries Tax



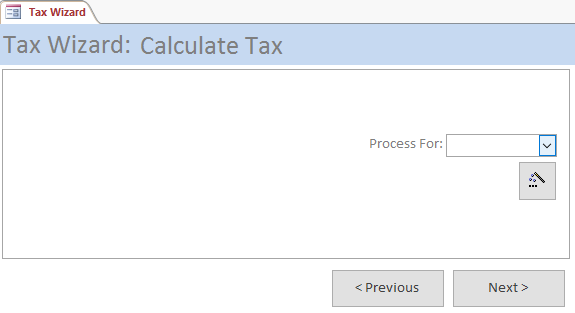
1. The same ‘Tax Data’ template of Excel can be used to import data into ‘Recoveries’ table to correspondingly reduce those amount types during tax calculation. You can move further by clicking ‘Next’ button.

#### Import Drawn CSV



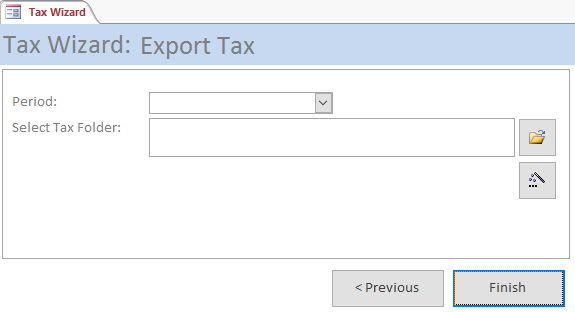
1. This form offers you to import downloaded tax sheet, which you would have downloaded in CSV format from IPP. Import the file. This will only append any data in ‘Tax Data’ table, thus, any changes in previous data are not lost. You can now click next to proceed further.

#### Calculate Tax



1. Now select a month for which the calculation is required to be done and click the magic-wand. The process would take some time say 30 minutes or so, but would do the final calculations. However, some finer points need some deliberations. First point is that you may recalculate the tax for any given month as many times as you want, but once you have calculated tax for any month, you cannot calculate tax for any previous month. I believed it to have been recovered already. Second, be doubly sure not to choose a wrong month as it will lead to results not catered for by me as of now.
2. Since it takes time to calculate tax, you might like to have a good tea-break. Even otherwise you might have been tired reading this boring user manual.
3. How was the tea break? ☺
4. Now you are ready to move to next and final step.

#### Export Tax

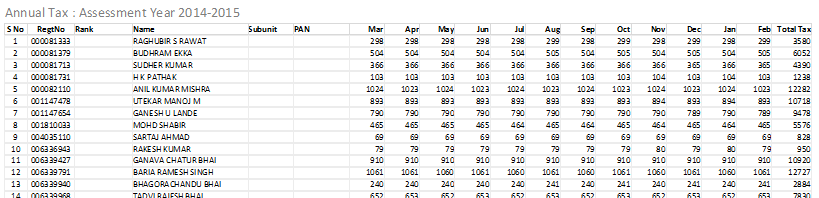


1. This is the final step in the wizard. Here you can select to export tax for any month or quarter in Excel sheet for outside consumption. Please note that after merging the Pay and Tax subsystems into one, Tax exports are not required by the system as it used to be the case earlier separated systems. The system now updates ‘Pay’ table for corresponding ‘Tax’ recoveries.
2. At this stage, you may want to click finish.
3. The wizard form is there to cycle you through steps; it does not do anything on its own. Actual business process is done by the sub forms loaded inside. These sub forms have one or more buttons, one with a yellow folder icon which opens a dialog either for file open or folder picker as per the context and the other one with magic wand that calls the actual action.
4. Though, the system can calculate tax at any time and distribute it over remaining months, but you must consider a scenario of calculating and recovering quarterly tax. Here, when you quarterly calculate tax the balance tax would be distributed over 9, 6, 3 and 1 month(s) respectively for respective quarters. Now, this makes the portions of tax recovered in respective quarters per 11.1%, 14.8%, 24.7% and 49.4% of the total tax. The last value may be a value in several thousands, it may become difficult for the individual taxed so heavily during Feb of each year. I, therefore, suggest calculating tax on monthly basis even if you wished to deduct it quarterly. However, I am for deductions on monthly basis so that the load of tax on individual is distributed evenly by just 8.3% of tax per month, even if it means some difficulty in timely deposit with income tax department due to disruptions in banking, which is quite common in disturbed areas like Kashmir Valley, and corresponding large number of UD entries. ‘Tax Deduction’ report is made for this purpose. In case of disruption, the tax deductions would have to be taken into UD register. I recommend instead of entering the entire list in UD, user may add just one entry as per the list printed using ‘Tax Deduction’; the list be signed on each page and placed in a file specifically opened for this purpose.

## Reports

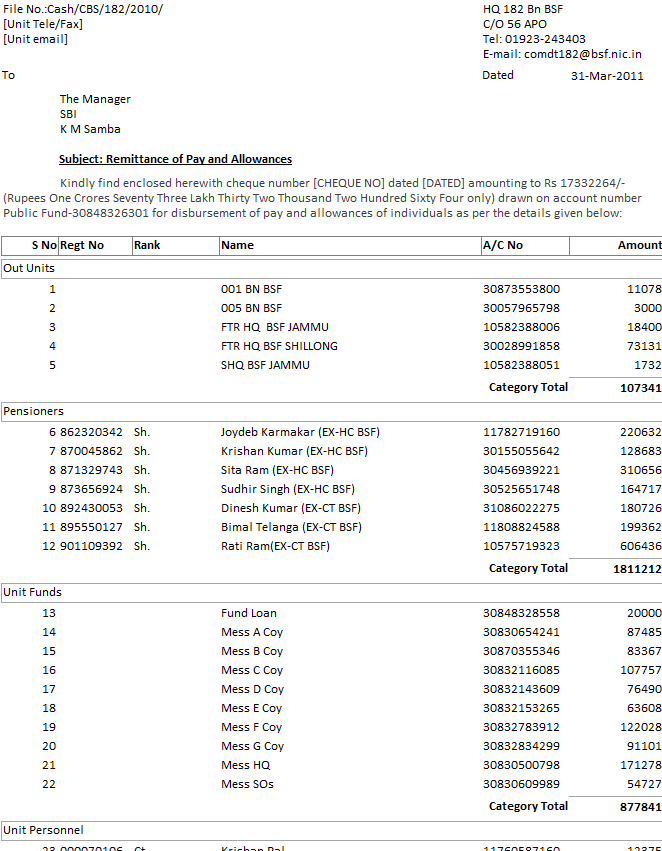
1. A system is incomplete if it is not able to provide outputs for user to disseminate processed information for all concerned and to keep certain details as record of the process.
2. All the reports in the system can be printed on normal printers as usual, as well as can be saved as PDF files for digital recordkeeping. To save a file as PDF, just double click on the report in any view other than design. Though, I successfully tested this PDF output feature on a few computers supplied in ‘Prahari’ in my unit, but it may fail in certain unforeseen circumstances. In case of such failure, I would suggest you install an application that provides you with a virtual PDF printer and then print the report to the virtual printer so installed. One lightweight, no frills, no malware application is named ‘doPDF’, which is free and can be easily downloaded from <http://www.dopdf.com/> website. However, it scrolls ads on the print dialog.
3. As of now, following reports have been designed in the system for this purpose. The reports are described one by one as under:

### Annual Tax



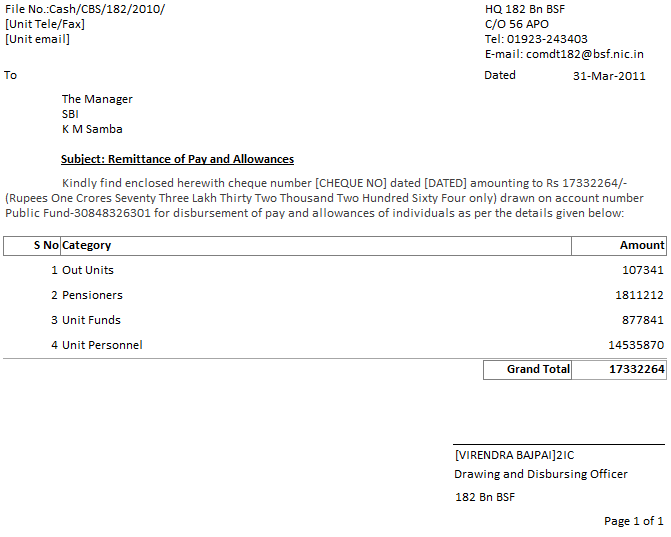
1. The report is straightforward. It shows monthly tax deductions for all individuals. Not much required, though.

### Bank Letter



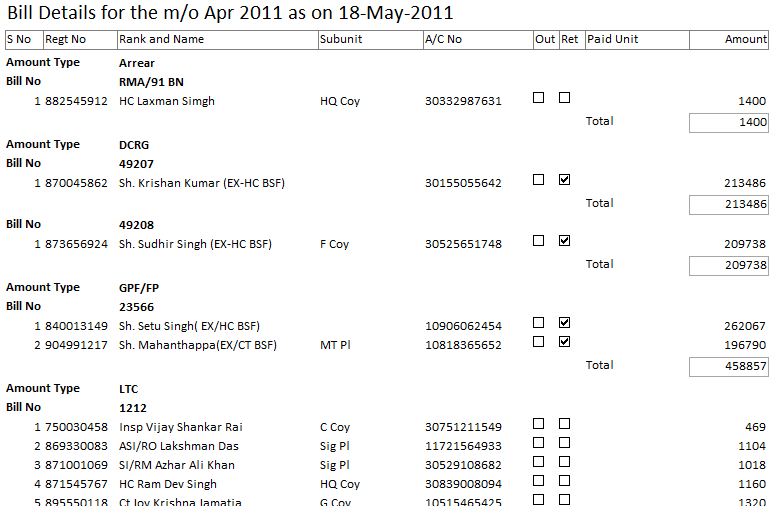
1. This report is designed to be printed for bank. As you can see, the report groups payments in four different groups, namely ‘Pensioners’, ‘Posted Out/Attached Personnel’, ‘Unit Funds’ and ‘Unit Personnel’, for ease of spotting a payment. I would recommend a single print of the report which is given to the bank along with the cheque.
2. When you open the report, it prompts you to enter ‘Cheque No’ and ‘Dated’ values to be printed on the letter of forwarding part of the report. If you just press Enter, the report will be opened without these parameters for your viewing, however, it is suggested that on printing the report you must provide correct values to these parameters as you wish to be printed so that proper letter format is created for bank.

### Bank Summary



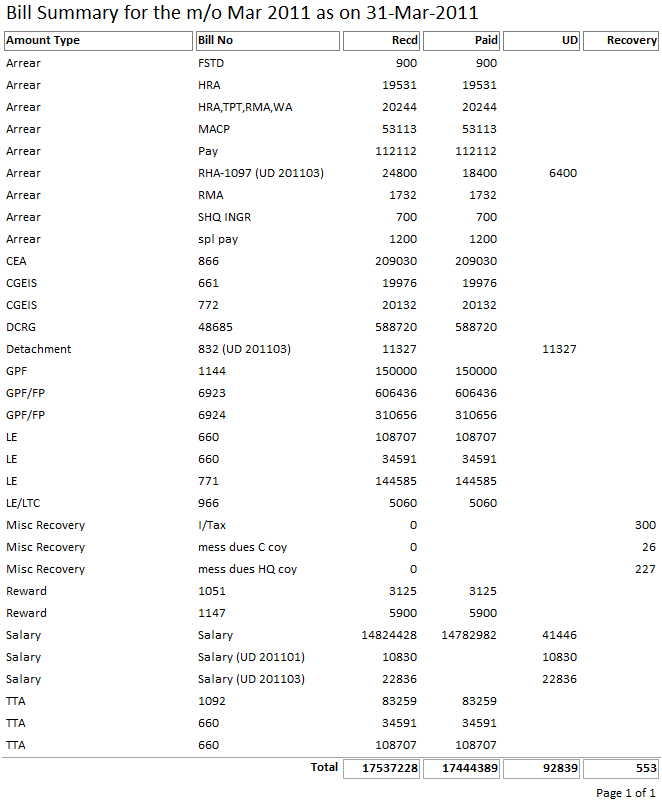
1. This report is designed to do away with duplicate printing of ‘Bank Letter’. It processes the same data for arriving at correct category totals, yet it hides the details to minimise stationary consumption. Since, amount to bank is mentioned elsewhere also, and thus, preserving full ‘Bank Letter’ was felt unnecessary for record.

### Bill Details



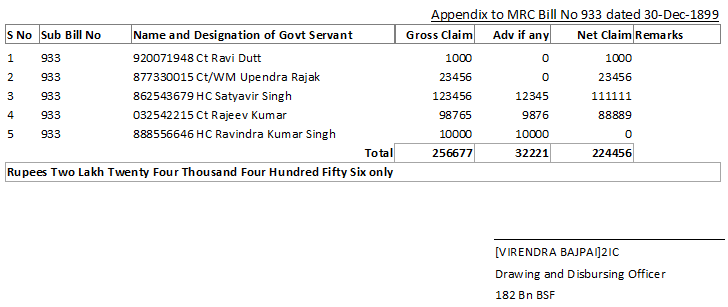
1. The report is designed to list personnel according to ‘Amount Type’ and ‘Bill No’ for checking of individual bill.

### Bill Summary



1. The report is designed to be kept as record in Cash branch. The report shows ‘Recd’, ‘Paid’, ‘UD’ and ‘Recovery’ columns for each bill to correctly arrive at the totals in the end.

### Bills Sent

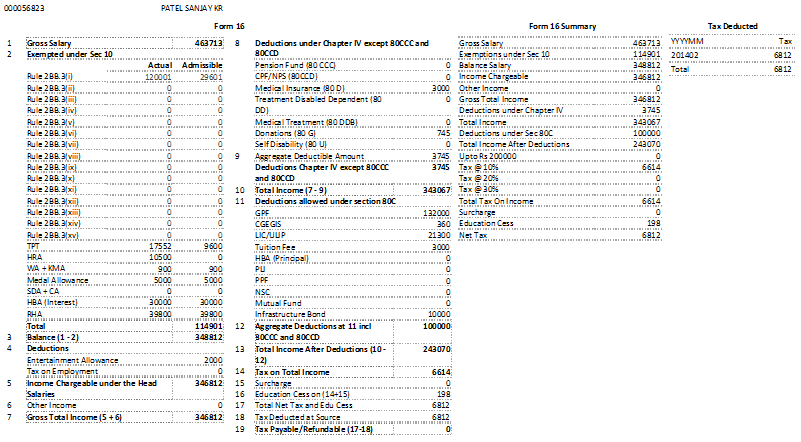
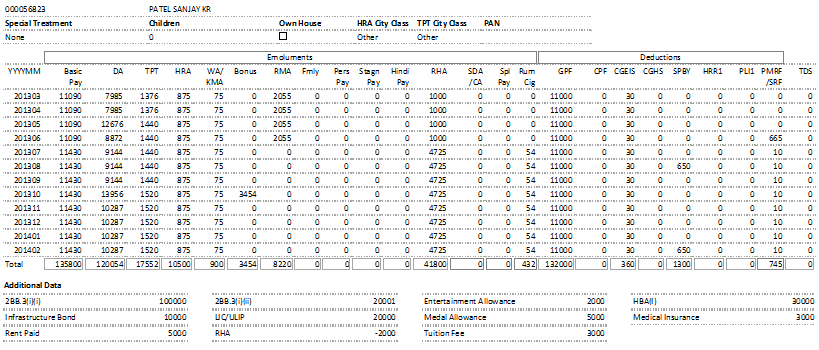


1. The report is designed for Accounts branch for sending bills to PAD. The report generates the tabular part of pro-forma commonly contained in the sanction orders. It is intended to write ‘as per list attached’ at appropriate place in the sanction order and this list be attached to that. While this may be able to reduce some work of the branch, it may alternatively be exported to MS Word using the option as shown below. Once the text is available in word to the user, it can be easily included in the main sanction order using copy and paste.



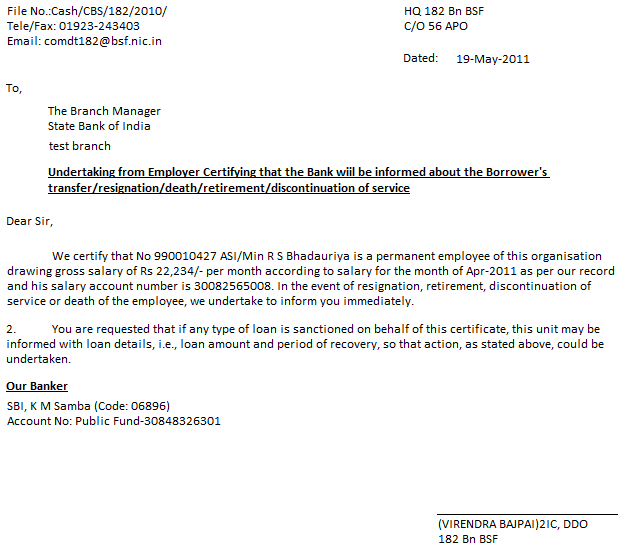
1. The benefit of this feature of the application to Accounts branch is that it reduces the data-entry efforts in the tabular portion of the sanction order. However, the main benefit is achieved by Cash branch in speedy payment to individuals. I would like to remind you that men working in Accounts branch will also be benefitted by this in the end.

### Calculation Sheet



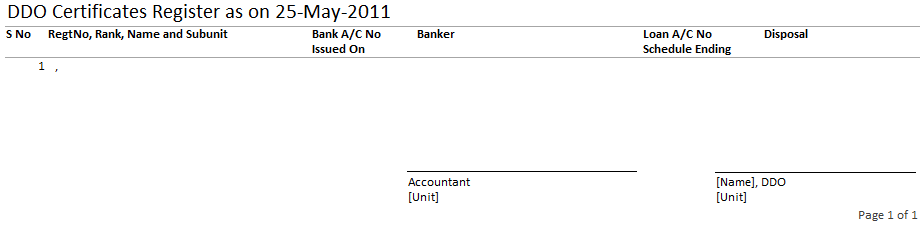
1. This is a two-page report for every individual is intended to be printed back to back and kept as record replacing any register maintained for the purpose. It shows data received from IT Wing and locally provided data (Data and Recoveries) on one page, and monthly tax recoveries, part of Form 16 and Form 16 summary on the other. This report would be needed for audit purpose.

### DDO Certificates



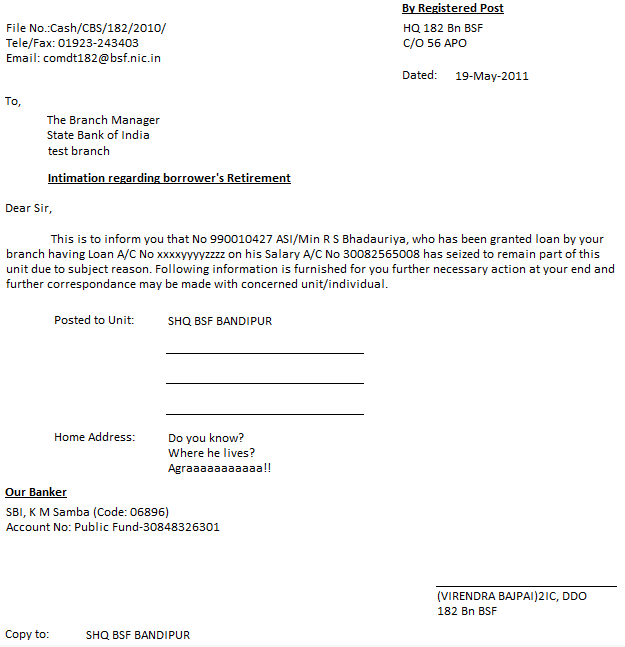
1. The report prints the format of the requisite certificate for individual. It is supposed to be printed using the form capturing records to this effect.

### DDO Certificates Register



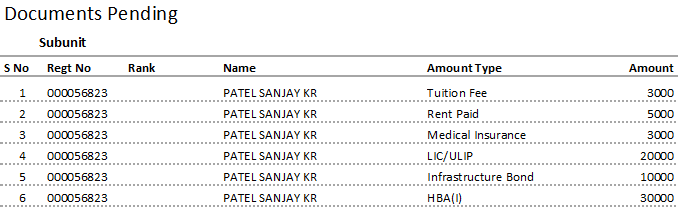
1. The report is designed to print a list of men issued with the DDO certificates for record keeping.

### DDO Intimation to Bank



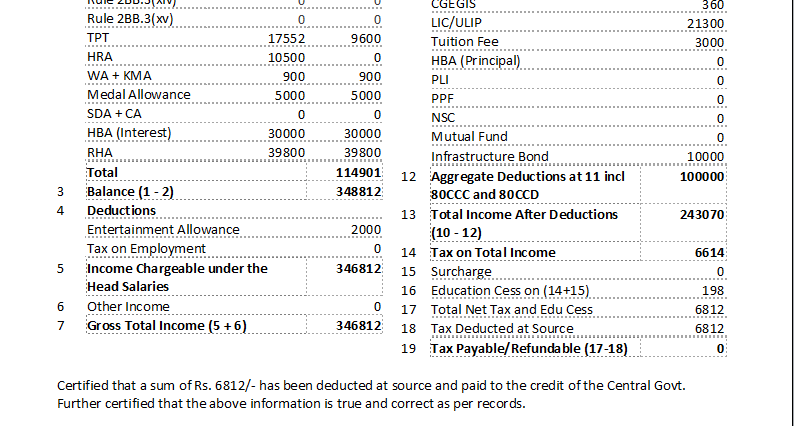
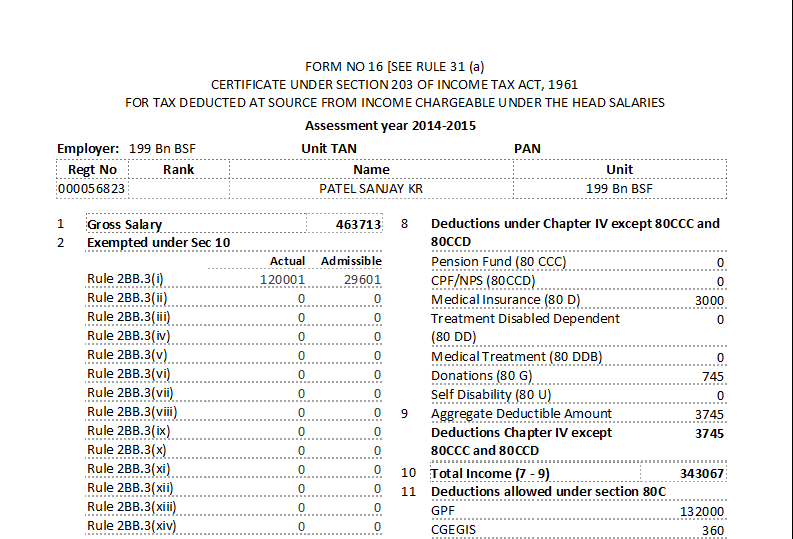
1. In the event of clearance of the individual, the cashier should look at the table to locate a record in this regard. If one is found, the report is required to be printed for intimation to bank and for unit to which he is posted.

### Documents Pending



1. This report gives subunit wise listing of pending documents against which data has been entered on receipt of undertaking by individuals.

### Form 16



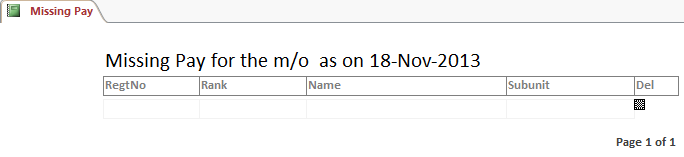
1. This report is designed to be printed for everyone at the end of the financial year. The sequence of pages is in order of subunit followed by Regt No to suite grouping of personnel in batches of subunits. However, in conjunction with ‘Selective Form 16’ form, selective ‘Form 16’ of an individual can be printed. I would recommend sending this print to the unit of the personnel posted out so that the unit may be able to incorporate actual TDS in whatever way they are doing tax deductions. In your case, you may, however, add an entry in the name of ‘TDS’ as ‘Amount Type’ in ‘Data’ table.

### Messing Recovery (Coy)



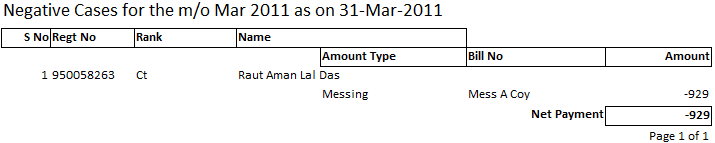
1. The report prints messing details of individual amounts in two separate groups named ‘Recovered’ and ‘Not Recovered’. Since each mess starts printing on odd page, the report may be printed back to back. Cash branch may not require its print in duplicate.

### Missing Pay



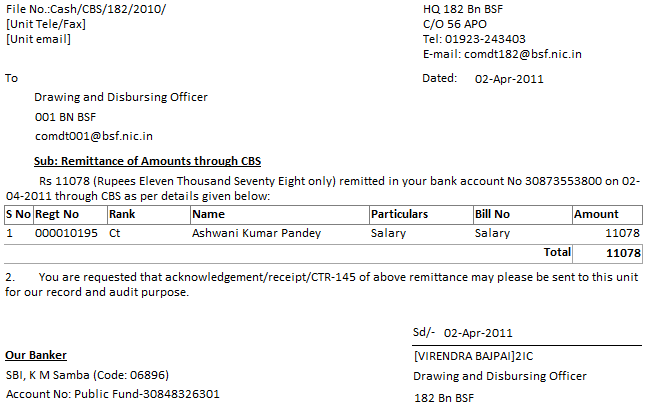
1. The report is designed after feedback from the participants during workshop conducted on 12th and 13th Nov 2013 at Teliamura. During ‘Imports’ with ‘Salaries’ selected, this report is automatically opened to give intimation in respect of personnel posted if one has not received his salary. I feel it serves the purpose by intimation and may not be required to be printed and kept as record.

### Negative Cases



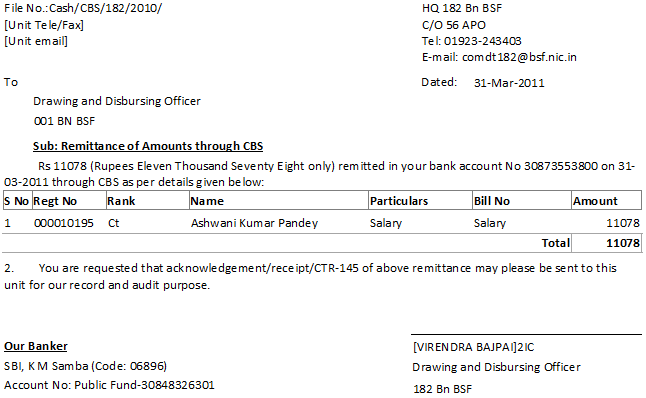
1. While payments and recoveries are effected using the system, it was felt possible to have recoveries more than payments made to an individual. In such cases it becomes impossible to recover complete amount required to be recovered; besides it could play havoc with the simple arithmetic of the system.
2. Name of the report is inspired by the similar report of IT Wing that was in use during my tenure. This report lists such cases with complete details of payments and recoveries due for the individual. User may then decide to keep certain recoveries pending or retrench a few of them to recover in instalments.

### Out Unit Emails



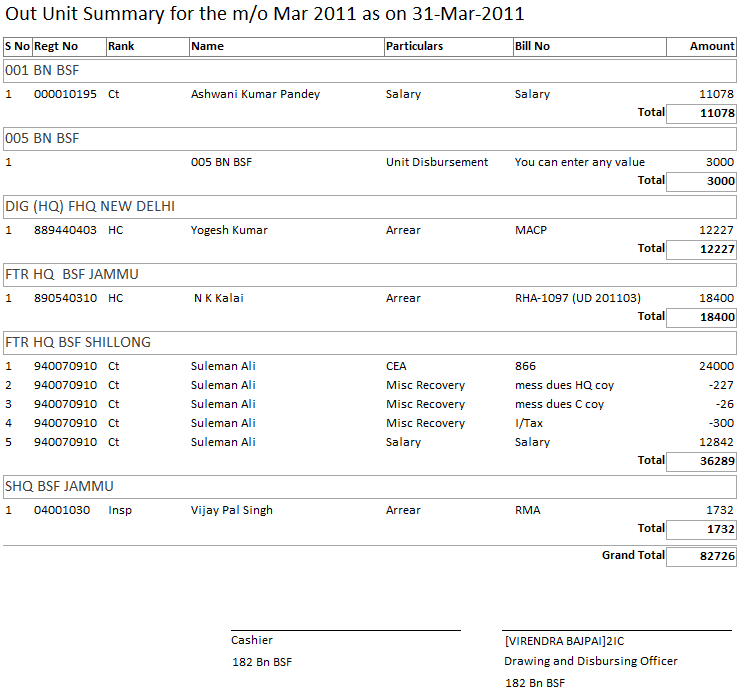
1. The report is identical to ‘Out Unit Letters’ except unit email address in addressee part and ‘Sd/- [Date]’ in the signature space. The report is intended to create a PDF to be attached in an email sent to the concerned units. Aha! This reduces expenditure on postage and envelopes also. Going green.

### Out Unit Letters



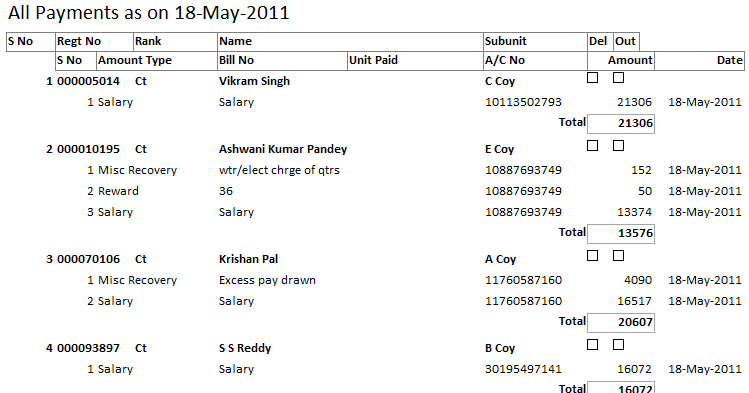
1. Use this report in case emails are not used. This report is designed for dispatch to units, where certain personnel have been posted or are attached. The system credits the amounts in the accounts of units, instead of individual accounts, for disbursement through units. The addresses of these units have not been catered for in the system to reduce data-entry job of Cash branch. The dispatch relies upon maintenance of correct unit addresses with unit despatcher.
2. Since all the units have official email addresses, it was felt that by using email for dispatch of report would further save on paper, envelops and postage charges. However, it was also felt that for certain unforeseen reasons, it may not be possible to send the report through email. Thus, while allowing the report to be viewed and printed normally, report’s ‘double click’ event was trapped to save the report to PDF file. If you want to send the report through email to concerned units you are advised to do the following:
   1. Double click on the report to get file save dialog appear. Now save the file with desired name at desired location. I would suggest you give traces of unit and date of transaction in the file name so saved.
   2. Open your email account and ensure that sent emails are saved. This may be one time only process if the email client settings are not generally disturbed.
   3. Send email to all concerned units attaching the PDF file saved at ‘a’.
   4. Open sent mail folder, open the mail you had just sent and take one printout of the email sent to concerned units. This print may be kept as a record of dispatch.
   5. If for some reasons, a few odd units could not be included while sending the email, selective pages of the report can be printed and dispatched through normal dispatch channel.

### Out Unit Summary



1. This report summarises unit wise payments made during the process. I would suggest printing of only one copy of letters for dispatch while dispatch numbers can be endorsed against units in this report by hand in case normal dispatch channel was used. In case the report was sent through email, the copy of sent mail may suffice as record of dispatch.

### Payment History



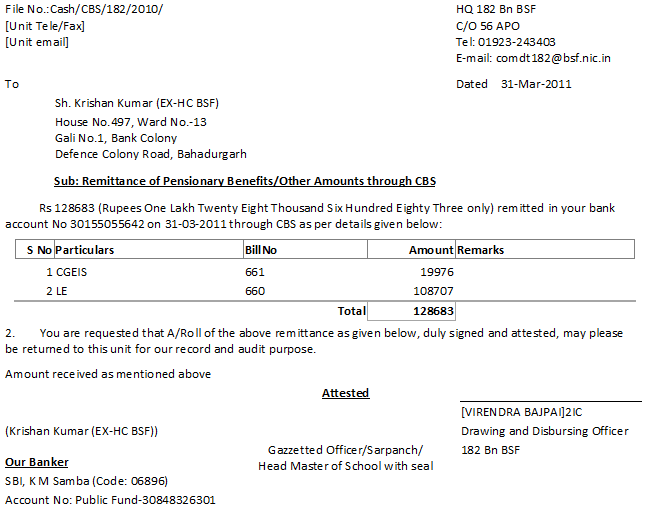
1. As discussed under ‘Payment History’ table, this report shows individual wise, amount type wise, bill number wise payments and recoveries using ‘Pay \*’ interfaces, i.e., leaving ‘Messing’ and ‘RMA’ recoveries. The data contained in the report is from the last ‘Initialize Tax’ action up to the last process. Please note that the data from current process is not included and it will be included during next ‘Initialize’ process. Since, the date of payment is taken from the system date during ‘Initialize’; it is advised to ‘Initialize’ new database on the day of transaction itself or you might prefer to fiddle with system date later.
2. In case a person is posted in another unit and paid some amount or a recovery from in unit person is made through ‘Pay (Other Unit Recovery)’ form, the records would show account numbers of the concerned unit along with unit paid as per actual transaction instead of account number of individual.

### Pensioner Labels



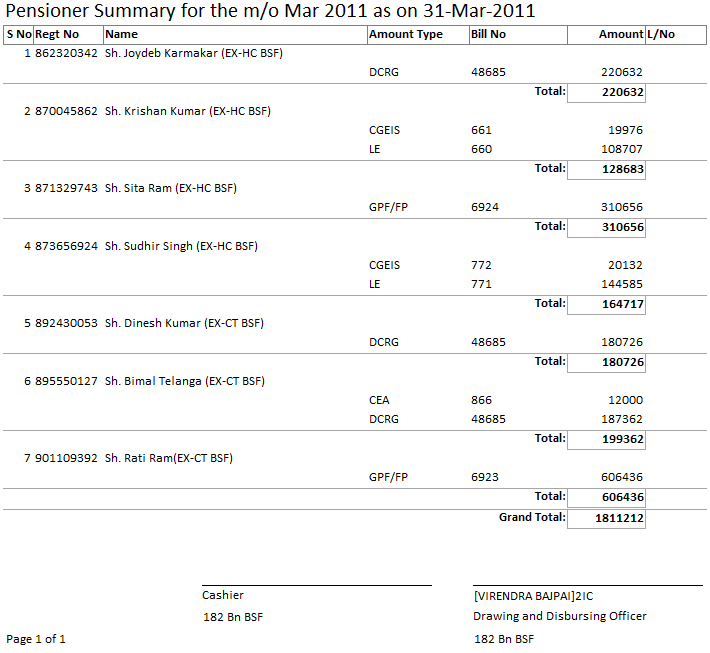
1. The report generates labels (multiple labels per sheet) for assistance to the despatcher of the unit. These labels can be cut and pasted on the envelopes for dispatch of letters.

### Pensioner Letters



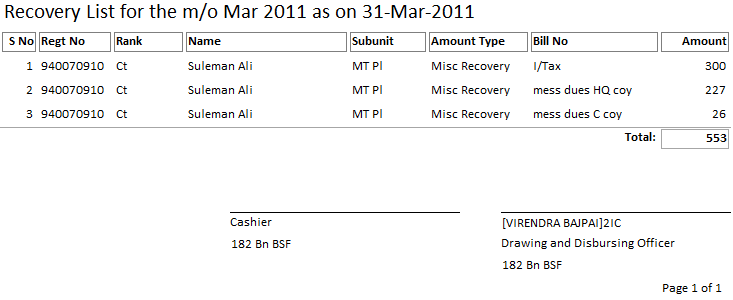
1. The report can be used to print individual letters of intimation for the payments made or recoveries effected in respect of individuals retired. The report generally takes half sheet of A4 paper (A5 size), thus, you may use properly cut stationary for its print.
2. To help dispatcher, a report named ‘Pensioner Labels’ has been created to print address labels for only those retirees that are included in the given transaction.
3. The system has yet another report named ‘Pensioner Summary’ which is described next. I would recommend that just a single copy of the letters is generated for dispatch, while letter numbers may be enumerated in front of the records of individuals in the ‘L/No’ column of ‘Pensioner Summary’ report.

### Pensioner Summary



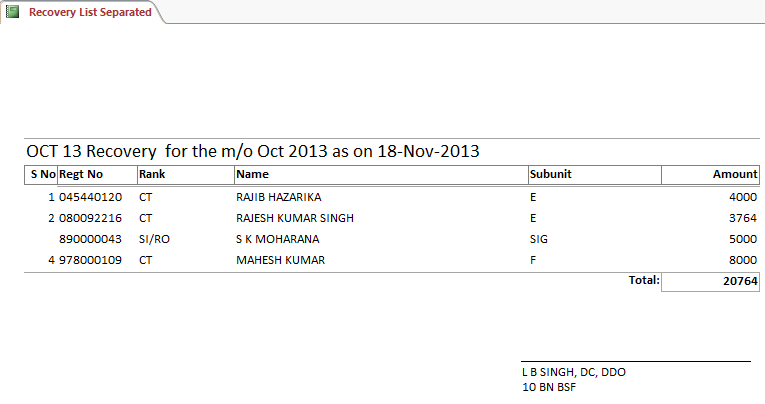
1. The report includes details of payments, recoveries and individual wise total in the transaction for each individual retired. It is suggested that the register maintained for retiree remittance may be discontinued and the printouts of the report be stitched at the end of financial year to form the register.

### Recovery List



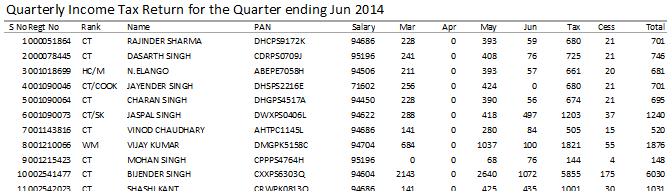
1. Though, recoveries are reflected in other reports generated through the system, this report clubs all the recoveries from individuals, effected through ‘Pay\*’ form, in one place. The report is suggested for record at Cash branch.

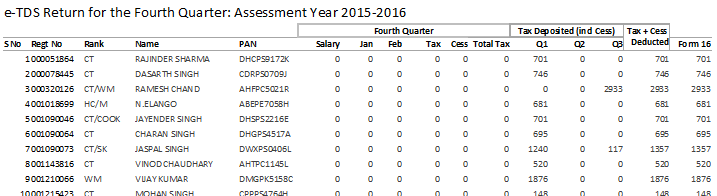
### Recovery List Separated



1. The report is designed to distribute the recoveries to individual branches effecting the recoveries. It works in conjunction with the table with the same name, wherein, bills may be selected for appearance here. Each ‘Bill No’ starts on odd page, therefore, the report can be printed back to back.

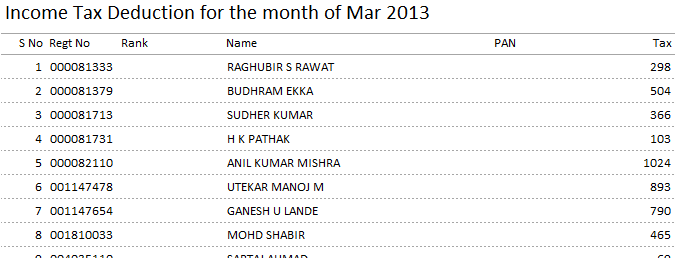
### Tax Return \* (Q1-Q4)





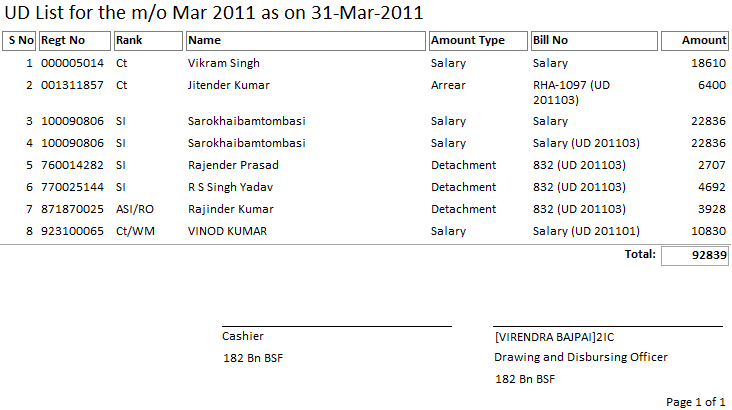
1. These reports may be handy when a return is required to be filed at Income Tax Department. This return of 4th Quarter includes data for other Quarters also for annual return filing.

### Tax Deduction



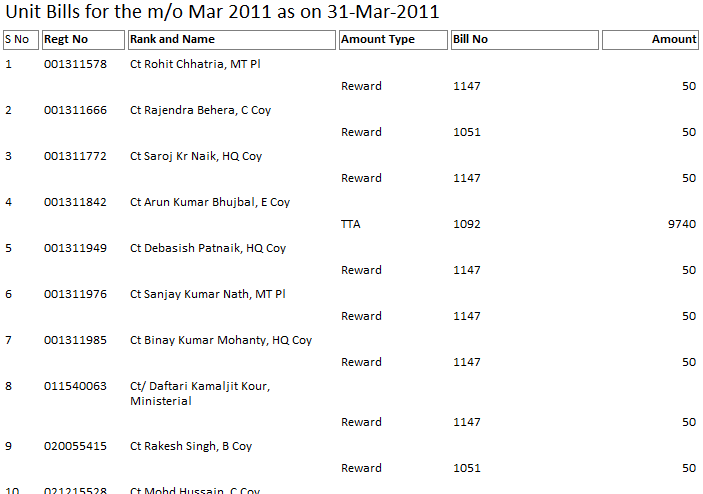
1. This report may be useful in depositing tax in concerned branch of bank and while keeping deducted amount as UD if needed.

### UD List



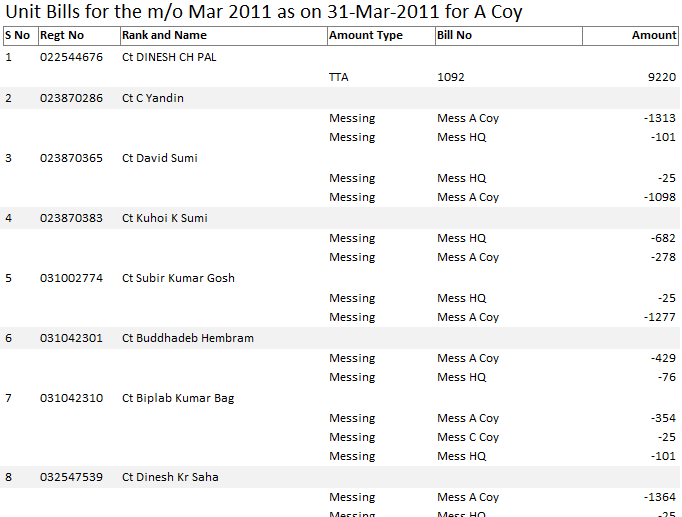
1. The report is designed to provide a list of ‘UD’ amounts. Though, it is not supposed to stop the ‘UD’ register in its present shape, yet it would help in carrying forward ‘UD’ Amounts in the register in respect of salary/bills easily.

### Unit Bills



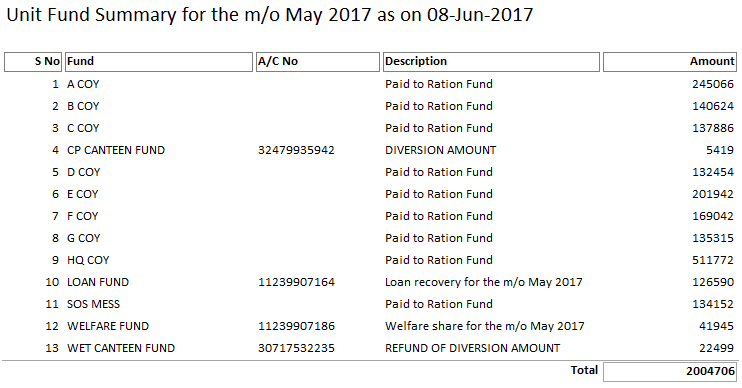
1. The report is supposed to be opened after ‘Separate Emolument’ form is used to decide, which emolument is needed on summary reports and is excluded here.
2. The report is designed for record of Cash branch. Since the payments shown in the ‘Bank Summary’ are sorted on ‘Regt No’ where showing the ‘Subunit’ of individual was wastage of print material, the report is designed to have the same order for easier matching during the audit. If subunit wise report is kept in record instead of this report, without printing subunit in ‘Bank Summary’, the desired matching would be a bit difficult in my opinion.

### Unit Bills (Coy)



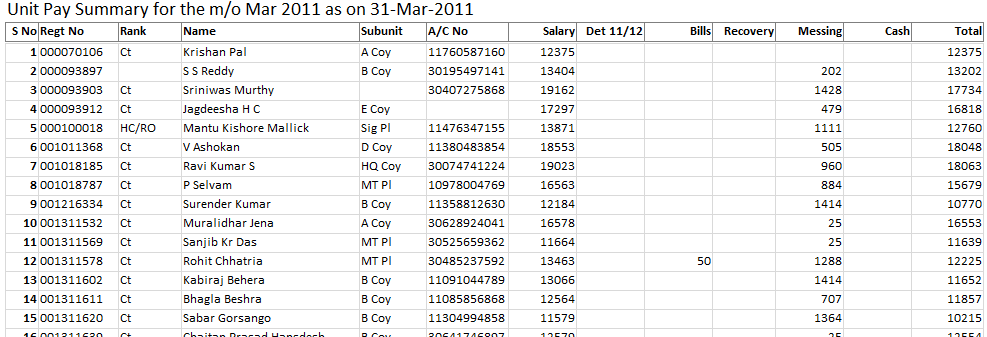
1. The report is supposed to be opened after ‘Separate Emolument’ form is used to decide, which emolument is needed on summary reports and is excluded here.
2. The report was originally designed for distribution to the subunits for intimation to individuals in respect of payments made to them except ‘Salary’, user defined ‘[Separate Emolument]’ and ‘Cash’ and in respect of recoveries other than ‘Messing’. However, it was felt later that such system leaves individual guessing about his ‘Messing’ recoveries. Even if ‘Ration Recovery Details’ is printed and distributed to ‘Messes’ it would not have helped individual in this regard. Thus, the report is modified to include ‘Messing’ recoveries from everyone. The report includes ‘Messing’ for only those personnel from whom recovery for more than one ‘Mess’ is available in the database as in case of single ‘Messing’ recovery, data listed in ‘Unit Pay Summary (Coy)’ would suffice.
3. I would suggest printing of only one copy of the report for distribution.
4. A separate report based on ‘Regt No’ has been designed with the name ‘Unit Bills’ for its record at the Cash branch. The report does not include ‘Messing’ to save stationary.
5. This report is designed to work in conjugation with ‘Unit Pay Summary (Coy)’ as the payments and recoveries omitted in this report are part of the other one.

### Unit Fund Summary



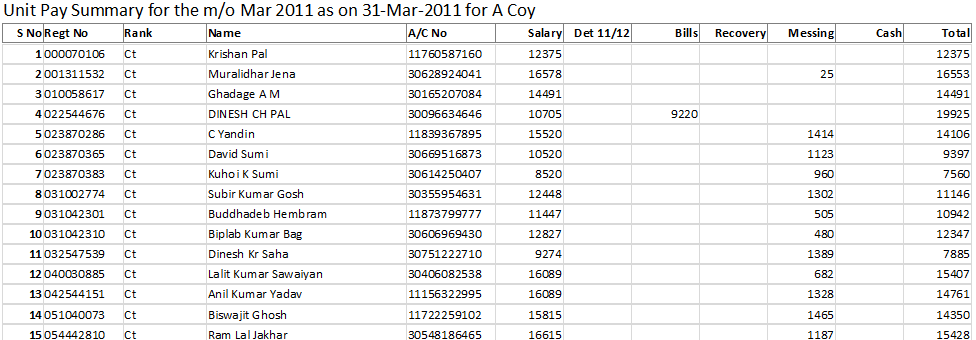
1. The report is designed to be distributed among the listed funds/messes for their record. Cash branch need not keep its copy as these payments are included in the ‘Bank Summary’. The report is designed to conserve some ink or paper while distributing prints for records. The expected size of print stationary is A5 (half sheet of A4 paper). As discussed under ‘Messing’ the amount recovered against each mess is paid to ‘Ration Fund’ as suggested by audit.

### Unit Pay Summary



1. The report is supposed to be opened after ‘Separate Emolument’ form is used to decide, which emolument is needed on bills reports and is excluded here.
2. The report is designed for record of Cash branch as summary of payments made to individuals in conjugation with ‘Unit Bills’. The report clubs all bills paid to individual under ‘Bills’ column leaving aside values for ‘Amount Type’ mentioned against ‘Separated Emolument’ taking its heading from the value against ‘Separated Emolument Title’. Recoveries and Messing are also clubbed under respective columns.
3. A single copy of the report is required to be printed for record, while Cash branch should distribute contemporary ‘Unit Pay Summary (Coy)’ for distribution.

### Unit Pay Summary (Coy)



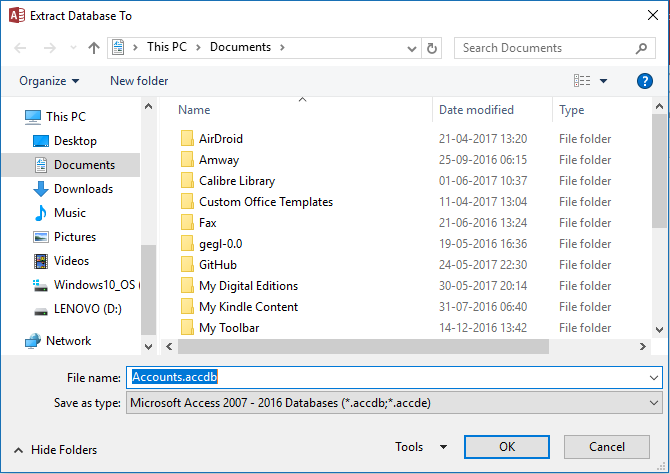
1. The report is supposed to be opened after ‘Separate Emolument’ form is used to decide, which emolument is needed on bills reports and is excluded here.
2. The report is designed for information to individuals as summary of payments made to them in conjugation with ‘Unit Bills (Coy)’. The report clubs all bills paid to individual under ‘Bills’ column leaving aside values for ‘Amount Type’ mentioned against ‘Separated Emolument’ taking its heading from the value against ‘Separated Emolument Title’. Recoveries and Messing are also clubbed under respective columns.
3. A single copy of the report is required to be printed for distribution to the subunits for their record, while Cash branch should keep contemporary ‘Unit Pay Summary’ for its record.
4. I would suggest you stop maintaining register for disbursement at subunit level, while ‘Unit Pay Summary (Coy)’ together with ‘Unit Bills (Coy)’ may take its place.

# External Extensions

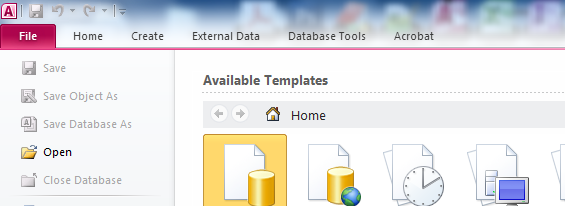
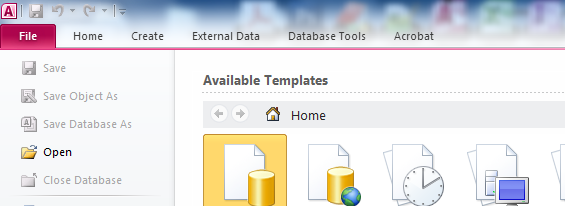
1. The application uses few external application extensions to extend the functionality of the system without overstretching the main application. Following extensions are designed so far:

## Accounts: Password Protected Access Database

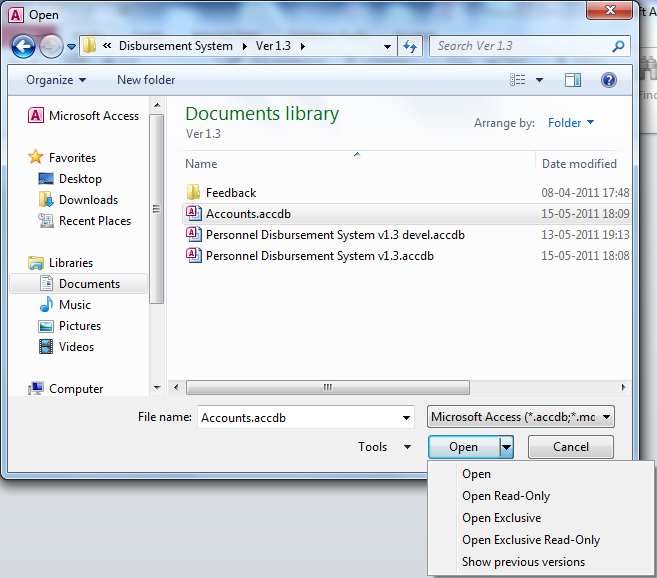
1. Having decided to use the current version of the database, this extension becomes mandatory. The database has only one table to store the account information of everyone. Though, it may be used without password, but then the cashier may not be able to exercise effective control over account number changes. The steps towards the use of this extension to password protect are as under:
2. When you first use this database, you would be asked for a location to extract the database as shown below:



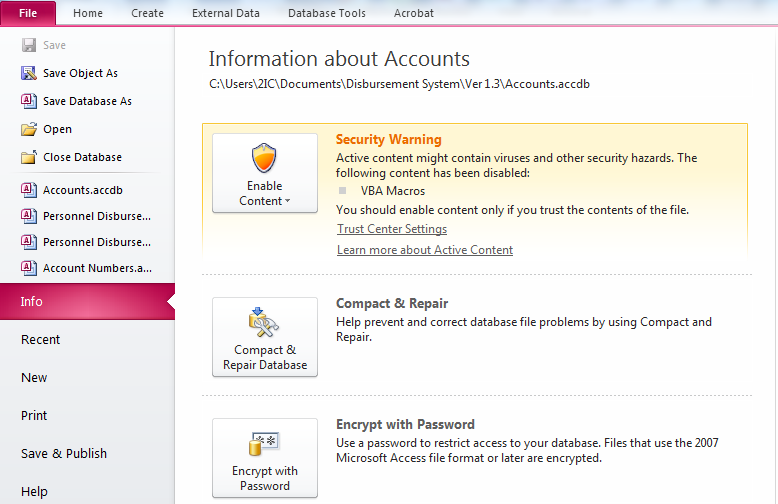
1. You can create it at any place of your choice and with any name. The database would be extracted and will open. You need not enable its contents if shown because it doesn’t have any code to run. Close the database. Next is the series of steps you would need to secure it with password.
2. Open Microsoft Access from ‘Start’ menu and then switch to ‘File’ tab.



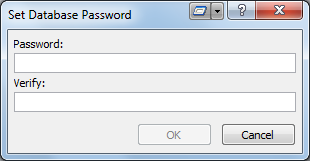
1. Click on ‘Open’. You will be presented with file open dialog.



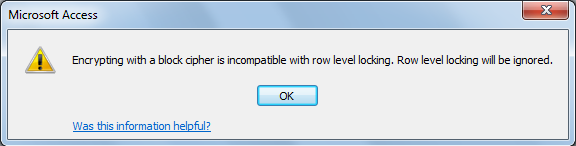
1. Having selected the database, click on the dropdown button at the right of the ‘Open’ button and select ‘Open Exclusive’. Having opened the database in exclusive mode, again go back to ‘File’ tab.



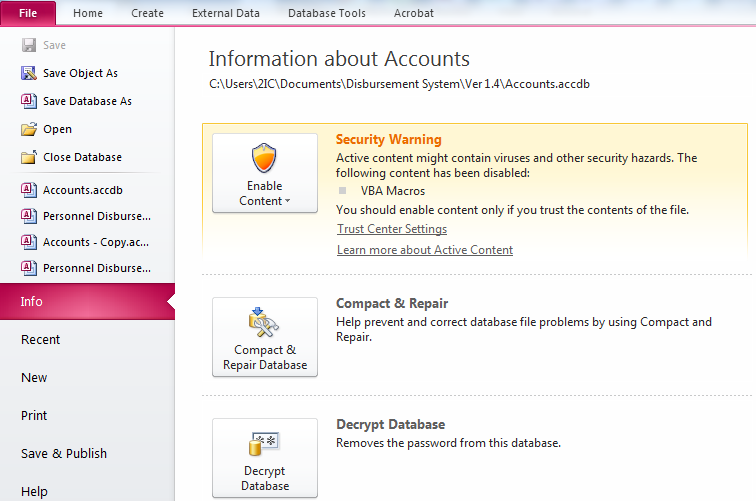
1. Now click on ‘Encrypt with Password’. This will show ‘Set Database Password’ dialog.



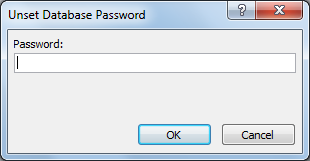
1. Now enter the same password in both the dialog boxes and click ‘OK’.



1. Click ‘OK’ again and close the database before any further use.
2. The database is now password protected. Once the database is used in the process during export, the database together with its password becomes your passport to the ‘Export Trickle File’. Each time you do the export, your passport is renewed for another export. **You must keep a backup of the renewed passport to be able to utilise the system further.** If you lose the physical database, or forget your password, please don’t blame me to your loss. See, I have already warned you.
3. In the similar way, the password can be removed using ‘Decrypt Database’ button on ‘File’ tab once you open the database in exclusive mode as described above.



1. Having clicked ‘Decrypt Database’ button, ‘Unset database Password’ dialog would appear as under.

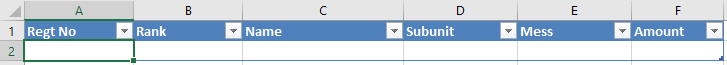


1. Having entered existing password and clicking ‘OK’ button would clear the password.
2. I would recommend that along with keys of the safe, cashiers need to handover/takeover database in decrypted mode so that passwords are not shared. User taking over should now use his own password to encrypt the database again.
3. If at all you need to unlock the database without ‘Accounts’ database, you need to read code carefully to know the procedure to bypass, but I would better not document it here.

## Excel Templates

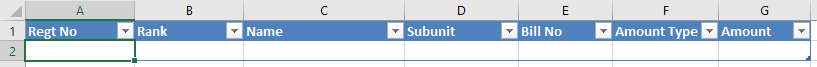
1. As I have already narrated the plight of the NCO who collects water and electricity charges from various individuals in my unit, some units have chosen to rest the onus of depositing the amount on individuals. This requires maintaining of record manually and later depositing the collection in Govt. Treasury.
2. Also, with the knowledge that many a times requisite data are available in soft copy format elsewhere, I tried to design certain templates in Excel so that the data is made available to various import procedures in the system to import and avoid any duplicity of efforts. These templates have been discussed in corresponding import procedures earlier, however, I would like you to present the structure here once again.

### Messing



1. This template can be used for import messing using ‘Imports’ with ‘Messings’ selected.

### Pay



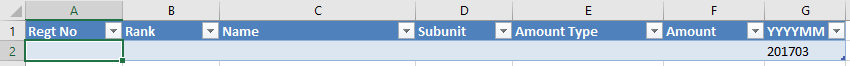
1. This template can be used for import of ‘Cash Withdrawal’, ‘Payments’ and ‘Misc Recoveries’ using ‘Imports’ with ‘Cash Withdrawals’, ‘Payments’ and ‘Recoveries’ selected respectively. You must note that in case of ‘Cash Withdrawals’, ‘Amount Type’ and ‘Bill No’ are both ignored, while in case of ‘Recoveries’, ‘Amount Type’ is ignored. I suggest you to wisely use ‘Bill No’ to describe actual type of recovery (short and sweet).

### Personnel



1. This template can be used to import data of personnel posted in the unit using ‘Import Personnel’, while at the same time it can be later used to update PAN of individuals using ‘Tax Wizard’ at its ‘Update PAN’ stage.

### Tax Data



1. This template can be used to import tax ‘Data’ as well as ‘Recoveries’ using ‘Tax Wizard’ at ‘Import Payments Tax’ and ‘Import Recoveries Tax’ stages respectively.

# Summary of the Process

## Onetime Process

### New Users

1. **Step 0.1**: Open the signed & packaged database and save it with a name, for example, ‘Master’. This database would not be required after step 0.5 and should be deleted.
2. **Step 0.2**: You will be presented with the ‘Start-up’ form; however, you might require enabling the contents. Having enabled the contents, the ‘Initialize’ form for initialization from previously working database would be presented. You should ignore and close this form for now. Opening the database second time onwards, it would present you with the ‘Configuration Settings’ form for customisation of the application according to user unit, however, you may open the form manually yourself. You are required to fill in the values in this form and change the value of ‘Setup done’ variable to value ‘Yes’.
3. **Step 0.3**: You are now required to enter data in the tables listed under Masters Level 1 and 2. While this can be done through various interfaces provided, an easier workaround has been provided for hierarchy to ‘Personnel’ table, i.e., ‘Ranks’, ‘Subunits’ and ‘Personnel’. To achieve this entry, prepare data in accordance with the Excel template provided with the name ‘Personnel’ and then import it into the system using ‘Import Personnel’ form. Since, the system highly depends upon correctness of 9 digits ‘Regt No’; care must be taken to make the list in accordance to its value in central pay system. The correctness can be tested by importing one salary downloaded. Having imported the salary, you would be presented with ‘Personnel Posted Incomplete’ form to show incomplete records. I would recommend you open ‘Pay’ table and delete all records at this stage. Now, if any incomplete record corresponds to any existing record with incorrect ‘Regt No’, you can copy correct ‘Regt No’ from this form and delete the record, search for person in your own data on ‘Personnel Posted’ form and then paste it replacing wrong/incomplete ‘Regt No’ field. Do this for each record shown on form for incomplete records.
4. **Step 0.4**: Complete data entry in ‘Amount Types Pay’, ‘Funds’ and ‘Messes’ tables.
5. **Step 0.5**: Create two more copies from the packaged database as at Step 0.1, one each for Accounts and Cash branches. I would suggest naming these databases as ‘BRANCH yyyymmdd’, where ‘yyyymmdd’ stands for year-month-date. Initialize both these copies of databases from the database named ‘Master’ created at step 0.1. Distribute branch databases to corresponding branches. This is done for the first time only. Now the database named ‘Master’ becomes redundant and should be deleted. Further steps are required to be completed in cash branch.
6. **Step 0.6**: Open ‘Personnel Tax’ and correct values of remaining fields especially the ‘Special Treatment’.
7. **Step 0.7**: Open ‘Amount Types Tax’ and map ‘Amount Types’ from ‘Amounts Types Pay’ table.
8. **Step 0.8**: Extract and password protect ‘Accounts’ database.

### Users of older split-database subsystems

1. **Step 0.1**: Open the signed & packaged database and save it with a name, for example, ‘Master’. This database would not be required after step 0.4 and should be deleted.
2. **Step 0.2**: Open ‘Import Process’ form and select database for last process of pay subsystem and import it. This will populate tables that are available in older pay database.
3. **Step 0.3**: Open ‘Import Process’ form once again and select database for tax subsystem and import it. This will populate tables that are available in older tax database and will also update ‘Personnel’ table for additional fields required for tax subsystem.
4. **Step 0.4**: Create two more copies from the packaged database as at Step 0.1, one each for Accounts and Cash branches. I would suggest naming these databases as ‘BRANCH yyyymmdd’, where ‘yyyymmdd’ stands for year-month-date. Initialize both these copies of databases from the database named ‘Master’. Distribute branch databases to corresponding branches. This is done for the first time only. Now the database named ‘Master’ becomes redundant and should be deleted.
5. **Step 0.5**: In personnel table correct the values for children in earlier system ported to CEA that may be needed to be in ‘Hostel Subsidy’.

## Main Process

1. **Step 1**: Accounts branch to keep on sending bills from their copy of the database. They may also keep on issuing DDO’s Certificates. Cash branch to keep on entering data in the process other than bills being entered by Accounts branch. For Cash branch, this would involve following:
   1. Entry of bills (not entered by Accounts branch), recoveries and cash withdrawals in the ‘Pay’ table. This may involve import of recoveries submitted by branches of the unit. ‘Pay’ template can be utilised in case data in digital format is available elsewhere.
   2. Entry of messing and RMA recoveries in ‘Messing’ table. At least HQ mess can be advised to submit messing recoveries in ‘Messing’ template.
   3. Feed any money transfer to any other private fund in ‘Funding’ table, while to any other unit in ‘Unit Disbursement’ table.
2. **Step 2**: Once pay processing is done by IT Wing, cash branch is required to download ‘Salary for Import’ report in Excel format and import it in the system using ‘Imports’ with ‘Salaries’ selected. In case a record for ‘Personnel’ is created during the import, you will be shown ‘Personnel Posted Incomplete’ form. If the personnel so added happens to exist in the database with wrong ‘Regt No’ you may require to cautiously undertake few steps mentioned in the documentation for the form or at **Step 0.3** above.
3. **Step 3**: Cash branch to import bills from the copy of database of Accounts branch using ‘Imports’ with ‘Bills’ selected and then include bills passed in its disbursement process through ‘Bills Received’ interface.
4. **Step 4**: Cash branch to select bills to be printed in ‘Recovery List Separated’ through form with same name.
5. **Step 5**: If the salary is part of current transaction, Cash branch is advised to download ‘NGO’s Annual Income Tax Sheet’ from IPP in CSV format. Run the ‘Tax Wizard’ form and complete the calculation for the month so that income tax is also deducted.
6. **Step 6**: Having completed the transaction entries, open ‘Negative Cases’ report to check if an individual is imposed recoveries in excess to the payments made to him as in such cases complete amount required to be recovered cannot be recovered. User will have to decide to keep few recoveries pending by marking as ‘UD’ or ‘Not Recovered’ or may decide to recover certain amounts in parts.
7. **Step 7**: If the salary is part of current transaction, cash branch to use ‘Separate Emolument’ form to decide the longest bill to be separated to ‘Unit Pay Summary\*’ reports.
8. **Step 8**: Cash branch to print required reports and match for any controls involved.
9. **Step 9**: If salary was part of transaction, print ‘Tax Deduction’ report to deposit the TDS. If it happens to be end of quarter also, print ‘Tax Return Q\*’ report of that quarter for filing return.
10. **Step 10**: Cash branch to ‘Save Reports to PDF’, print a copy each of ‘Bank Letter’, ‘Bank Summary’ and then create the bank-file using ‘Export Trickle File’ interface. This would require approving of bank accounts with ‘Accounts’ database. Write the bank-file on a CD or any other media used by bank and complete the transaction. Had there been tax deduction, complete the tax part of procedure also.
11. **Step 11**: On completion of the transaction successfully with bank, cash branch to ‘Initialize’ new database on the same date as the history of payments is stored with the date of initialization as the date of payment. If the salary for the month of February was just over, ‘Initialize Tax’ also so that new tax process is setup.
12. **Step 12**: Cash branch to provide a copy of ‘Bill Summary’ to accounts branch. Accounts branch to mark received bills using ‘Bills Received’ interface from the ‘Bill Summary’ and initialize their new database.
13. **Step 13:** The database in use immediately before these initializations now become redundant and may be deleted or stored as archive (in CD, etc.) to save disk space and more importantly file clutters. Your record of process in PDF files is in the folder you selected during ‘Save Reports to PDF’, which may also be backed up for record.

## General Guideline

1. Updates were originally designed to be included at initialization stage at the end of the process, where older packaged database will be replaced by new packaged database or you may choose to continue the same process in the new application using ‘Import Process’ instead of ‘Initialize’.
2. As Accounts and Cash branches would be using different copies of the database, responsibility of keeping the data in sync in the tables especially mentioned under Masters Levels 1 and 2, after the one-time process is over, rests with the two branches. For the database ‘LE’ and ‘L E’ are as different as ‘LE’ and ‘MRC’. Similarly, ‘Inspr’ is also entirely different from ‘Insp’. Thus, care must be taken in entering any data in these masters and you should not forget to get it added in the database of the other branch. I would even like to advise you to arrive at a common value before it is entered if no standard abbreviation is available. Though, Accounts branch needs only ‘Regt No’, ‘Rank’, ‘Name’ and ‘Subunit’ fields to be completed in the ‘Personnel’ table, yet if they complete the record in their copy, would be far better and therefore advisable.
3. Care is being taken to test new update before release, yet few odd errors may be noticed after the release. Once you download an update of the application, you should keep the older version until you are satisfied with testing of new update at your end also. Once you are satisfied, you should discard older version to avoid any confusion at a later stage. This would also save you some disk space and some file clutter.
4. Further, I would recommend you to backup PDF reports, initialized database created for next process and attached Accounts database. This backup should not only be kept on optical media like CD, should also be kept on some other computer/server so that in case your PC crashed and backup on optical media is found to be corrupted, you are able to restore prepared application to start the process.

# Revision History

## Version 2017-18

### release 1

1. Merged NGO Disbursement v1.5.2 and NGO Tax v2.4 to streamline as one package. Income Tax Rules 2017-18 incorporated.

### release 2

1. Bug fixes arising due to table name changes and fixing of performance issues with calculating tax.

### release 3

1. Bug fixes. Added a new form ‘Separate Emolument’ to help decide, which ‘Amount Type’ be separated on ‘Unit Pay Summary\*’ reports.

### release 4

1. Bug fixes related to tax return.

### release 5

1. Bug fixes related to ‘Initialize’.

# A Word from Developer

1. Though the application design, started with a much lower aim, has emerged well to handle nearly any disbursement to individuals through CBS and deducting tax whenever salary is paid. You can have different sets of databases for different bank accounts, e.g., welfare fund, education fund, etc. Handle the individual as retiree if individual letters are required to be generated. However, responsibility of maintaining the database in synchronised state rests with the user. I could only assure you update per account through ‘Initialize’ or ‘Import Process’.
2. I have tried to keep the system as simple as possible, however, it might contain some ‘bugs’ or errors which despite my best efforts might not have been detected. A feedback on such problems will help improve the system. Once the lists are entered in the system, I assure you that I would not tax you for re-entry of lists in subsequent versions. I must admit my limitations as developer and limitations forced on me by MS Access. Strange! But, true.
3. I wish you all a happy computing in the field of disbursement and tax.

❖