Fact Data Modeling

Fact Data Modeling Day 1 Lab

How Meta models Big Volume Event Data

Fundamentals of fact data

Transcript:

52:16
going to be working with uh mostly with this table select star from game details
52:21
let's just look at this table real quick so this is our table and this is what we're going to be working with
52:27
so one of the things that what we're trying to do is this table's actually
52:34
terrible like there's so many things that are wrong with this table and uh we're going to go over a lot of like
52:40
what's wrong with this table but um there's uh like let's let's just let's
52:48
just think about this for a second so game details like the grain so when you
52:53
when when you're working with fact data the grain of the table matters a lot and the grain is going to be the what is
53:00
considered the the the the lowest common denominator like the unique identifier
53:05
of this table and for game details in this case we're going to be working with
53:12
um uh uh we're going to be working with um
53:18
this right and so this is mostly one row here is a player and their points so in
53:23
this case I'm pretty sure we have game ID uh team ID player ID and then uh there's
53:33
obviously I think that that's pretty much it so what we want to this is like when we identify the grain of the table
53:40
which is like for every game every team every player we have uh that's kind of the unique identifier here right and
53:47

what we want to do is let's just go ahead and see if there's any duplicates in this table first so what we want to
53:54
do is we want to say okay um this is a very common query when you're working with logs that you're
54:00
going to want to run is you identify the grain of the table then you have some sort of count and then like because what
54:08
we're saying is this should be unique and then this is what we want to kind of
54:14
uh clarify so this query see it takes a little bit because
54:20
it's like aggregating a lot of data so it looks like there's uh almost two of every
54:27
record in here so that's one of the things that can happen right and that was actually kind of intentional because
54:33
I wanted to show people like sometimes like when you're logging uh you end up getting double data so that's going to
54:41
be one of the very first things that we want to do is we want to create a a
54:47
filter here to get rid of the duplicates so that is not too crazy right so let's
54:54
go ahead and create a thing called a duped as and then uh just kind of move
55:00
him in here and then say in this case we can just put a star here and then uh
55:08
so we have a row number here and then over and then uh Partition
55:15
by and in this case our partitioning is going to be game ID team ID player
55:22
ID call this as Rona and then I want to do here is I'm going
55:28
to say select star from duped so this will give us all the same
55:33
columns right as the last table but now we're going to have this nice little
55:38
ronom kind of feature as well so at the very end here you'll see ronom right and so ronom in this case is
55:48
there's we have a bunch of ones here but like you'll see if we say like order by
55:53
row num descending you'll see that there are duplicates in this table because we

55:59
saw that with the count one kind of thing right wow I like how this query
56:06
takes a million years like because it's like has to process all the data and like game
56:12
details so you see here are all these R nums to so these are all the people that
56:17
are duplicated all right so these are all like duplicate records that we want to get rid of so in that case all we
56:24
want to do is we want to say where row N equals one and that is going to get rid of our duplicates so that's going to be
56:31
our start query that we're going to work with right A lot of the times like what you you'll end up doing here is you're
56:37
going to have like an order by here there'll be some sort of like other uh thing that you order by so that you
56:43
always pick the F you always pick the first row uh but you know what's interesting about this data set is there
56:49
is nothing to order by and that's actually one of the other problems with this data set that we're going to solve
56:54
so um okay so we have our um duped uh kind of game data now and we have this
57:01
rolling this is looking great um one of the things that we want to talk about
57:07
here is like there's probably a lot of uh things in here that don't matter like
57:14
for example okay so we have all of this data and we want like one of the things about this fact data is that it's very
57:21
denormalized right because you see how like we have this like team ID and then we have like team abbreviation team City
57:28
and we have like player ID player name and then like all sorts of like other kind of columns in here that like really
57:36
uh um are not really necessary and it's
57:42
it's interesting because we actually have uh both columns in this table that we don't need and columns that are

57:48
missing because one of the things that uh if you remember from our fact uh uh
57:54
presentation is that we need that when right and there's no when column in here
57:59
at all and so the when column actually is we get that from game so let's let's
58:07
go ahead and get that in our we can just probably uh join that here or we can say
58:12
uh join games g and so this join here will give us our this is to be game
58:22
details details. game ID so so this is going to give us our um an let's call
58:30
this GD just so because this is obnoxious okay so we we we want to keep everything from game details but then
58:37
from game we have you'll see there's this game date EST
58:43
so uh back to your question we can actually use this to determine like okay
58:49
are these duplicates are they not duplicates so in that case we actually probably want to throw that in the order by here we say order by
58:57
this and then we'll just pick the first one uh like based on the game the game date and if those are also the same then
59:05
like it I think it comes back to it it goes back to like it doesn't matter which record we pick so now if we query
59:12
this let me put the game date first and uh format things a little bit so if we
59:20
run this you'll see now this code is going to be looking a little bit nicer so you
59:26
see how we have uh we have the the game date we have the game ID we have like
59:33
the team ID um and then we have like player ID but you see like we don't need
59:40
team abbreviation and team City because remember when we're doing fact
59:46
data modeling like uh if if you can join something cheaply then we don't need to
59:53
put any of that data in with the fact right and because team how many teams are there in the NBA 30 and like even in
1:00:00

a 100 years how many teams are going to be in the NBA 100 like it's not ever going to be a it's never going to be
1:00:07
like big data right it's it's ever like all the you can put all the teams in the
1:00:12
NBA for the next 250 years and they will all fit in Excel easily so the fact that
1:00:17
we have the team abbreviation and team City in this table is an Abomination and we should not have those but games is
1:00:24
different right and that's one of reasons why we are bringing in that game time right because games is going to
1:00:31
grow if we in 250 years how many NBA games have have played you know
1:00:36
thousands hundreds of thousands there's going to be a lot of games so like not having this game time is going to be a
1:00:42
very uh it will impact our analysis of our facts a lot more than because if we
1:00:50
have to join that in for all the 100,000 records this query is going to get really slow because you saw this qu
1:00:56
right now takes 7 Seconds right and that's only on 10 years of data so imagine if we were doing 250 years of
1:01:01
data like this is going to take a lot longer right it's going to grow kind of uh like even like it's not even going to
1:01:07
grow linearly right it's going to it's going to be even slower than that let's start to think about the columns that we
1:01:13
care about here right so obviously we care about game date EST and game ID
1:01:19
because game ID is something that like well let's look at that table to see if there's other things from game that we
1:01:26
want to pull in because if there is maybe that's what we want to add uh if
1:01:31
not like okay so okay I think there is one more column
1:01:38
from game and then if we bring in one more column we probably don't need to bring in game ID because all the rest of
1:01:45

it like doesn't matter right so in this case we have all of these kind of
1:01:51
columns in here most of these columns are aggregate Colum columns right um
1:01:57
like for example assist home assist Reb like all of these are kind of aggregate
1:02:02
columns we do have um some things here that I think are important though which
1:02:09
are going to be the the home team ID and the game team ID or the home team ID and
1:02:14
the visitor team ID because we want to be able to see if a player plays better
1:02:19
when he plays at home or when he plays away so we need these two but we probably aren't going to store them as
1:02:26
columns we're just going to use them to determine other things right so in this
1:02:31
case um we the other column that we care about here is season uh you see how
1:02:37
there's season here so let's let's go ahead and put season in here g. season and a g. home team ID g. visitor team ID
1:02:46
we're going to use these mostly uh to to compare the team ID in game details to
1:02:52
the team IDs in to these team IDs to say like is is it a home are they playing at
1:02:58
home or are they playing away uh so there'll be kind of like a Boolean that we'll end up using here but we don't
1:03:04
really need game ID after that because the main reason for that is every other column in here is an aggregate right
1:03:13
that we can essentially just aggregate the game that happens on that day and we can get all of this data ourselves so a
1:03:19
lot of this is like derived so that's pretty much the only columns that we really need from the game table and so
1:03:27
that means that we don't have to put game ID in there so in that case let's just put the other columns in here real
1:03:32
quick so we have season home team ID visitor team
1:03:37
ID okay and then let's go back to game details let's kind of look at all the

1:03:42
columns in here so you saw how we had U we do need team ID here right um so in
1:03:50
this case we can say uh Team ID equals home ID right to say uh and we can say
1:03:58
and we can say this as so we need team ID as well so let's
1:04:03
let's not mixed these actually real quick because we what we have all this already we know these are coming from game so let's make a new column here we
1:04:09
can say team ID equals home team ID as and this is U playing at home we can say
1:04:16
like dim is playing at home right that's
1:04:21
the because if they if they're equal then and then it also has the the false this will also have the false so we
1:04:28
actually don't really need visitor team ID we only need home because like
1:04:34
there's not three teams in the NBA right so in that case to make this query more efficient we're going to get rid of
1:04:40
visitor team ID and get rid of visitor team ID here and uh we don't need to have home team ID as a column right we
1:04:48
just need we need it to have it be this dim is playing at home because this is a very valuable uh column that we can also
1:04:55
have so we can have Team ID here now okay so
1:05:01
other columns here team abbreviation team City we don't need those are we can just join on team ID and that'll be a
1:05:08
quick fast easy join okay player ID we uh we probably need player ID and uh
1:05:15
player name it's an interesting one right because uh the number of players in the NBA grows uh a little bit faster
1:05:23
than the number of teams but not that much faster right where it's like it's probably still um one that we can just use player
1:05:30
ID or maybe uh like adding player name I think adding player name is nice because
1:05:36

then we don't have like the this is a great example of where we can add a column to make the queries nicer so that
1:05:43
people can just like know who the player is and they don't just get some integer so I think adding both of those is
1:05:49
probably Fair player ID and player name um because just because player also will
1:05:55
grow a little a little bit faster than uh team but it will grow less fast than
1:06:00
game so we'll we'll bring him both of those um so nickname we don't need uh okay
1:06:09
start position I think we do need because that is the attribute of a game
1:06:15
because of the fact that uh like for example LeBron James Sometimes he plays small forward and sometimes he plays
1:06:21
power forward and so it depends on the game which one he's starting in so we do need start position because that's an
1:06:27
attribute of the player in the game this is like a part of the fact and then um
1:06:33
if we go down further here we have uh other columns here we'll we we'll put we'll keep comment in for now and we
1:06:40
might end up doing some other things with comment later um okay let's go through a lot of these
1:06:47
uh ones here that are probably interesting so probably Min minutes
1:06:52
right minutes field goals made field goals attempted uh we don't need fil goal
1:06:59
percentage right that's uh a waste right because that's just um uh FGM divided by
1:07:07
fga so we don't need that I think we can have the the three-pointers right uh why is
1:07:15
this like not going down okay there we go so we have field goal we have the three points made we have the three
1:07:21
points attempted we can get rid of that we can do the again we have free throws made free throws attempted our goal here
1:07:28
right is we like anything that's easy to derive like all these percentages like we don't care about right so um we can
1:07:35

have all the rebounds orb DB and Reb those are all the the rebound columns
1:07:41
and then uh obviously we'll just keep them all here we'll say assist steal
1:07:47
block um what is this why is this like not
1:07:53
letting me okay like we have a turnover here so uh this column is dumb uh so I
1:08:00
would probably rename this to column because of the fact that you see it's like blue you see how like we're getting
1:08:06
this like squiggly here because to is actually um a keyword right in in SQL so
1:08:13
what we can do is to and then I would actually just rename this as turnovers because that's what it actually is so
1:08:19
that like we can we don't use keywords using keywords in your columns is bad like imagine calling your column select
1:08:26
that's like a terrible name for a column right so we have personal fouls points and we'll keep plus minus in there as
1:08:33
well so these are all the columns that I think we should keep because they're all
1:08:39
like uh fundamental nature of the fact so this is really close now but let's go
1:08:47
ahead and just run this query and because I think there is uh there's one more thing that I want to show with what
1:08:55
this is is doing okay it's running it just take it takes like eight seconds
1:09:01
right wow that's a slow one like I did not expect this to take 15 seconds like
1:09:08
this this is like not that much data right um we might want to put like a wear clause in here somewhere to like
1:09:14
have this be filtered down so here we go this is our new data set that we have
1:09:20
right so now we have our game date we have our season and we have the team and then we have dim is playing at home
1:09:27
right and you'll see it it does have the check right so it is like there is home team and a away team then you have our
1:09:34

player ID start position so you'll notice that some people have a null start position which probably means they
1:09:39
didn't play um okay so one of the things that you'll see here is um there is uh
1:09:48
this comment is uh an interesting one where
1:09:54
there is um you see there's like dnp like there's a couple different uh
1:10:00
things for this comment that I think are um interesting and like because this
1:10:06
comment is a really hard Dimension to work with because it's like kind of like very high cardinality but you see the um
1:10:13
the first little bit there is wow dude my my uh okay there we go um you'll see
1:10:21
there's like dnp DN so there's three there's nwt dnp and DND those are the
1:10:28
three that uh are there and what they stand for is did not play which means
1:10:33
they are sitting on the bench they just didn't play and then there's a DND which is did not dress which means they showed
1:10:40
up to the arena and they were there but they weren't ever going to play because
1:10:46
they didn't ever even like wear their uniform right they didn't wear their Jersey and then nwt means that like they
1:10:52
weren't even in the arena they were like not even there so you see here's DND
1:10:57
right all these different uh uh is did not travel right so I honestly think
1:11:03
that these columns here like we want to essentially look at these together to
1:11:09
see like maybe these are other facts that we can learn more about these players with so I would think that like
1:11:16
this is this column is a great example of like a raw data column that we would want to parse so in this case what we
1:11:23
can say is um um I think there's a
1:11:29
so so there's a way to do this like so let me show you how this works
1:11:36

this is like a very strange postgress um thing what I'm going to put
1:11:41
one more thing in here I'm just going to put uh I'm just going to put one uh I'm
1:11:47
just going to filter this down for now so that like this query doesn't take so freaking long oh a 10 okay we'll do 10
1:11:55
104 so that we can uh this query should really be fast there we go there it's
1:12:02
now it's like instant um so we're filtering down to just one day of data so that like we don't fil we don't
1:12:08
process everything at once so one of the things you'll see here is we have this thing called position so you see uh um
1:12:16
this is uh like equals like so what this is doing is like this is doing like a
1:12:21
string position so we're saying like is dnp in the comment right and you'll see
1:12:27
it is in this comment and it's at position one right uh so in this case we want to say this is greater than zero
1:12:35
and then what we want to do here is we can say coales this with zero because we
1:12:40
want this to uh be a Boolean right but like when it's null if there's no
1:12:45
comment we want this to be false because we know that it's not dnp because uh so
1:12:52
now if we run this let's call this as um as dim did not
1:12:59
play so if we run this there we go so now you see how we
1:13:05
have uh it has the check marks for those days that were did Dim did not play
1:13:14
right and so this column is going to be way better to work with than that comment column right and like this is a
1:13:20
very common thing to happen when you're working with fact data so one of the things we want to do is we want to add
1:13:26
in a couple more here that's like dim did not play dim did not dress and then
1:13:32
we want one more here which is dim um this is not with
1:13:38
team and then this is nwt and so this will give us all three

1:13:43
of those columns and that should uh you'll see we'll have all three of them
1:13:49
now and then they'll be which ones are kind of checked off right here did not
1:13:56
dress all sorts of things like that right and um uh like one of the things
1:14:02
about this though is that they kind of kind of cascade on each other right because of the fact that you have like
1:14:07
if they did not dress they did not play and like so these kind of all of
1:14:13
these things kind of like Cascade on each other because like um but like uh so that's the one thing to think about
1:14:19
right so but that is something that we can do later on that's like I think that like having the be like this and like
1:14:26
just logging like the raw like is this data here or not and not really baking in the business rules at this point is
1:14:33
probably the better play and then letting analysts kind of work with these columns themselves uh later on is
1:14:39
probably the the better play so that they like they they can see exactly what was in the data so that's pretty much
1:14:47
what is in that comment column so because we now know it's in that comment
1:14:52
column we can probably just not even have it right that's an it's an interesting tradeoff here of like if you
1:15:00
feel like you've parsed everything or if you haven't but I'm feeling feeling
1:15:05
pretty good about it so I think we can get rid of it so this is now looking
1:15:12
pretty close to what we want um let's
1:15:18
let's go through all the rest of the columns here and just see if there's something else that might be uh might be
1:15:23
missing so the game stuff looking great team ID great playing at home then you
1:15:29
have um player ID player name then you have their uh start

1:15:35
position and then uh okay
1:15:40
so I hate this column I hate this column so much like like this Min column like
1:15:49
what what are you going to do with this column like like like this is like I
1:15:55
this is not this is not a column that I would want to use right so I think that
1:16:00
we probably want to change minutes here to be um maybe fractional uh instead of
1:16:07
this because this should be uh this this is a string right now right so that's a terrible column right so let's go look
1:16:14
at let's change this uh you can do thing called split oh split part okay so I think if
1:16:23
we do this is it that okay let's say like as minutes right and maybe uh is it
1:16:32
split part two as seconds I think that's what we want but
1:16:38
then we can maybe uh turn that into a decimal there we go that's exactly right so now that is going to give us what we
1:16:45
want um so that is essentially what we want here
1:16:53
um I think what we can do here is kind of like I just like to use fractions
1:17:00
like I don't think putting minutes and seconds like this is this is a great example of like okay when like what is
1:17:06
the query pattern that our analysts are going to be looking for so in this case what I would say is we can say uh we can
1:17:14
cast this as real and then what we can do is we can do
1:17:20
uh plus this we can say cast
1:17:25
as real and then this is going to be minutes so I'll I'll paste you guys this
1:17:32
cleer in just a second and I'll show you like what this is going to do so
1:17:38
now well that looks
1:17:45
like what oh you're right you're right

1:17:50
there's I missed the division I was like what you're totally right thanks for the the catch there of dividing by 60 there
1:17:56
we go there we go now now this is looking this is looking better right so now we have like it's now we like this
1:18:04
is now a usable column right where when people are looking in doing analytics
1:18:09
now they can do things like field goals per minute and they can do free throws per minute or rebounds per minute and
1:18:15
you can easily turn this into a rate that is going to be um a very powerful
1:18:21
thing that you can do with this column that's a big thing to remember when you're doing fact data modeling is like
1:18:28
are the columns that you're even giving useful right and so now I think we're
1:18:34
pretty close here to having what we are looking for um for our table so let's go
1:18:41
ahead and make our ddl because I think that's probably um because I think hold
1:18:47
up okay that is a date okay good so um let's go ahead and create this ddl so
1:18:52
we're going to say create table um we're going to call this fact game details so this is going to be our table
1:19:00
here um okay and so do we care so like this is another
1:19:07
great example of like where we want to think about each column name right and
1:19:13
uh so this First Column do we care that it's eastern time probably not it's the
1:19:20
it's a date so I think the First Column here is going to be game date which she going to be a date and then so one of
1:19:28
the things with fact data is a lot of the times you want to uh label the
1:19:33
columns either as measures or as Dimensions so in this case season here
1:19:40
like oh so our our game date is probably not game date it should be dim game date
1:19:46
and then you have dim season and this is an integer and then you have dim team ID

1:19:52
this is an integer right I think or is this a
1:19:58
long oh I think I think we want to be careful there I think I think it should be good I think because this is only
1:20:04
like one billion so I think we can say integer but it's like that one's pushing it right and then we have that dim is
1:20:10
playing at home and this is going to be a Boolean and then let's go over again so
1:20:18
uh I like to put the more identification columns first so we have like game uh
1:20:23
dim team ID then we should have dim player ID which is going to be an integer and then uh dim player name
1:20:30
which is going to be a text uh so then after that is where um
1:20:38
The Columns can be kind of like flipped I think the start position is probably going to be the next good one though like damn start
1:20:45
position it's the text right then then we have the all the same columns that we had like dim uh did not play Boolean dim
1:20:54
did not dress Boolean dim not with
1:21:01
Team Boolean okay so great um then we have a
1:21:08
couple other ones here then we have um minutes so in this case minutes is
1:21:13
actually um a measure so in this case a lot of times people like to put m in
1:21:18
front of it like M minutes because it's the number of minutes that was uh measured and this is going to be a real
1:21:25
or you can say real or a decimal they both work I like real here because decimal makes you like like provide like
1:21:32
the actual like Precision uh so field goal FGM should be M FGM because that's
1:21:38
field goals made it's going to be an integer then M fga
1:21:44
integer right let's do a couple more here MFG 3M integer
1:21:51

MFG fg3 a integer right because you want to put
1:21:57
all of these over right into what they like so that people are aware right
1:22:02
because if you if you have these naming conventions of like okay if you put dim
1:22:08
that means it's like these are columns that you should filter on and group by on and M are these are columns that you
1:22:15
should Aggregate and you should like do all sorts of math and stuff on right so we're can say mftm say
1:22:22
mft m o Reb M
1:22:29
DB and mreb right and then uh I think we're
1:22:34
almost there then we have uh M assist M
1:22:43
steel M Block M
1:22:50
turnovers and I think oh we have three more so then there's um M personal
1:22:58
files M Points then M plus
1:23:04
minus okay so great now uh we have to do one more thing here which is what is the
1:23:13
the primary key of this table so that we can make sure that we have more guarantees on this right so we can say
1:23:20
primary key so I think in this case we're going to have have a dim game
1:23:26
date and then uh dim game date probably dim player
1:23:33
ID and I I mean technically you can put team ID in there as well like but like
1:23:41
is that really necessary as the primary key I think it is like I'd put it in there too because you might be filtering
1:23:46
on that and that the primary key helps create indexes and that would be my one thing I would say why we'd put team ID
1:23:53
in there so let's put all three in there we're going to say dim team ID dim player ID the reason why you don't need
1:23:59

to put team ID is because it's like can a player be on two teams on the same day
1:24:06
and I think the answer to that is no like unless someone can like switch teams halfway through a game or something like that and I don't think
1:24:12
that that's actually possible and so now what we need to do is essentially do all
1:24:17
of these over into the right kind of uh
1:24:23
columns here right so okay so we created the ddl and then
1:24:30
what we want to do is essentially move our query with a bunch of as right so we're getting this as dim game
1:24:37
date as dim season as dim team ID and remember we
1:24:43
want to fix the ordering here so dim T then we want to put player ID player
1:24:51
name as dim player ID um okay so then we had
1:24:58
um start position this is as dim start
1:25:04
position okay so now we're we're close then what I have start position and then we had the playing at home then we have
1:25:11
all these and then I think everything else is in order and I don't have to freaking worry about it yeah okay so
1:25:17
then this is as M minutes then can you do column selection mode here oh yeah oh
1:25:23
I I I should have have done this like before see oh that's so much better so
1:25:29
satisfying to do that oh wait no we got to do it as like um GNA put it on the other side here right oh because these
1:25:35
are as right oh man it's you actually uh you can't really it's probably a way you
1:25:40
can do it that way but like uh whatever uh let's do it uh
1:25:46
mfga m f g3m this is not going to take that long M fg3 A as m
1:25:54
FTM as M FTA as m
1:26:01
o as MB H as

1:26:07
mreb M assist so one of the one of the big things I'm trying to illustrate here
1:26:12
is like yeah like you should be changing the name of things like if you are uh
1:26:19
doing fact data modeling because like a lot of times the names of columns that you're given are
1:26:26
terrible and this is a great way to fix them okay I think we got it here so now
1:26:35
what we can do is we can um like we can do an insert into here so
1:26:41
let's go ahead and do that so we can say up here I'm going to get rid of this uh
1:26:47
thing let me turn off column selection mode I'm going to get rid of this real quick and then I want to say insert into
1:26:54
fact game
1:27:00
details so now this query is going to run this quer is going to take like I
1:27:05
know like 20 30 seconds oh there we go it's done there we go I don't have to we don't have to debate about why it's slow
1:27:11
even though it took two and a half minutes it definitely should not have taken two and a half minutes but it it because it's only like a couple thousand
1:27:17
row of data but um okay so now we have all of our data here and we can see all
1:27:24
of these different columns and one of the things that is really nice about this is now we have all of like people
1:27:33
like we follow all the right naming conventions we aren't we don't have any duplicates of like excessive things that
1:27:40
we need right because uh one of the things that like is like sad is that we lost the um uh the teams right but if we
1:27:47
just join teams T right on t. team ID equals g. team ID right and then we say
1:27:55
t. star GD doar we do that right this is just going to be gdt might what oh is
1:28:02
dim team ID that's why so obviously you want to model
1:28:09

everything that way but then okay so now you see we can just bring in those columns right and we can already like we
1:28:16
can have columns we can just bring them in and it's really cheap because that team
1:28:21
column is very uh just not not expensive so that can be a very powerful way to uh
1:28:28
use your team or to bring teams into this even though like we removed them from the data set so like one of the
1:28:36
things I wanted to show here though is like like okay so we have we have like
1:28:41
all this data here but like let's just I I think uh one cool column to do here
1:28:47
like I like let's find the player who like who didn't uh like so let's do case
1:28:53
went um dim not with team so like let's find the player uh then one and the
1:29:00
player in the NBA who like wasn't who like bailed out on the most games as um
1:29:08
most bailed let's call this call this a bailed
1:29:13
numb right so we say Group by one order by two descending right now this query
1:29:20
is like way faster right boom and now you can see exact ly who this guy missed
1:29:25
21 right you see like this is the number of people who like you can see exactly the number of times right that they did
1:29:32
that but like maybe it's different though right because you also have like count um you can say count one as numb
1:29:39
games because one of the other things to think about is like okay but what about that like kind of bail percentage that's
1:29:46
probably the last thing that we want to think about here and then then I think that will be the end of this presentation but if we say like this
1:29:53
right so if we if we cast this as a real this is we're going to call this as
1:29:59
bail percent so we want to order by three descending instead so now this
1:30:05
query is really um better right so so you see here's our bail percentage like
1:30:12

which didn't that didn't sort what that definitely didn't sort

1:30:21

right like that's these numbers are oh it's because it's this is four that's

1:30:28

why okay there we go so this is probably the better way to look at it okay so this guy BJ Taylor he he has one like

1:30:35

he's 100% he's shooting 100% so there's some people who are like half the time 20% of the time so but you see the

1:30:42

people who had like the the number the high numbers like like that they had like 20 Bales but they actually just had

1:30:49

a lot more games so it was like for them it was more of a volume thing so but you can see how like this query that we just

1:30:56

ran here is very powerful and we were able to answer really cool questions from this data set that like would have

1:31:03

been a massive pain to answer with the old with the old uh data model so this

1:31:09

is the whole idea behind fact data modeling is can you build data sets

1:31:15

where you can answer questions like this really quickly right and like obviously you can do amazing things with this like

1:31:21

you can say like the number of points right you can say as total points right you can see all sorts of like whatever

1:31:27

kind of aggregations and stuff you want right you could also put in things like dim uh is playing at home and then you

1:31:33

Group by two right and then if we Group by two in order by six then we can see like okay who is the person who has the

1:31:40

highest bail percentage uh when they play at home or whatever right so and

1:31:45

you can see it's this Elliot Williams guy because he's 100% right and so um

1:31:51

that's the whole idea here right is can you make queries that are or tables that are easy to query fun to query and that

1:31:59

is if you can do that that is going to be a very powerful thing for you as a data engineer congrats on getting to the

1:32:06
end of the day one lab I'm really impressed with your Hands-On abilities here if you're taking this class for
1:32:12
credit on the platform make sure to switch out to the next link so that you can get the credit that you deserve and