

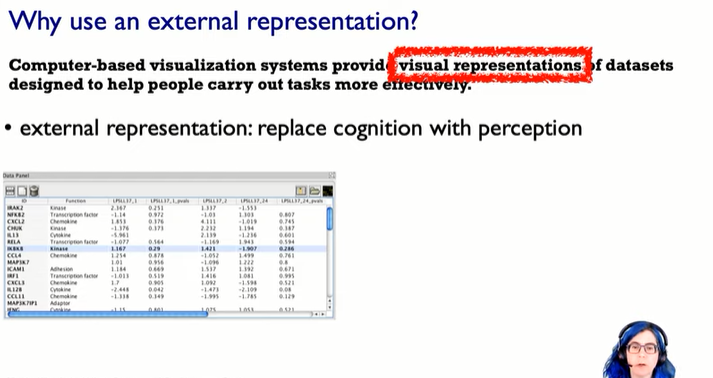
Augment human capabilities (make (something) greater by adding to it;)

We don’t know what questions to ask in advanced… i.e., there might be only one question that we need to ask regarding the data which might help us understanding the data completely…. But for certain scenarios we might not be able to understand the data completely even after asking 100s or even 1000s of questions

Basically, in conclusion I want to say that we number of questions to be asked changes according to the data provided and can’t be predetermined hence a human in the process is required

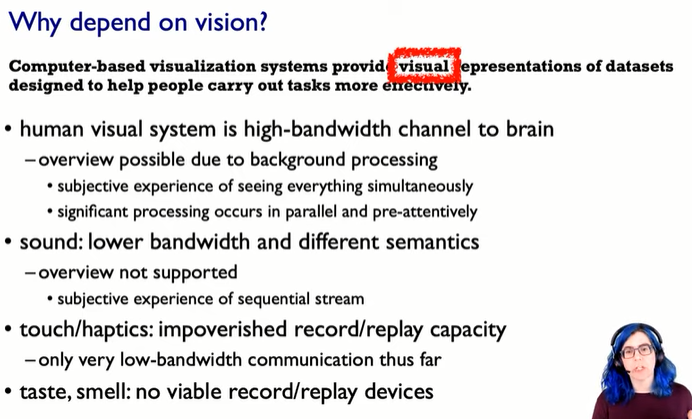
Possibilities 🡪 Human in the loop

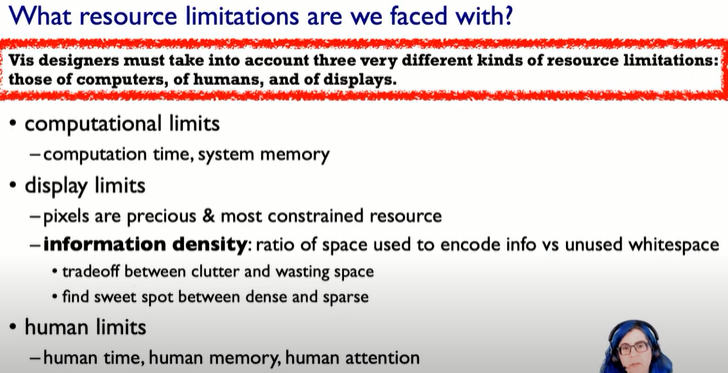
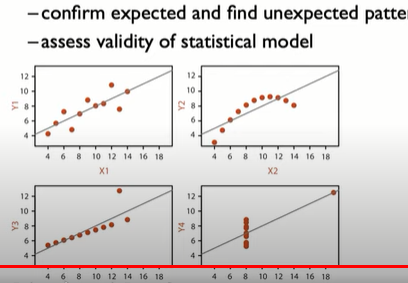
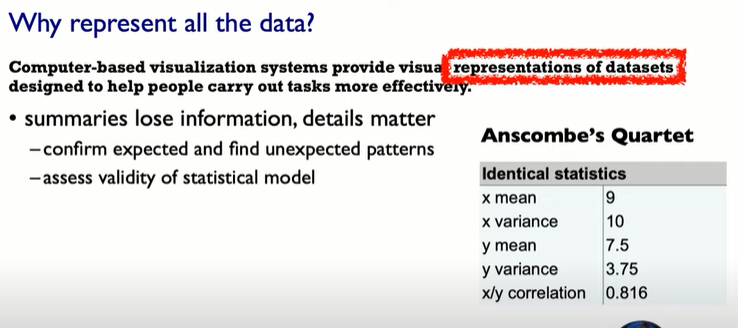
Stepping stone 🡪 You might want to develop a model and thus to dive deeper you may require a human in the loop….



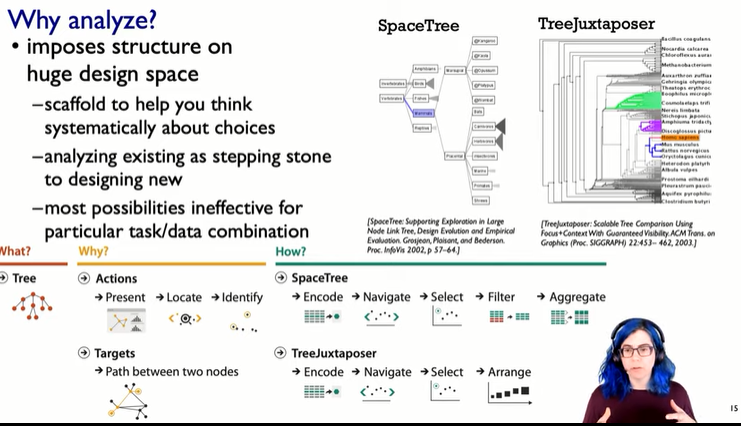
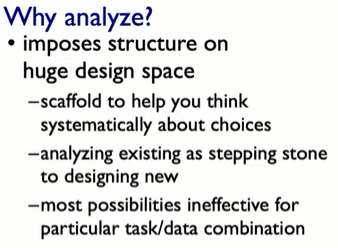
We do it as to replace cognition with perception in order to free up cognition and use the brain for higher level questions than the really low level

* As we run out of the short term slots and we keep on forgetting the just read items thus making it difficult for us to do multiple comparisons

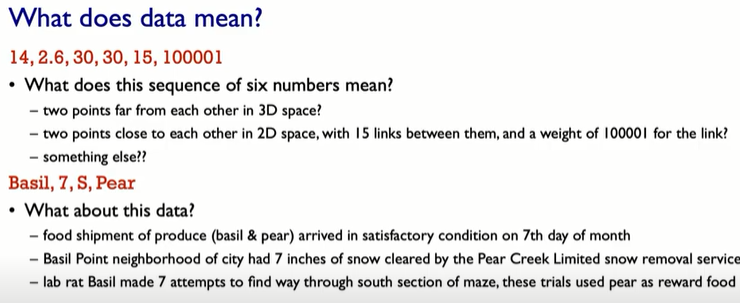




Computational ,display and human limits

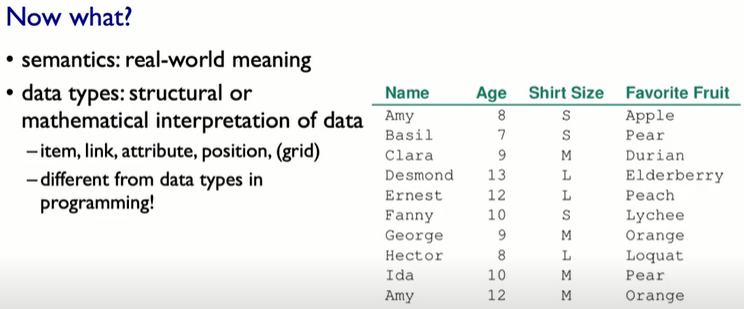


2nd video



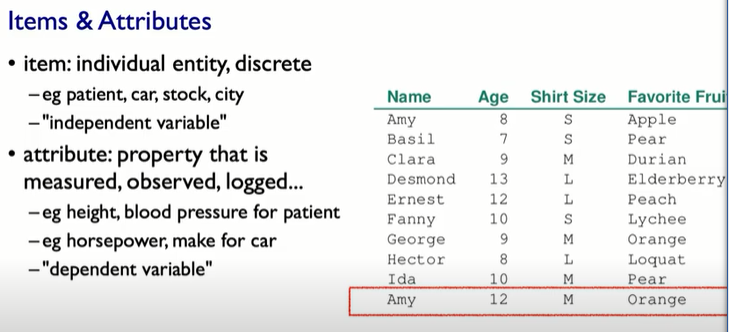
One person’s meta data others person data….

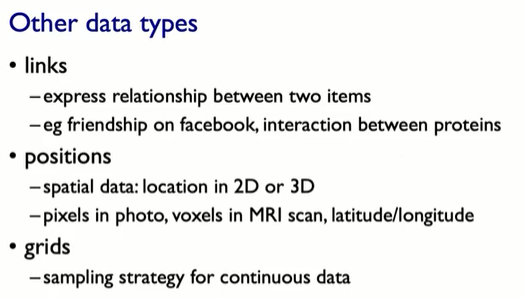
Here datatypes don’t mean the programming data types but they mean the semantics(real world meaning) of what particular entries in some data file

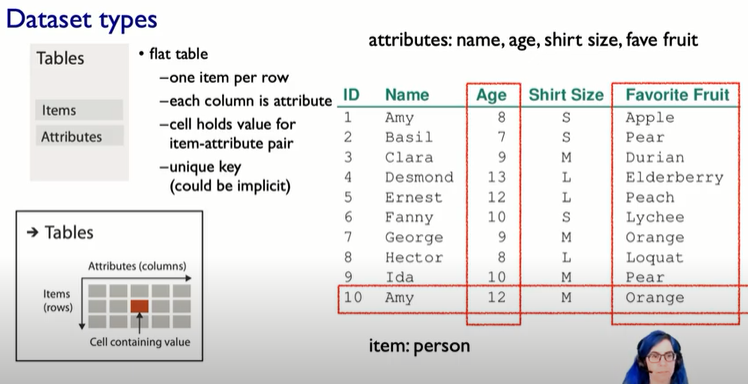


Item 🡪 Means individual entities that are discrete like a patient in hospital (independent variables)

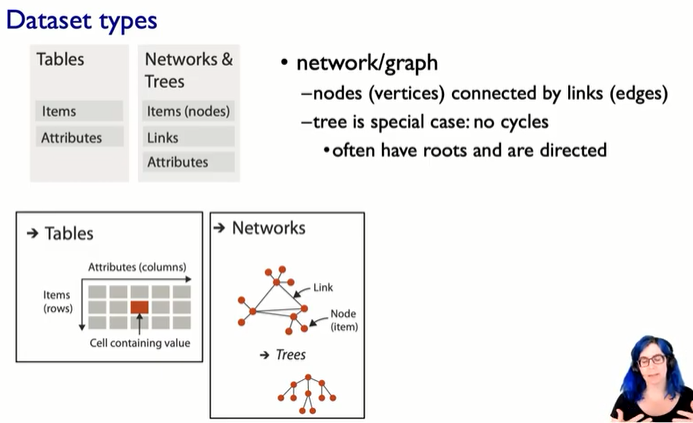
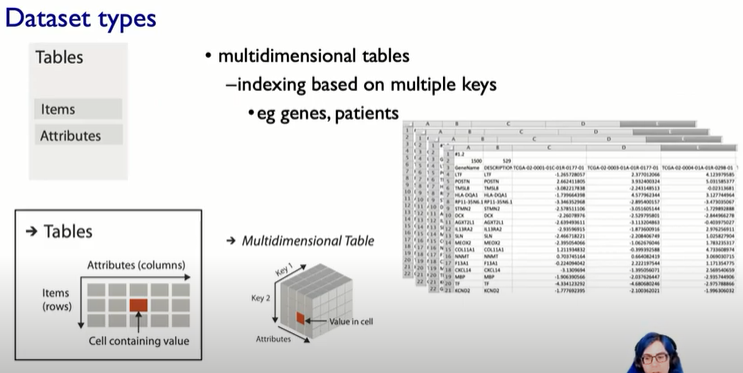
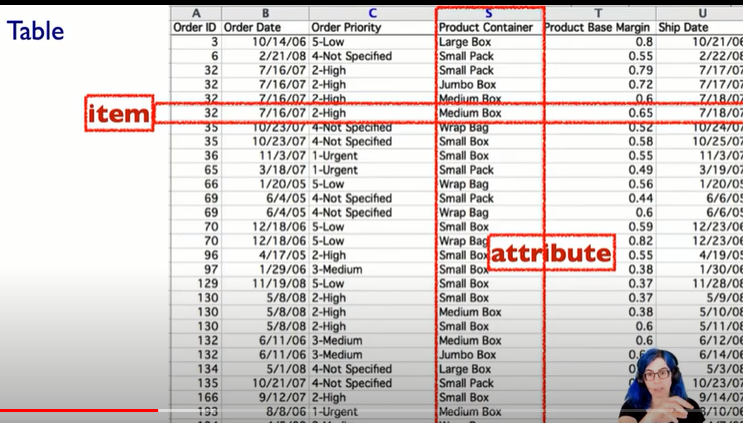
Attributes 🡪 dependent variables(property of an item)







Implicit is written above for unique keys as numbering 1,2,3,4,… is used but for explicit unique key we might use customer id basically a unique key which we assign to them



Following are some important attributes examined in this chapter:

* **Node attributes**—Not all nodes are the same. Which has the most connections? Which is the oldest? Which has the biggest change? Use visual attributes such as color and size to reveal node data.
* **Link attributes**—Which links have the strongest connections? What are the different types of links? In what direction do the links point? Visual attributes such as color, line width, and arrows show link data.
* **Labels**—Who is that? What is that node? Perhaps one of the most important elements (particularly in smaller graphs), labels should not be an afterthought, but carefully planned so that they are clear, legible, and informative.



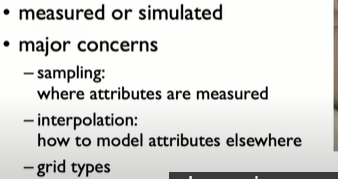
In spatial dataset attributes are tied to different spatial locations or regions.

CELLS : contain values from some sort of continuous spatial domain, you know .

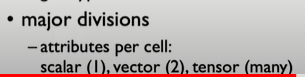


Here cell contains values of temperature , pressure ,wind velocity from continuous domain but for a specified longitudinal and latitudinal location.

Cells has a continuous domain but contains discrete values attached to it

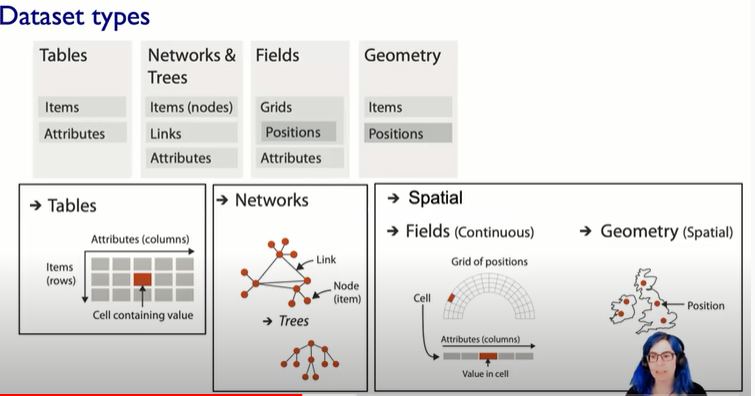


Interpolation is **a statistical method by which related known values are used to estimate an unknown price or potential yield**

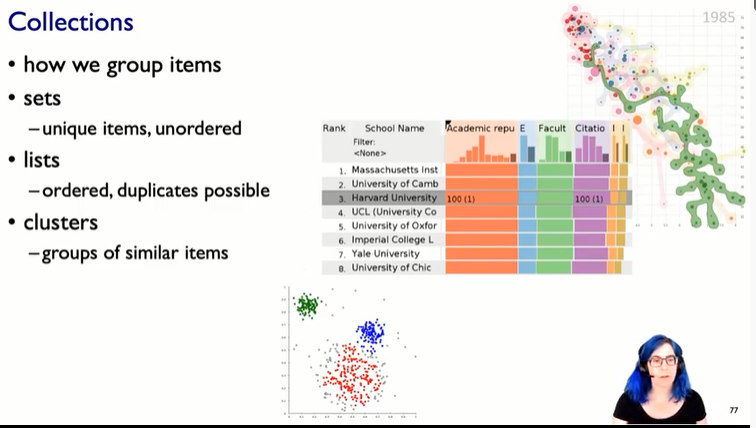


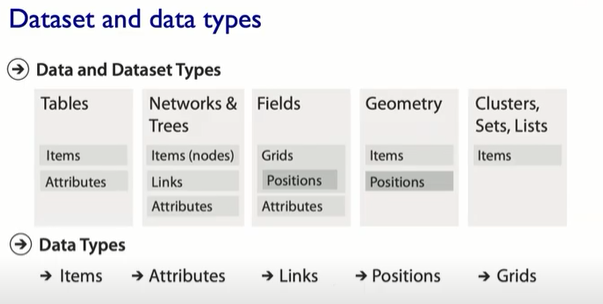
Vector 🡪 contains 2 data in each cell.

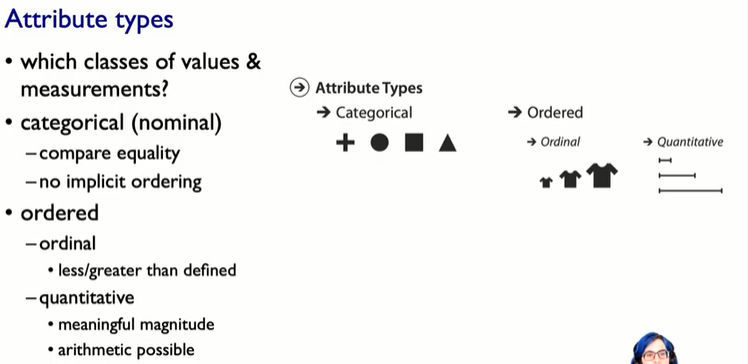
Tensor🡪 contains more than 2 data in each cell.



In computer graphics the geometry you take as given to you and then you think of how to render that geometric information according to some sort of synthetic idea of where the cameras and the lights are . In visualization , we do worry about drawing that picture , which is equivalent to rendering the graphics , but the key point is that the geometry is the result of a design decision by the visualization designer . So we could for example take one geometry in , and then make choice about things how to abstract that . So when we’re thinking about data abstractions . We are given geometry in some case ,or we might compute it, and then we might choose to abstract that geometry , depending on the task that we’re dealing with





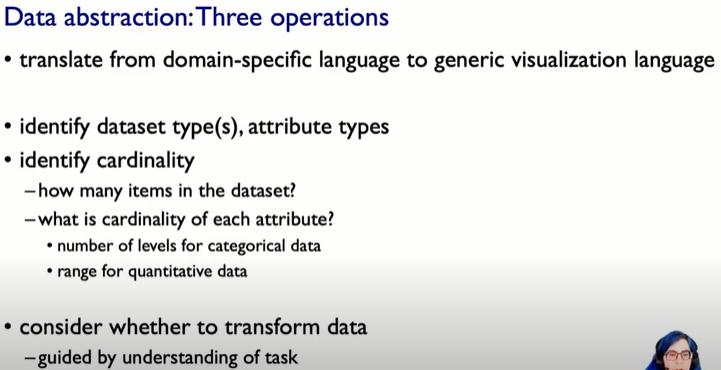


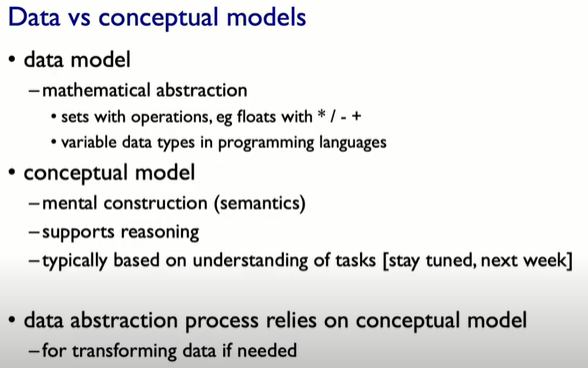


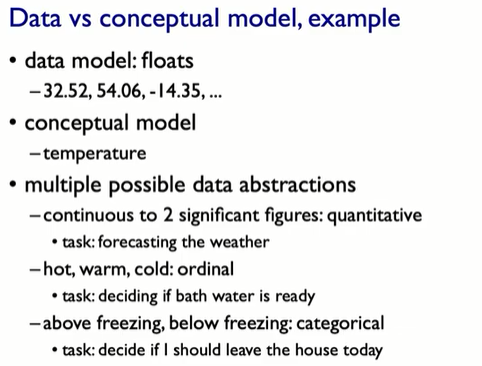
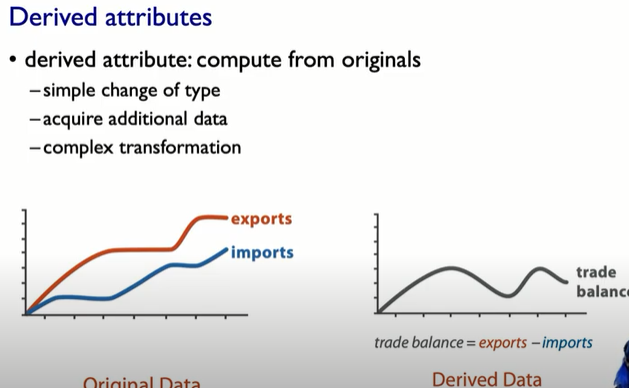
Quantitative quantity 🡪 Attributes on which performing arithmetic operations makes sense

Ordinal 🡪 things which can be ordered and but performing arithmetic operations doesn’t make sense

Categorical 🡪 just distinction

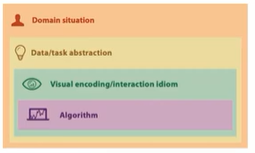
 Transforming the data might help to visualise in a better way…

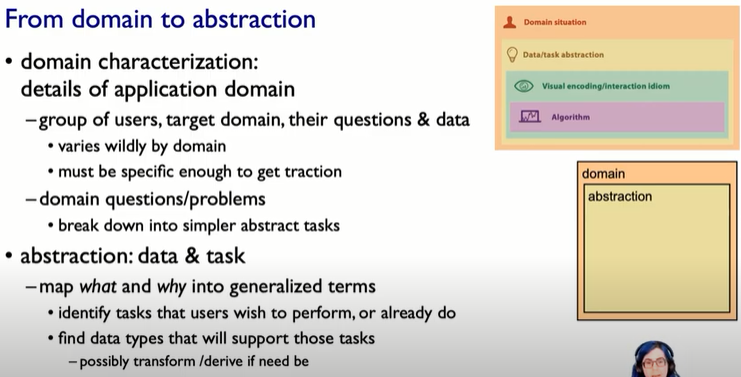
Quantitative data 🡪Range  

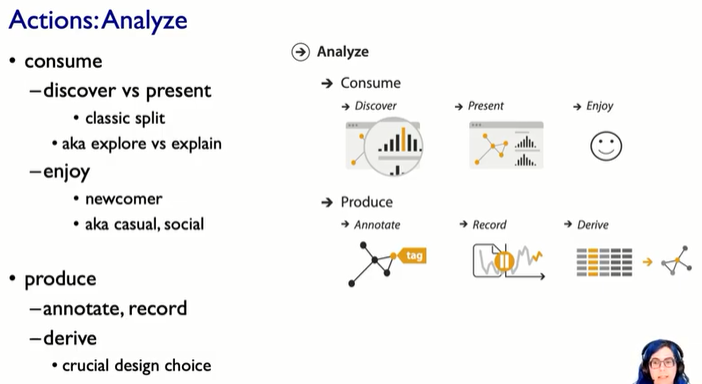
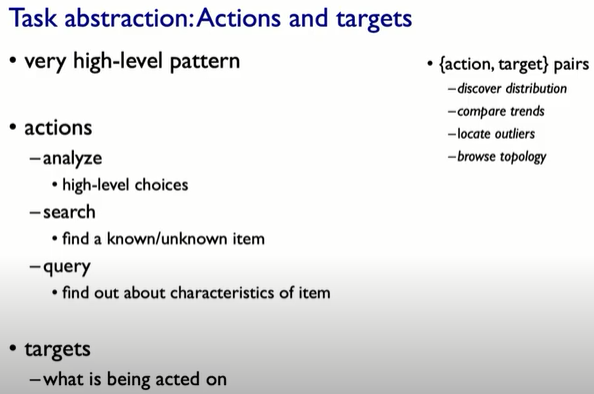
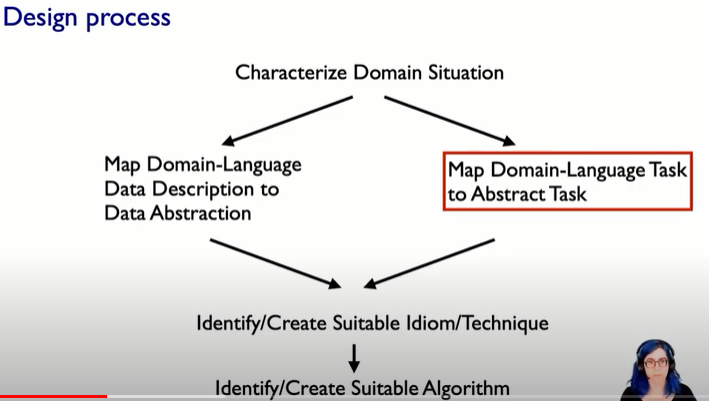
Trade balance(dependent) = exports(independent) – imports(independent)

Reduces cognitive burden

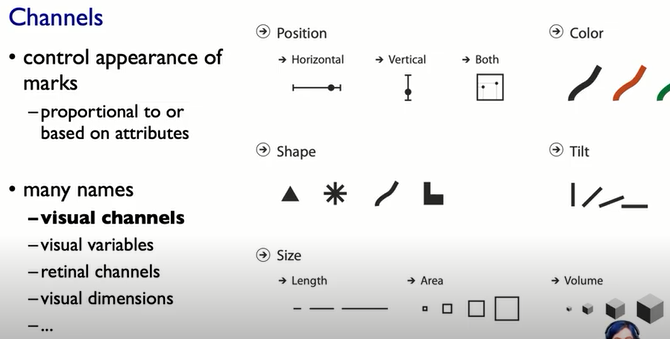
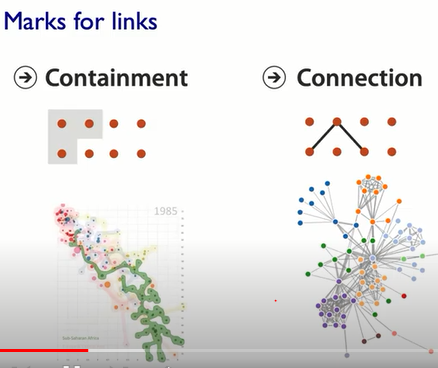
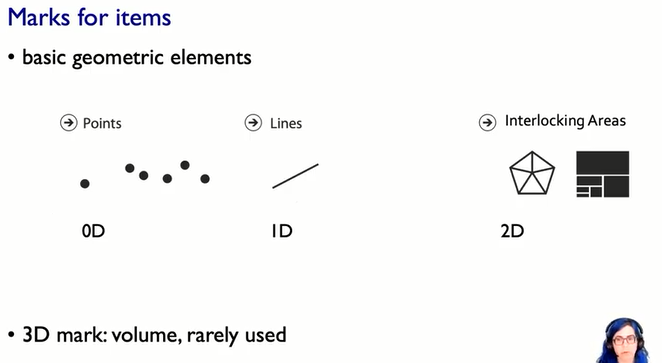
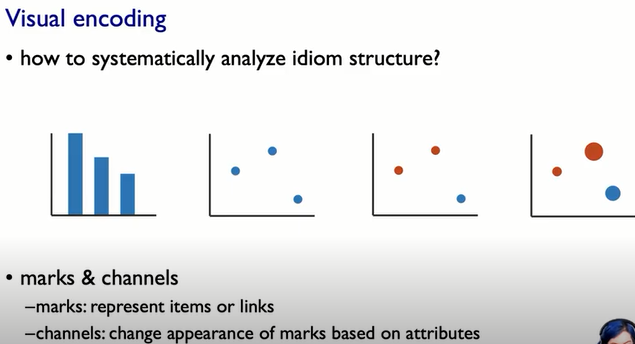
2nd Video

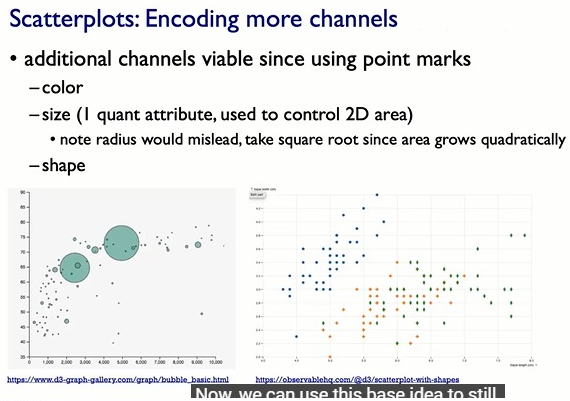
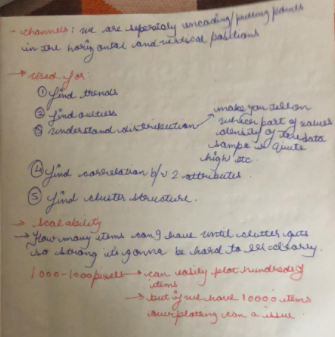
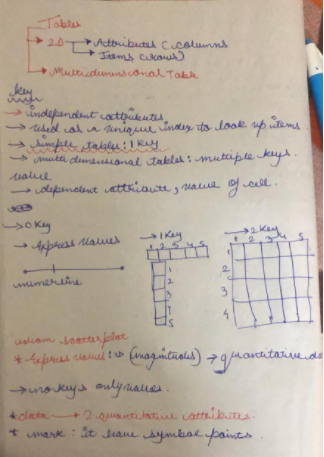
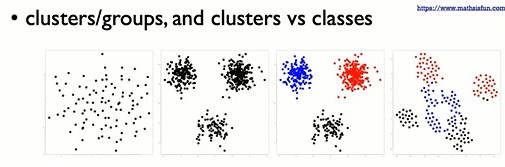
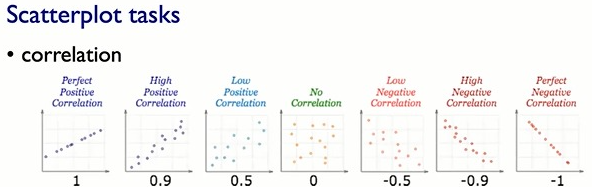


Abstraction is **a technique of hiding unnecessary details from the user**

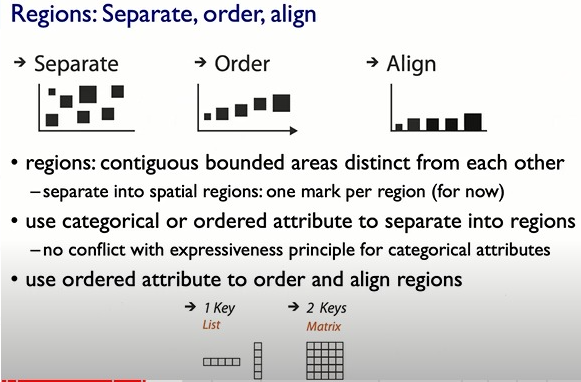


**Video -6**

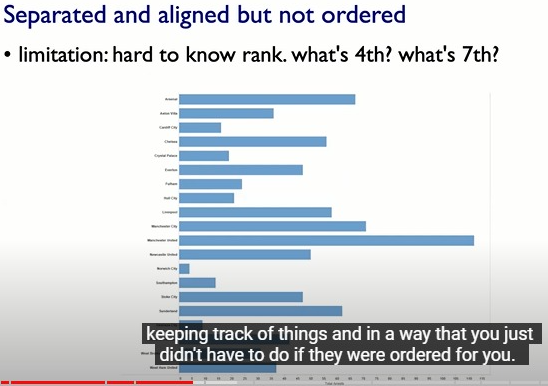
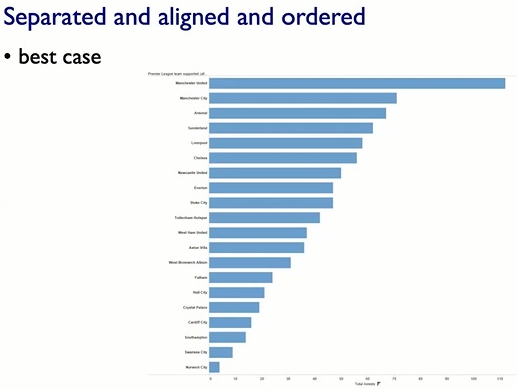


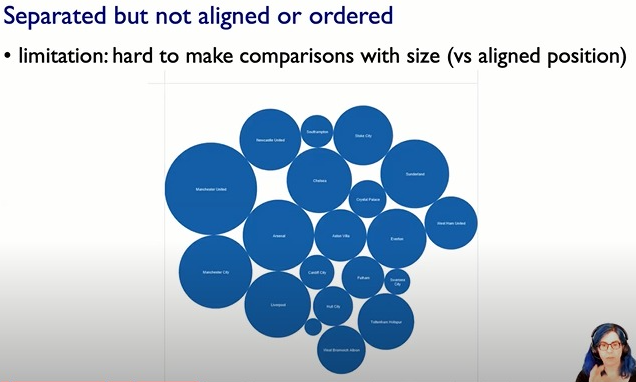
**It's important to take square root for size encoding as these are encoded using square of the quantitative data**

**cluster --> Density/colour{To draw idea regarding different groups in data}**



**We want to use regions to separate .order ,align**

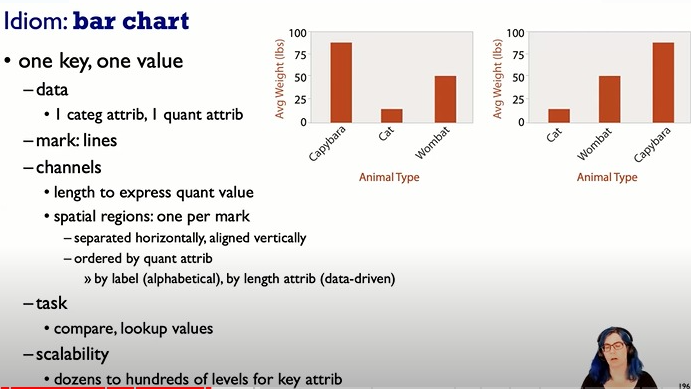
**One mark per region means the plotted points don't overlap**



**Size coded circles**

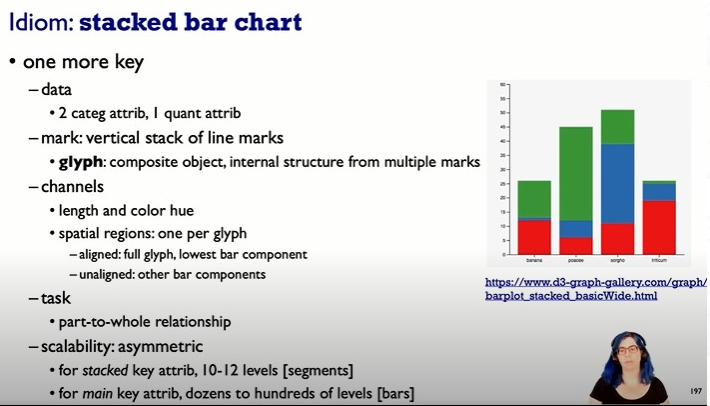
**These are even harder to know as only via looking and no scale aligned you have to figure out which one is the largest ...then look for smaller and so onn**

**Hence harder to make fine grade comparisons**



**Bar chart is used for --> Categorical or any ordered data like(time.date)**

**High precision comparison**



**High precision comparison can be performed to only the lowest bar component as others are not having common base value(quantitative x value)/not aligned**

