Preparing Data - Data Integrity

In this video I wanted to talk about data integrity and why it's really important to get it. This will be super important while we're doing data analysis or while we're doing data cleaning.

I wanted to talk about things like these. Maybe you're talking to a government official and then you're asking for a table to do computational analysis later, and then you receive something like this or something like this. And I wanted to talk about the differences between these files. Things that we see here that makes this not the table that we want, and make this data not structured in a way that will be useful for us in our data journalism workflow. So these tables here you see that there are elements that aggregate the data maybe here we have the total, we have like subcategories, we have empty spaces, we have colors.

These are all elements that make this table eye pleasing for humans, right? The same thing goes here, you have empty spaces to make it easier to identify this. Everything here is connected to here and then we have a header here in a different color, and then we have columns here that are joint, and then we have the sum. Those are all tables that are aggregated so that we can analyze. But really these tables there are meant to be read on a screen or printed on a paper, they are tables for humans. Usually they present consolidated data that comes from tables at our structure, and those are the tables that we want with the granular data that generated these reports.

The thing is that those, these reports that we're seeing here usually you know like pivot tables and reports that are PDF files, they are useless for computational analysis because they have no integrity in terms of the previous ability to make calculations, and I'm gonna talk about why those things are different. So when we take a look at things like these, this presents months in Portuguese and then we have data about safety in Brazil too. We have year here, and then we have a different type of data here. We have "n/d" here which might be a "not available" or no for "no data". But this is very different from the reports that I presented before.

Or this one, which presents the medal table for the first Olympic games in the modern era. This is gold, silver, and bronze all in Portuguese. So here we have the year, and the city, and the country, and then the type of medals. These are tables that are meant for machines, right? They are machine readable. They can be processed by specialized software and they allow us to create tables for humans, like we mentioned before. And they're excellent for computational analysis.

And let me talk about the differences between these two types of tables that coexist in the world. Here we have the table, the structure table that I showed about safety data in Brazil. And this might sound very basic but this is super important when we're searching for data sets, and we're cleaning and analyzing the data sets. So for example in a table, in a structured table we usually have rows here in blue and then we also have columns. In each one of those columns and rows we have only one type of data. We don't have two types of data. We don't have numbers and then a text, or then names of cities and countries in the same column. We have only one type of data. We also have here in blue what we call the primary key. So the row number that is associated with a certain register. And here we have the name of the columns that will help us identify what is within the column so that we can make calculations later. Here we see for example in the year

column in this, we have the year 1896 and in the country column at this we have here, "Estados Unidos" which is United States in Portuguese. So, we see that there are only countries here, only cities and only years. Now take a look at this one.

This is a data set. This is a table that shows airports, different airports in Brazil but it also shows the acronym for states next to the airport name. It also showed the total here at the top. And it adds and mixes numbers with percentage. This showed the numbers of flights that were either late or canceled in Brazil in the recent past. So here in the first column usually the column, like I mentioned, has the name of the data, so an associated name that will describe what's going on here in that column. But we only have one type of data. We should have only one type of data per column. And here we start with a total that should not be here. Also there is the name of the city and the acronym of the state. So we should not have these two data, these two types of data coexisting in the same column. We need to split it so that we have two columns, one that will talk about the cities and another one that will show us the acronyms of the states maybe. And here the same thing. We have the absolute number of flights that were late and we have the flights that were canceled, and then we have the percentage of flights that were late and canceled.

Again, we need to separate these two numbers, separating two different columns so that we have only one column per type of data. This will be useful because if we need to make a calculation, say sum, or division, or average, or the top number, we won't have anything else that will compete with the data within the column. So, it's really important to have only one type of data so that we can make these kinds of calculations. So, like I mentioned each column has only one type of data and of course duplicated rows should be removed, otherwise you're adding up values that should not be added. But also watch out for misleading typos, right? "door" is still different than "Door" with a capital D. Those are two different entities for a computer. Although a human being might not recognize the difference between "door" and "Door" and might not think that is a big deal, that is a totally different entity for a computer.

So, it's really important that we maintain the same kind of typing for all of the values that we have on our data set. So, if we want to do all caps or all lowercase but we need a standard that we apply for all of the values. The same for this one which has an empty space before " Door" so empty spaces that are trailing or after the words, before the words, those are also counted as different entities in a column. So, we need to, we need to make sure that we have a standard typing for all of the values in the data set. So that's why it's important in the cleaning phase to remove all of the spaces or fix the typos so that the data in the column it has integrity. And like I mentioned as well the first row, usually has the name of the columns and usually we avoid including aggregated data like totals in the raw data source. That's because we're going to use software that will process the data, say apply formulas that will add so we don't need the total to be adding or being on top of the data, the granular data, the raw data that we have. So what we do usually is to remove all of the totals, and I'm gonna show you examples on the data cleaning class of how we can make sure that those values are removed so that we have a clean data set to make calculations later.

And what is the most common format? Usually when we talk about databases we're really talking about CSV files most of the time. Or we want to have a CSV file because it's an open format, it's plaintext, it's compatible with almost all if not all of the processing, data processing applications. And CSV stands for comma-separated values or even characters-separated values. To create a CSV, they also call it TSV if it's a different

separator, you can create it really easily. A CSV file is just values that are actually separated by a delimiter. And the delimiter, it could be a comma, or it could be a semicolon, or it could be this character that's called a pipe, or even could be the tab. So when you press tab and it gives that space, it could even be that character creating your CSV or if it's a tab usually you have a convention that you're going to name it a TSV file. So you just type in, you could create, you know in a text editor you just typing the names and then you put a comma. And usually what happens is this this first row here, this will be the columns row and when you create a new line, the new line will be the new row on our data set, on our database. And then we have the value here, the first value will be for the first column, the second one for the second one, and so on and so forth.

So really look at what we're doing here with this is creating something like this. So this code here the CSV code here, will actually create when we visualize in a spreadsheet program like Google Sheets, will actually see the first row here and then the next row here with the values for each one of the columns. That's why it's super important for CSV files to have the same amount of elements in every single line. So it doesn't get mixed up, right? So it doesn't have, if you have a file that has seven columns like this one, the second line needs to have seven values as well. Even if it's an empty value it needs a comma there indicating that's an empty value. The most common delimiters are, like I mentioned comma, semicolon, pipe and also the tab when you press the tab on your keyboard. So when you create CSV files, fortunately you don't have to type all of the the values yourself on on a text editor. Most applications will export the CSV files for us. So just pay attention to the delimiter that is being used to generate the CSV file. Sometimes you trying to import a CSV file and then if something goes wrong it didn't recognize the delimiter depending on the country where you are. There is a convention on the kind of delimiter that you use, for example in the United States it's a comma. But in Brazil the standard it's the semicolon. So, that creates some confusion sometimes so you might want to check that out too if doesn't work in your case.