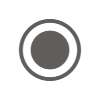
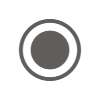
**Assignment 3 Part 1 Info-20250324\_110150-Meeting Recording**

March 24, 2025, 6:02PM

1h 43m 59s

 **Shawn Graham** started transcription

 **Kitty Wong** 0:03  
Nope, just get oh.  
Thanks, Shawn.  
Yeah, I just look just as mean, I guess.  
It's just me, Mario.  
All right.  
Well, there's good.  
At least we're recording.  
Alrighty, OK.  
So today I would like to discuss your Nixon last assignment in this course.  
So this is going to be our assignment #3.  
So of course, where you will find all the information as usual is on D12.  
Starting off with.  
Of course under the content.  
The content area here into the assignment documents, of course.  
Where you found assignment one and assignment two. Now we have assignment 3OK.  
So what? You're gonna also notice that assignment three there is like three different sections, OK?  
So the first section here does.  
Just label assignment three has some basic information, some documents that I am going to be discussing with you. OK. And the next part you will see that is broken up too. Well, assignment three-part one and assignment 2-3, Part 2.  
So this assignment, this third assignment is broken up into the two parts where the two parts you are going to be doing separate submission and they both have.  
Well, different submission dates.  
OK.  
So I am going to just be discussing part one today, OK.  
Part 2.  
We will talk about probably sometime next week.  
So you are going to need to have all those documents, so you're going to, well, I mean, we're going to be going through all those documents and of course the most important one is going to be for now your assignment three-part 1 document. This is of course.  
The details of what you're going to be doing.  
And of course, all the scripts that you're going to need to actually complete this part, one of this assignment.  
So if you bring up, of course this first document.  
And I'm just going to go to that one. It is of course our good old PDF document that is going to describe everything that you need to know for this particular assignment. OK. So of course, this assignment, you are going to be working with plcql.  
Specifically, you are going to be now creating a brand new or a PLC clip.  
What the heck? The PLC code program from scratch and you're going to of course be testing it to make sure it works.  
OK.  
But what makes this the first part?  
Part One is you're only going to be working with what we call clean data.  
OK, so clean data just means there is not be no problems.  
With the data so you do not need to actually have any exception handling included in your program.  
For exception handling, that is our next part, Part 2.  
So.  
A little information about this assignment.  
So you're gonna be, of course, presented with a prom. And as I mentioned, you are gonna be writing a PL. SQL program from scratch.  
And you are gonna be, of course, having to meet all the guidelines and restrictions that are listed and that I am gonna be discussing. OK. And you're also gonna be expected to test your solution.  
Specifically, of course, to make sure it works.  
What you're gonna notice here that it does tell you that you need a basic understanding of accounting, OK?  
And I wanna preface this by saying, well, in software development, we're always gonna be developing software in.  
Well, in, in, in, in.  
Junction in junction, not in addition.  
In addition to, well, other types of business.  
So different types of industries.  
We as a programmer or a software developer are not expected to understand everything. That is of course dealing with that industry, but as a programmer we do need to have some basic understanding.  
So there is going to be the different documents that I have pointed out to you earlier.  
So this is in the document called accounting terms.  
And a ungraded activity that helps you.  
Well, make sure that you do understand what you are going to be.  
Doing for this assignment.  
So we are gonna be well, I am gonna be introducing you to some very basic accounting terminologies.  
In the next section, this is just some basic instructions, so this assignment is going to be completed in your assigned groups.  
So the same groups that you have been working on, of course in your previous two assignments.  
And what you are going to be submitting is the one script that is their solution. The PL. SQL program of course, onto D2L under the assignment section. OK and the due date is.  
Already posted OK, so if you go of course to the assignment section, you are going to of course find the submission folder or assignment three-part one.  
So part one is due April 4th, OK? And as I mentioned, Part 2 has a different due date.  
So that is where you gonna be submitting of course.  
Your solution?  
The actual PLC Co program.  
Script.  
Some basic suggestions here of course.  
Make sure that you are reviewing the problem that we are going to be discussing and how you going to be graded.  
In this document OK.  
Very similar to your previous assignment, every time you run the script it is going to probably change something in the database, so make sure you refresh, meaning we run all the scripts before between every time you run your program.  
Hey, as I mentioned you are expected to test your solution completely to make sure that it does work and when we say it does work and it is going to be modifying or updating.  
The database tables.  
Correctly, as I am going to be describing later, as I had also mentioned, this is something to deal with accounting.  
So this is dealing with people's money.  
So it is going to be very important that you are not corrupting the data, so you are maintaining data integrity because people get very angry if you screw up their money.  
So that is some of the basic suggestions listed there.  
Hey, starting on the next page is the actual prom that you are going to be solving for this assignment so specifically?  
We're gonna pretend that you are. Of course running some software company and you are going to be having a customer or client called the we keep it storage company.  
OK.  
More specifically, they want you to write a program for their accounting system. OK. And of course, they've given you all the files that you're gonna need.  
Along with those data files, they have also given you a read me file. OK, that really just tells you exactly how to run those scripts so that you can create their database correctly. OK, so this is part of the scripts that you can download again that I had.  
Shown you on D2L. So in your previous folder there.  
So this is just a Part 1 scripts and when you download the SIP file.  
Of what you're gonna find is, well, when you unzip it, you're gonna get all these files.  
So you're gonna have a read me file in here and this read me file is basically well, the order of the scripts that you should be running. OK, so for part one, we are just gonna be doing this first one. OK, so you don't worry about the 2nd.  
One yet.  
So the first script you're gonna run is the create script.  
And then the constraint scripts and then the load script, right?  
And then you're going to have to have some test data set that is already given to you, OK?  
So this is just an example test data set.  
Your your solution should work.  
Of course not just with this data set, but any type of data that is at least clean.  
OK.  
So that's what's in this.  
Read me file.  
So what you're gonna find is, of course, all those scripts that are mentioned in the README file here along with also an ERD. OK.  
So the ERD of course.  
Of what this database looks like and if you bring up that ERD is just gonna be your basic erd, OK?  
OK, so you have this.  
We keep it storage company.  
And there is a few tables.  
And there are a few tables that you're not gonna be using, especially for this part of the assignment. OK, but there are some tables that are gonna be very important that I'm gonna be describing in a few minutes. OK. But there is the erd there involved.  
So that is the basic information that you are given. OK, so First off.  
This is what we call a double entry accounting system, OK.  
So you may be wondering, well, what the heck is that?  
So that is something that I am going to be talking about right now that is in this accounting notes document. OK.  
So again, the accounting notes document is the one that is on D2L.  
Under the assignment three section. So this first document that is called accounting notes K and some this are just basic.  
Information or basic terminology that you need to know as the programmer to make sure that your actual program is going to be doing the correct things. With this accounting system.  
Yay.  
So that is the accounting notes document that I have here.  
One of the first things I want to mention too is.  
Just going to bring up this thing that I have already opened.  
So this is the Wikipedia article on something called Enterprise Resource Planning. OK Enterprise resource planning or ERP systems are usually well things that we as a programmer are going to be dealing with in our in our life.  
Hey. So you when you graduate, of course.  
You are very very likely.  
Likely to be dealing with some sort of ERP systems, and that's because why like this is because of this diagram.  
OK.  
So ERP system is the core of every business.  
And every business will have some components of this or even all of those components, they that are going to be of course done with some sort of software and those are usually going to be the type of software that you will be working with in your careers, so.  
One of the things you will notice here is.  
This involves, well, quite a few different things.  
Some things that you may be familiar with, or some things that you may have heard of.  
And some other things that you may not be aware of.  
So things like of course.  
Sales or procurement production. So procurement production, that call can also will all come from the e-commerce, OK.  
So e-commerce is just a fancy name of, well, of course, anything that we people can actually buy and sell online.  
So any website, any web application that you may be developing.  
Yeah, that does, of course.  
Sales or things like that with inventory, those all are of course all done with some sort of software, OK. And of course, other things like human resource that might be of course like payroll systems.  
So of course we need something like like payroll systems too, of course.  
Make sure that we're paying the people we that work for us, OK?  
But the very kind of the bottom, the little purple pumpkam, the purple little section.  
Down here is of course accounting OK.  
So accounting is, well, a very important part of every business where it basically keeps track of well.  
Any money and resources of a company, OK, specifically their finance, finances and accounting systems.  
Are very.  
Important not in the sense that it just keeps track of money, but we also need to make sure that we are adhering.  
Into the basic rules and laws of the country that you're dealing with. OK, because a lot of our companies, especially if they're like public companies, publicly traded companies, all their finances are going to have to be very transparent and very visible. And it has to, of course AD.  
To the local laws of the nation.  
OK. So accounting systems is?  
Very important.  
So this is also why we wanted to take this opportunity for you to introduce you to this kind of.  
Very common type of systems that we would you may be working with in your future.  
But like I mentioned earlier, just because you're dealing with accounting systems doesn't mean you have to learn everything about accounting, OK.  
So that's why people have, you know, accountants, those.  
Certified registered people.  
That actually deal with the the details of accounting, OK, but as the programmer it is also important to actually know a little bit about what we're working with.  
So that of course, the system that you're gonna be developing, well, it's going to actually make sense.  
It is gonna well be doing things that are supposed to do, and of course using the correct terminologies.  
OK, as we actually interface with our clients or our users.  
OK, so that is what this document is all about.  
OK.  
Specifically you because you are going to be working with the accounting system that is already in place.  
There are different things that you have to be aware of of what exactly is involved in this accounting system, OK?  
So some of the main things that you need to know is First off, there are different what we call account types.  
Eight the different account types are listed here.  
The first type is what we call assets.  
OK. So assets, I like to think of this as things that the company will owns, OK. So assets are things that have value, OK that the company actually owns. So for example, assets can be of course just cut the cash today that a company actually has anything.  
That they may be receiving through their various accounts. OK, property that they own or of course any investments that they have.  
And the one thing that you are going to notice, the most important thing here as the programmer, especially for this system that you're going to be developing.  
Excuse me.  
What we call the default transaction type for this account type. OK, so for assets the default transaction type for any asset accounts is what we call debit.  
OK.  
The second type of accounts is liabilities.  
Liabilities is what I like to think of, of what the company owes to somebody else.  
OK. Or another company. OK. So examples of liabilities is what we call accounts that are payable.  
So there's our money that you're going to be paying out. Of course, any type of mortgages or loans that the companies might have.  
OK.  
N.  
Again, there is this thing called a default transaction type for this liabilities type of accounts and the default transaction type for liabilities is what we call credit.  
The third type of account is what we call owner's equity specifically for a company that is a sole proprietorship.  
So owner's equity is what we call cumulative account.  
And there is ever going to be only one of this account, OK.  
So for the previous ones, assets and liabilities, there can be of course multiple OK for each type of asset, so liability?  
But for owners equity, there is ever only going to be one single one of this account.  
OK. And this is really just kind of well what the company likes to see?  
So whether they're making money or they're losing money, this is the account where they want to actually, well, see what's going on.  
Hey.  
There is something that accountants use to calculate this kind of how much profit or loss that the company may have.  
OK. And this calculation is given in this formula, OK.  
And from a programmer's perspective, especially for this assignment.  
You don't really have to care too much about that. Hey, but what you do need to know is this thing called the default transaction type.  
OK.  
So again, there's a default transaction type dealing with the owner's equity account, and this is a credit transaction type.  
OK.  
Those are the three main different account types.  
This owner's equity because this is a cumulative account.  
The cumulative account well comes up, comes out from well to other what we call temporary accounts, OK2 temporary accounts that I'm going to call 3 a 3B here a the first one is revenues. OK. So revenue accounts are of course anything that is coming into the.  
So revenue?  
Any money, any income that is coming into the company, OK.  
Again, there is this thing called the default transaction type. Dealing with this type of accounts, and this one is a credit account.  
And the other subcategory of our owner's equity is expenses.  
So expenses is are another temporary account, of course on expenses. So things that the company is going to have to pay such as things that they have to pay for other companies like utilities, loans, interest, things like that. OK.  
And.  
Oh my goodness, where does this line come from?  
Why? Why is my Microsoft Word doing that lately anyways?  
Ignore that line, I'm sorry.  
So.  
For all accounts, just like before, there is this thing again called the default transaction type for this account and for expenses this is called.  
This is a debit type of account, OK.  
So those are just the different types of accounts that you are going to be dealing with, OK. And you are going to be seeing them of course in the database.  
OK.  
So going back to just this query quickly this erd.  
What? You're gonna actually find is a table called First off accounts, OK.  
So this is all the different accounts for this because company.  
And of course you can see.  
See that has related to this other table called the accounttype OK and the accounttype here has of course a primary key that is a accounttype code OK.  
And then there is also this default transaction typed field. OK, so this default transaction type field is exactly what I had just mentioned.  
Oop. Sorry rob.  
Hey, with those different types of count, so the default transaction type.  
And of course, if you actually run the scripts which I am going to do right now.  
My scripts OK, so of course you want to run them in the order.  
Now I'm wondering if my other README was incorrect. OK, so the order given here, the first one you should be running is the create script.  
And then you should be running the constraint script.  
And then you should be running the load script.  
OK.  
So once you run those first three, what this creates is the actual accounting system.  
That for this company, of course, you can take a look at some of those different accounts or the different tables that I had just mentioned. So there is of course this account type table.  
And you can take a look at this account type table.  
And what you're going to notice is that there is a few different things, such as the first field.  
So.  
I have to Scroll down like this first field here. A/C is the account type which is the primary key and then the next field is this D which is the default transaction type. OK and then the description OK.  
So the description just tells you what what the different account types here are.  
So those are the three fields that you see in this table and the first one you see is the account type code, the primary key.  
A for the assets, OK and the default transaction type for our assets are D for debit, OK and so you can see that there is a note here our default transaction type can only either be the letter D or C.  
And this already pre created for you in the database is exactly what I just mentioned.  
OK, so accounts.  
The default transaction type is debit.  
Hey, so that is exactly what we see in this table.  
And our liability liability is just the letter L here.  
Hey and the default transaction type is C, which is of course credit for liabilities.  
And you can see the same thing for all the different other three account types, such as our temporary.  
Account expense expense.  
The default transaction type is D debit.  
Then we also have revenue. The cattle code is re and the default transaction type here is credit and then OE is our owner's equity.  
And again, the default transaction type here is credit. OK, so that is the account type table.  
That you're going to be dealing with and the other one called account.  
So this is all the accounts for this company.  
Currently OK.  
So when I say currently it means well, you know it could change in the future. They might have more accounts or they may have less accounts.  
So or they may change something about each of those rows in the account.  
So again, these things are not static, but this is what it is right now as we are going to be developing this solution for this company.  
And what you're gonna find here is, of course, there is also a primary key.  
So the primary key for account here is the account number. OK.  
So this account number is this first field account number, OK and each account have of course a unique number that you can see here.  
OK. And the next one is just of course the account name here, which is just of course a varchar.  
OK. And the relationship to the account table is a foreign key here, which is the account type code. So the type that account type code is this next field, OK, so you can see right away that it has.  
The first few is a a type, so a type of course a down here or back in our previous account type. This is a asset account.  
And then of course L for liabilities etc.  
So this is what those letters here mean.  
So our first four accounts is assets. Our next two is liabilities.  
We have 3 expense accounts.  
We have two revenue accounts and we have the one OE account.  
OK.  
So those are the accounts that you are going to be of course dealing with in this assignment.  
Alright. Any questions so far about the different account types here?  
So if there are no questions.  
The only the next thing that.  
We are going to be talking about is some rules that must be followed.  
In terms of accounting, OK, so this is going to be something that you may or may not have to do that is specified here.  
So the first one, the more if the most important rule is what we call the debits equals the credits.  
OK.  
So that is something that you are going to have to make sure that it is the same.  
A for Part 2 OK.  
So we don't have to care about that for this particular first part of the assignment, OK. And the other thing that the other accounting equation, OK.  
So this is not something again that you need to actually validate in this assignment, OK.  
So I am not going to be discussing those right now.  
So the next thing I want to talk about is specifically.  
What is happening with this accounting system specifically?  
What we call this double entry accounting system.  
OK so.  
Before we talk about what a double entry is.  
What you need to know what you are going to be doing for this assignment is that.  
You are your program is going to have to process transactions.  
OK.  
These transactions are from a holding table called the new transactions table.  
So this new transactions table, we'll go back to our ERD is this new transactions table, OK. And there's going to be data in this new transaction table when you run the test script, OK.  
So this the data set. OK. So right now before I run it, what you are going to see right now with this new transaction table is there is nothing in it right now.  
Yep. Hey, so there's nothing in right now.  
OK, after you just run the 1st 3 scripts, but what you're gonna have to do at this point is run that 4th script and this is going to insert.  
The test data.  
OK.  
So that's all it does.  
But what you're gonna find in this test data set if you actually examine this.  
Not to open. Is this just a whole bunch of insert statements specifically into this new transaction table that we were just talking about?  
Nick.  
And what you're also gonna see in the script is that.  
It has been labeled. I have given you some some extra comments in there to tell you the different types of transactions, OK, or specifically the different numbers of transactions because what you are going to be doing is you are going to be of course processing everything in this.  
Table that has now been created and you are going to be.  
Inserting them.  
Processing them.  
So you're going to be inserting those transactions into two other tables.  
So this is the transaction detail table and the transaction history table.  
So those two tables is of course also in this particular DRD.  
So we have the transaction detail table and the transaction history table OK and to start off with both of those tables are again empty.  
So we can again.  
Look at it transaction transaction log.  
Can spell transaction history.  
Hey, there's nothing in there.  
And of course the other one is called transaction detail.  
And again, there's nothing in there yet.  
OK.  
So those are the two tables that you are going to be inserting into, OK.  
Of course you can note that there is various fields that you have to insert Kay so that in the transaction details.  
No table there.  
Is one primary key which is referencing the account number.  
And this account number, of course, is the same account number from our account table. OK.  
So this is of course also a foreign key.  
The second one we have is the transaction number, which is a second primary key.  
OK. And the second primary key is actually also a foreign key.  
Into the transaction history table. OK, so this is actually referencing this second primary key here in this other table, OK.  
And then there's also other things like transaction type, so transaction type again can be debit or credit. So it's only 1 character.  
And then also there is a transaction amount.  
Right.  
In the transaction history table, there is again the primary key.  
There is a transaction date and there is a description K so all those are actually coming or many of them are coming directly from this new transaction table such as the transaction number. Here OK.  
So the transaction number is that transaction number. The transaction date is that transaction date.  
The description.  
Is that description.  
Hey the account number.  
Is.  
That account number.  
OK. And then the transaction amount is?  
That transaction 011 transaction amount.  
Oh, translation type I was right.  
Transaction type and then transaction amount. Sorry.  
I'm I am confusing myself of those different fields.  
So that's where all those information comes from.  
So what your program is going to be inserting is just directly those data.  
Right.  
So is everybody OK with those first two tables here?  
What your code is gonna be doing?  
Any questions about that?  
And that's the easy part.  
OK. The next part is again what I mentioned earlier, since this is an accounting system, we have to make sure that.  
We are dealing with the money part well correctly, OK.  
So when you are processing those new transactions from the from the new transaction table.  
While you are inserting into the two other tables, you're also doing an update.  
OK.  
Specifically, you're updating the account balance in the account table.  
OK.  
So the account table of course, as I had shown you earlier.  
Which I will just do it this way.  
Account table.  
OK.  
So the account table here of course has this last field called account balance.  
And right now it's all cyl.  
But of course, what you're going to have to do is once you process all those new transactions, right?  
So again this script that you are going to be this data set that you are going to be processing once you process all those, all the account balance has to be updated OK.  
So that's essentially what you're going to be doing.  
OK, but there's updating.  
There is a little trick to it, because this is what we call again, this double entry accounting system.  
So this is the funny part.  
OK.  
So I'm gonna go back to my accounting notes here.  
So for double entry accounting system, hey.  
The words debit and credit doesn't mean what you think it is.  
So what we think as kind of regular non accounting people, OK, we always think that a debit means you're well taking money out, OK.  
So when you perform a debit transaction, let's say on your bank account or when you do a debit on your credit card, OK, that means you're taking money out. And that usually means as well you subtract.  
From the balance right?  
And when you do a credit transaction, so you're doing a credit on to again your bank account, that means we are well putting money into the account, OK. And that usually means addition, OK.  
But but.  
Isn't really what happens in a double entry accounting system.  
And there is an example here which.  
I'm gonna ignore for now, OK, so.  
What I want to do, sorry, I'm just gonna bring up my paint.  
Actually.  
OK so.  
For a regular person.  
Again non accounting person.  
If you have a course, what you have is something like you have a bank account.  
OK.  
And then of course you will have some of some balance in there and then you will have well bills to pay.  
And most of the time, well in, of course in our current date and age, what you may.  
Have already set up is just kind of pre authorized automatic payments, OK.  
So let's say you have.  
What we call the due date, yeah.  
The due date for those bills.  
So let's say it is whatever is today's date. I am so confused on what day it is.  
Let me just quickly.  
Today is the 24th. OK, let's say is the 25th, is the due date.  
OK, then that money is just gonna be subtracted, OK?  
So if I'm a bank account, we're gonna basically subtract that amount on that day.  
And it's done.  
So that's kind of usually what we normal people deal with.  
But with a company so again, especially for companies that have to make sure that they are reporting all their monies, all the profit or any losses that they gain K it is a little bit more complicated, OK and this is what this double entry accounting system is doing.  
Wait. Hey, so.  
Instead of, well, let's say we just pay our pay our bills at the due date.  
For a double entry accounting system for a company, what is gonna happen is, well, there's two different steps.  
OK.  
So the first step is well, when we receive the bill.  
OK.  
So when we received the bill, let's say we receive it today on the 24th.  
OK.  
What we're gonna have to do at this point is we're gonna have to plan.  
We're gonna have to make sure that we know that we are going to have to pay this bill at a future time. OK, so that means.  
We need to actually have well.  
Something like account payable.  
OK. And we're gonna have to note that we are gonna be having to pay this bill, OK?  
So let's say this is like my Netflix bill and this is, I don't know, whatever it is right now, 2099.  
Without tax.  
OK.  
So when we receive this bill again, we have to make sure that we are noting this in our account payable. OK.  
So we are going to make sure that we are going to make, we are going to keep track that we are going to have to pay this.  
$20.99 in the future, OK.  
So we basically we add this is the amount that we are going to have to pay in the future in this payable account, OK.  
So that is the first step.  
And then.  
When the bill is due, so on the due date. So this is of course we said it was the March 25th, OK.  
So again, this is when we actually have to pay this $2920.99, OK?  
So when we actually do that, then what is gonna happen is now we actually pay this bill.  
So this is of course from something like you cash or whatever account that you're paying from.  
And this is when we actually have to subtract that amount.  
The $20.99 from that account to actually pay this bill.  
Right. And at the same time, because now we've paid the bill, we also need to update our account payable.  
To make sure that we know that we have already paid.  
This K so that means we're no longer owing that $20.99.  
OK.  
So that's just an example of how things work in the company.  
So actually when we're dealing with official accounting software.  
Right. So that's what we are going to have to or that's what you're gonna have to do for your software, for your program as well, OK.  
More specifically is well.  
Whether it is a debit or credit.  
Transaction K.  
Then.  
It is going to determine whether you are going to be adding or subtracting.  
OK.  
So I do have a basic rule for this that I am going to.  
Write out over here.  
So this is my basic.  
My basic rule for this accounting system.  
So.  
It is just going to be an if else class OK so if.  
The default.  
Transaction type.  
Is the same as the current transaction type though, just just like that.  
OK.  
10.  
We add to that account.  
If not, of course, this means that the default.  
Default is not the same as the current transaction type.  
Oops.  
K.  
Then you subtract.  
From the air kept.  
Right. So keep that brew in mind when I.  
Am going to explain this a little.  
I'm just noticing the time, so maybe we should take our usual 10 minute break first.  
And then I'm going to talk in detail about exactly what your program will have to do to make sure that your account balances are correct.  
So it is 1249. So we come back at exactly 1259.  
See you in 10.  
Hello, welcome back.  
Wave hi S emojis.  
Anything of that sort? Thank you.  
Already. So hey, so as I was mentioning before the break.  
So our double entry accounting system is a little different. OK, so.  
Back to our.  
OK.  
So what we are going to have to do is determine whether we are going to be adding or subtracting OK given.  
The account type.  
OK, the default transaction type and then the actual transaction OK.  
So what I am referring to is specifically.  
The test data set that you are going to be of course running and processing in your program.  
So what you notice in this particular data set?  
I have something called transaction 1K and transaction one actually has two rows.  
K and they're the same transaction because First off you would notice that I'm using this sequence that has been already created in the database. OK, so I'm using the next Val and as you may recall, next Val would generate the next number in the sequence. OK for this.  
1st row but the next row is using the curve valve function.  
The perv L function was just returns the same number.  
So what this means is that these two rows this first two rows here are going to have the same value for this first field and this first field is the well oh boy.  
Let me see if I can undo this a little bit. There we go.  
OK, the first field is of course this transaction number, right?  
So this transaction number here. So this first two rows gonna have the same transaction number, OK.  
So First off, you will notice that this transaction history, this transaction number here again is a primary key, OK.  
So the transaction history is gonna contain only.  
One row per transaction.  
Hey so even though there are two rows here in this new transaction table, but because they have the same transaction number they are the same transaction that I'm just calling transaction one here OK.  
So transaction one is going to be one row that is inserted into this transaction history table.  
And those. But because there are two different transactions.  
Rose in this one transaction, both of those are going to be inserted into the transaction detail history, OK.  
So the detail here. OK, we'll have both of those rows, even though this one only has one row.  
So that is the first step that you're doing when you are inserting.  
So this first this step here when you're inserting those transactions into those two other tables.  
Right. But like I said, the more important part is this next part, OK, which you need to of course update the account balance. OK. So as I've mentioned earlier, to update the account balance, you need to actually decide whether you're adding or subtracting and this is gonna de.  
On the default transaction type of that account, OK.  
And then whether we are doing a debit or credit transaction according to my little rule here?  
So this default transaction type this comes from the actual account.  
That you're dealing with.  
And this transaction type, the second one is the current transaction. OK.  
So let's just look at this first row. If our first transaction here.  
So as we know, the first field is just a transaction number, the second one is just a date.  
And the third one is just a description.  
OK.  
So those are the 1st 3 fields in this particular table, OK.  
So those are the simple ones that we just talked about.  
OK, the next field is this account number.  
So of course, as I've described earlier, this account number is 0. Nope.  
Is the account number of the account that you are going to be dealing with for this particular transaction. OK, so this account number.  
Is this valley OK?  
So this next valley here and this 1250 is something that you are going to find, of course in the actual account table. So in the account table that we were looking at earlier 1250 is this first account which is our cash account.  
OK. And once you look that up in the account table, what you have to do now is you have to determine?  
What the default transaction type of this account is? OK, so this is again referring back to.  
This account type table. OK, so once you've looked up the actual account number, you know what the account type code is. This allows you to look up what the default transaction type is in the account for that account type.  
So that is what you're going to have to do first.  
So your your program is going to have to look up this A and of course a as we had mentioned earlier is the asset account.  
And the asset account has a default transaction, so this is a debit transaction type.  
So our assets have a default transaction type of D, OK.  
So I'm just going to write that down here.  
This is ad for debit.  
So that's what the first information you're looking at.  
And then what?  
You're gonna now have to do once you look that up is you're gonna have to compare it with the transaction type for this current row. OK. And this is the next field. This next letter. OK.  
So this next letter, which is of course in reord this is this transaction type, OK. And again the transaction type can be D for debit or C.  
For credit.  
And this.  
This is gonna now tell you what type of transaction we're dealing with currently in this role for this account.  
So we're doing a debit transaction on this account 1250, which is an asset account.  
Today. So this one. Now we are doing a now a debit transaction.  
This is the current 1K, so we're doing a debit transaction on this asset account, which has a default transaction type of again debit.  
And once you have those two pieces of information, then you do this comparison.  
If they are the same, meaning that there is a debit and a debit, then we are going to add.  
Right. So what that means is you are going to now add, sorry going back to add the balance.  
On the transaction amount here. OK, the transaction amount is 30,000. You're going to add this to this account?  
Out 1250.  
OK.  
So this is gonna be added of course to this first account, which means this account balance should now be 30,000.  
OK.  
So that is the first row.  
OK. And in the same transaction transaction, one, we have the 2nd row which again we know is the 2nd row because it has the same transaction number and it still has a date and it still has a description OK, but it's going to now deal with a differe.  
Account. So this different account this is 305830583058.  
In our account table, this is our service revenue account, OK.  
So the revenue account of course is a type of revenue account, OK.  
So the first thing you gonna do for this processing, the 2nd rows you have to look up this default transaction type for revenue account.  
And the revenue account is of course in this account table. OK. So revenue account with the code re.  
And the default transaction type.  
For revenue accounts is credit, OK. So we're now gonna note I'm gonna use a different color that this time we're now doing a credit.  
Transaction. OK, so the default transaction type for revenue is credit.  
And this current transaction for this row, this one row that we're processing.  
This transaction is a credit transaction, OK.  
This credit transaction means that our current.  
Transactiontype is also credit.  
And again, once you have those two pieces of information, then you can do this comparison, OK. And if they are the same meaning that both credit, then again we are going to add.  
OK, which again means that we are going to add this transaction amount 30,000 to this account 3058.  
So now what's going to happen is that 3058, OK, the balance is now going to be.  
30,000.  
Look.  
Does that make sense so far?  
What does adding and subtracting thing looks like for now?  
Alright, so let's find a different one.  
This 11851850 is.  
Set.  
K.  
That's also a debit.  
Sorry, I'm just gonna try.  
I forgot exactly which one this is.  
1250.  
1250 is there.  
That's another one, OK.  
Let's just.  
Do.  
This one. So so that's.  
So you're gonna do the same thing? Of course, for each of the transactions. So transaction 2 here, of course also has two rows. OK.  
And again, you know, as the two rows, because it is the same transaction number.  
And again, all you're going to do is you're going to look up the default account transaction type for this account, compare it to this transaction type.  
That is going to determine whether you're adding or subtracting OK that amount.  
Out in this account, OK.  
But what I want to jump to is now transaction 3, transaction 3 here. This first row. OK, this first row we are dealing with this royalty revenue which is in this account 1250, OK.  
So again, 1250, if we look at it.  
1250 is this account, which is an again an asset type of account and our asset type of account. The default transaction type is the debit.  
OK.  
So that means now we are going to be doing.  
A debit transaction. OK, so the default is gonna be the debit.  
And then this.  
Oh, this is still a debit.  
Damn it.  
I thought.  
I thought I found one that was not the same.  
I can't remember which one.  
Sorry I'm I'm trying to find one that is not the same so I can show you.  
Oh, it's this one.  
That's what I was. OK, so that's what I was looking at.  
I I jumped too far is the 2nd row in this transaction too.  
Sorry, Kate. So this one.  
OK, so same account. OK, SO1250. OK, SO1250 again. We've looked this up.  
This is an asset account with a default transition type of debit. OK.  
So we are going to be dealing with the account that has a default transaction type of debit, OK. But then for this particular transaction, this role, this transaction row here, this is a credit.  
That transaction, OK.  
So this means we are now the transaction type here is credit.  
And once again, we have those two pieces of information we're going to do the comparison, OK. And this time we know that it is not the same, OK. The same means the default transaction type is the same as the transaction type. So meaning it is either a credit.  
And the credit, OK, or a debit and a debit, OK.  
But when we have.  
A debit and then the credit, then they are not the same.  
This means we are now going to subtract that amount from the account.  
OK, so for this particular transaction, OK, we're now going to subtract 30,000 from this account, 1250, OK. SO1250 of course, is the same account we had earlier that we did adjust the account balance, OK. So it was 30,000.  
And now we're going to be subtracting that 30,000, which then now takes the balance back to 0.  
OK, so that is what your program is going to be doing for each of those rows. Of course, as I mentioned, you are going to be inserting them into those two tables.  
So let me find a different colour, OK.  
So you're doing a insert.  
Into first this transaction history.  
OK.  
But again, this is a primary key on that transaction number. If you try to insert more than one of the same transaction with the same number, you are going to get a well, of course a constraint error. OK.  
So you got to make sure that you are only inserting one per each set of transaction. OK.  
But then for each row in the new transaction table.  
OK.  
So the in the detail table, every single row will match that same row. OK, so.  
That is, that's not one to one.  
All right.  
So that is the first two insert and of course then you are going to be updating.  
OK.  
The correct account, specifically the balance, OK.  
So this is where you have to either add or subtract according to this rule.  
OK.  
And the final result would be after you have of course, went through every single row in this day's test data set, and you have the correct balances in each of those account.  
So.  
Starting off with.  
My database in this again this previous date where we had.  
Right. Again, the account all has zero right now.  
And there is the new transactions that I have imported from. Of course this test data set.  
OK.  
So this test data set has a total of 12 transactions but 28 rows.  
So that's exactly what I have here.  
OK.  
And then.  
Also at this point.  
Our transaction history and transaction details are empty.  
So now when you have created your program that is going to do exactly what is supposed to do, such as my solution here.  
OK, when you run this program, of course there should have no compilation errors or runtime errors.  
And what this is going to do is, of course, as I've mentioned, it is going to 1st off.  
Update the two tables or sorry, insert into those two tables so those transaction history now has exactly 12 rows for each of those 12 transactions.  
OK. And our transaction detail table will have exactly 28 rows matching all 28 rows in this entire data set.  
OK. And then also last but not least again the.  
Account table. Where the hell is my account table?  
There we go.  
And my account table should be all updated to the correct account balances.  
After all those additions and subtractions.  
So that is what your program is gonna have to do to process.  
This particular data set.  
OK.  
According to the rules of our double entry accounting system.  
So some other things that I have just included. OK.  
So yes, you are given this one script.  
That's going to help you test your solution. OK, but again, there are just samples, OK, so don't hard code your program to only work with this data set, because, well, I'm also going to be running a completely.  
Different data set to test your program. OK, so it shouldn't matter what type of data, so it shouldn't matter what accounts we're dealing with, what type of transactions we're dealing with. And of course whatever amount is in each of those rows in this new transaction table, your program.  
Is still going to do all those inserts and updates correctly.  
OK, so that is this first one.  
OK. Or does part one of this program?  
Any questions so far?  
About what?  
Your program does.  
Ute.  
The last thing that your program should also be doing is once your transaction has been processed, so your program has actually gone through and inserted again into the transaction detail and the transaction history tables and also updated the balance correctly in the account table.  
It should be removed from this holding table.  
This holding table, of course, we're talking about this new transaction table where you actually processing the data.  
From K and remove of course just means you are going to of course delete it.  
K So you once you have done this.  
In this particular new transaction table, you're also going to be doing one last DML statement, which is going to be a delete. OK, so.  
Again, just like my my program has ran after I have processed all.  
All those transactions correctly, there is nothing left in my new transaction table because for part one we are only dealing with clean data, so there are no errors.  
There are no problems, so there should be nothing left in this particular table.  
Every row should have been processed correctly.  
All right.  
So just like all our other labs and assignments, there are some guidelines and restrictions here that I have noted.  
So some of the things I've already discussed.  
OK.  
So first thing assumption you can make is that every row with the same transaction number is, well, the part of the same transaction. OK.  
So this means that just like I had described before.  
Every set that has going to have this two or more, there is.  
There's one that has even more.  
Than more than one, OK.  
So there is going to have of course the same transaction number.  
So this is Nix Val.  
And then we have curvil and curvil, which makes this three rows the same transaction.  
So they all have the same transaction number.  
So that is what this first point is.  
You can assume that that is the case.  
Then yes, they are going to have more than one row is going to have at least two rows and some of them three.  
And all the information in this transaction has the same transaction history information.  
So same transaction number, transaction date and description. OK, so this is again exactly what you see the same transaction number, same transaction date, OK and the same description. OK.  
So all the same transaction has the exact same three values.  
The same first three values, what is always going to be different?  
Is the next three values.  
Oh.  
The next thing is just a suggestion or a recommendation.  
Using two nest, a cursor will make this program not just easier, but much more cleaner and elegant.  
OK.  
But you don't have to.  
OK.  
So that's really up to you as you are developing the solution. OK. And of course I had shown you an example of what nested cursor looks like back in units.  
Six, OK. So you can take a look at my sample code in Unit 6 on how to.  
Work with nested cursors.  
OK.  
All everything you need is already in the scripts.  
So please do not modify the scripts or of course add anything.  
To the current tables OK.  
So that means well with current structure or any constraints OK1 the things that I mentioned earlier is that you're definitely going to get a constraint thing with this transaction history table, OK.  
So if you try to remove this constraint to make your insert work, then it is not going to.  
Work on my database because my database will still have that constraint.  
OK.  
So do not modify or don't add anything to any of the scripts, OK?  
Another thing is do not use what we call table records or any other type of arrange solutions, OK?  
So the table records is that bad data structure that I talked about a few weeks ago that we should not use and I had also bagged that explained to you exactly why it is bad. OK, so everything that you need for this assignment you already have been doing.  
So far.  
K.  
So you're not allowed to use this other types of data structure, OK.  
This does not mean you cannot use a record data structure.  
A record is the one that we talked about either, of course using the A custom record creating your own custom record or using the row type attribute. OK.  
So those are fine, OK and you probably need to when you're using a cursor.  
You need a record of some type to hold the actual data through every iteration.  
OK.  
So that's not what we're talking about here in this restriction.  
And now no restriction is that you can only do a select into or subquery.  
Or you cannot do.  
Sorry, you cannot do a select into or select subquery against the new transaction table.  
OK.  
What you are going to need to do to process the data in a new transaction, it has to be an explicit cursor, OK?  
So you have to define an explicit cursor.  
OK, for the actual data from this new transaction table so that you can process each transaction into the.  
Other tables.  
Another thing is that your program should be just one single anonymous block.  
OK.  
So when we say one enumer block it can it can mean just well, you can have embedded blocks. OK, embedded blocks is when you have say a sub block of course embedded into another block.  
OK.  
So those are not considered separate anonymous blocks.  
Hey, but if for some reason you do have multiple blocks or other things like store programs, Ki am only going to ever look at the first one and that's all you're going to be graded on.  
Other things is just best programming practices such as do not use things like go to exits or save points in your program.  
OK, if you're using a basic loop then you can of course have a exit when condition to control your loop, but not anywhere else.  
Along the same lines, do not use. Continues OK if you do use a continue.  
It should be done appropriately.  
OK.  
So you should not be using it too well.  
Control your loops.  
OK.  
If anything, just don't use. It continues are bad anyways.  
And now the last last restriction here is not hard coding any values in your code.  
Kate, of course.  
This does not include things like counters or temporary variables, but what you should be of course doing in your code is anything you need should be declared as constants.  
That are declared and then in the declaration section and used in your executable section.  
OK.  
So that is all the information about this assignment, three-part one.  
Any questions?  
All right. So if there's no question, the next part is how you are going to be graded for this part. One of assignment three, as I mentioned earlier, I am going to be using a different data set to test your program, OK.  
So what I've shown you today that my program is running of course is against this set that you are going to be using.  
OK. Or you have you have been given?  
You are of course.  
Allowed to create your own data set to test if you like.  
But just so you know, I am not going to be using that particular script to test your code.  
The first part here is.  
If you violate any of mariju here, you are going to get an automatic 0 on this assignment, OK, and this includes things like having compilation errors or runtime errors. OK.  
So your code should have no errors in any any way.  
And of course, if you have changed or added or modified anything in the database, this is going to give me well.  
Some sort of syntax errors or runtime errors.  
Another thing that is going to get you a zero on this assignment is of course using some of the things that I had just described, like using a go to statement on exit or save points, OK and all the other data structures that you're not allowed to use.  
Also, you're not allowed to use store programs.  
So that is.  
Getting you a zero on this assignment, and if you did not violate any of those rules then I will move on to the next page to grade your assignment.  
OK. Then the first one is just simply what we have our static white box testing our static white box testing is just simply going to be a code review and a code review is that you have followed all the restrictions.  
And and everything that was stated in the guidelines in this document. OK, so things like hard coding your data.  
Using a select or subquery on the new transaction table that is not your explicit cursor.  
OK. Another thing that I'm going to be looking for is documentation.  
So specifically this is the header documentation and any inline documentation OK.  
So the header documentation is something that most people forget. I know some of you were very good with it in the in the last.  
In the last lap you've included some of those. OK, so the header documentation was something that I had introduced you to a while ago. OK.  
So this is Unit 2, where I had given you this particular template, right?  
So what I'm am expecting that you're going to be getting graded on is First off, this header block, OK.  
So you should be including those information.  
In your assignment submission.  
OK. And then of course the inline documentation that describes each block of code OK.  
So that is the documentation here.  
OK. The other thing that I'm going to be looking for is, well, you're saving your data correctly.  
Specifically, are you saving it or are you saving it in the right place?  
OK.  
So that is the code review section.  
OK. And this are just going to be deductions as noted here?  
Half a point or one point deduction for those.  
And then the next part is gonna be me testing your actual code, whether it is correct and is performing this correctly.  
Hey so for this section this is of course your main section. Your total grades is gonna be out of 40, OK.  
So the first thing I'm gonna be looking at is your new transactions. OK, so new transactions.  
Table. So as we mentioned earlier.  
Once you have finished processing the transaction then you are going to be removing them from this table. OK, so you do not remove it. You get. Of course you don't get any marks, but if you are removing it correctly then you get the 10 marks.  
OK. And then anything in between that spectrum?  
The other thing I'm gonna be looking for is.  
You are inserting into the transaction history.  
And the transaction detail table OK, so each of them is again 10 marks. OK, so from anywhere that you have correctly inserted everything to not inserting anything, OK, that will give you anywhere between those zero to 10 marks.  
And then.  
And the most important part in our accounting system is updating the account balances properly correctly, OK.  
So this is again meaning that you are adding or subtracting those transaction amounts correctly to each of those accounts.  
OK.  
So of course, if they are not updated correctly, you are not gonna get those full 10 marks or if they are of course change.  
Exactly as expected.  
Then you're gonna get those 10 marks.  
Right, so this of course is out of this 40 marks. OK, then the 40 marks. OK with minus anything in this section.  
Is gonna be your team total?  
And just like our previous assignments, I am going to be.  
Adding a pure evaluation form onto teams OK and this one. If you fill it out, it is going to be worth 5 marks. So whatever your team total you have gone, I am going to take the evaluations from you and your group members and I'm going to be.  
Using this as a multiplier.  
As a percentage, OK. And of course this is going to be multiplied to your actual team total.  
Giving you your subtotal and of course if you complete this form yourself, you get those five marks, which means your individual marks is going to be out of the 45.  
OK.  
All right.  
So that is how you're going to be graded for this assignment three-part one.  
So the other documents that I had mentioned earlier is this exercise OK?  
So this is not a graded exercise.  
So this is just for you to understand this double entry accounting system so that you know exactly what your program should be doing. Specifically if it is.  
Well, updating the balances correctly.  
OK.  
So again, this is the most important.  
Part of our accounting system, OK, so you're not going to be submitting this. This is not graded. OK, but again is going to help you with this assignment.  
And what is in here is very simple, just some some basic math.  
And this should again be something that you are doing before you even run your program so that you know exactly what your program should be producing.  
OK, so you need to know what your expected outputs are before you can actually compare it with your actual output from your program.  
So this particular exercise is very simple.  
We're again doing the pretend for some company.  
We have various accounts and various balance right and this is the current state of the account and their balance is at this moment, OK.  
And then what you're going to do is you can go through each step.  
OK. And you are going to pretend you are going to get some bills such as this Internet service provider is billing you for $80.00. OK, So what you're going to do is you're going to have to enter this into this accounting system, OK?  
Which specifically states that you are going to be doing a debit transaction on the Internet expense account. OK, so Internet expense account of course is this one we're talking about. OK. So we're going to be doing a debit on this.  
And what you're going to have to do is again, make sure you're tracking the account type, OK, our expense account.  
Going back to our notes.  
Our expense account.  
Is.  
A default transaction type of debit, OK.  
So this means oops, this means that this is gonna be a default transaction of debit. OK. And we're doing a debit on transaction on this.  
OK. And just like my rule earlier.  
If your default is the same as the current transaction type, then you are going to.  
Add to this account.  
Which means you are gonna be adding $80.00 the transaction amount here on this Internet expense account.  
And at the same time, you're gonna be doing a credit on the accounts payable.  
The accounts payable is this first account. An account payable is a liability, a town type and of course the liability account type.  
We can go back to.  
Oh sorry, I keep going to the wrong.  
Liabilities. The liability account type has a default transaction type of credit, so that means for this one we are going to be working with a credit account and this is also a credit transaction, OK.  
That means again, we are going to be adding.  
We're adding the same transaction amount $80.00 right?  
So that is.  
What I would like you to use.  
To practice what you're supposed to be doing in your program or what your program should be doing. So you're gonna do the same thing with BC and D, etc. OK.  
And then once you've done that, you should end up with the new balances for each of those account.  
And once you have determined that you can take a look at the solution document.  
The solution document for this first update here.  
OK.  
So those balances is what you should end up with. OK, such as our first one when we.  
Added $80.00 to 1500. You would notice that the.  
Amount should now be 1580.  
Or.  
Our Internet expense 0 + 80 of course.  
After that, it would end up to be just 80.  
K and all the other ones are updated with the rest of those various transactions.  
So that is the little exercise here for you to, well, to help you understand what your program is going to be doing, specifically doing this double.  
Entry accounting system.  
Great. So please spend some time working on this. OK, so my instructions here was just working with your group, but of course this is up to you.  
And I'm not grading this.  
I'm not checking this, but I do highly recommend that you go through this exercise.  
OK, to really help you understand exactly what you're going to be doing for this assignment.  
OK.  
So again, the most important part is updating those account balances OK and updating those account balances is going to depend on, well, what type of accounts you're dealing with.  
And of course the current transaction.  
So according to again my TLD order to Alden rule.  
Any questions about anything regarding our assignment? three-part one.  
Go.  
So quiet no one has said a single word today.  
Uh.  
Alrighty K.  
So if there are no other questions, at least for now, hey, I am going to.  
End the meeting.  
But the first thing I want to actually mention is Nick's class on Wednesday.  
We are going to be covering our unit #10, which is actually our next and last unit for this course.  
And currently I can't walk.  
So I will let you know tomorrow if I can walk by Wednesday.  
Otherwise, we may have another online class.  
I'm sorry I'm having a really rough time lately.  
Right.  
Anyways, so that's it for me.  
I will see you next class.  
And have fun doing this assignment.  
I.  
I I hope so too.  
Thanks nick.  
Bye.

 **Kitty Wong** stopped transcription