DIT635 - Category Partition Method Activity

You have created a service intended to find all instances of a **pattern** in a **file**. You want to now design system-level test specifications, then apply the category partition method to limit the number of specifications.

`find(pattern,file)`

This pattern can contain spaces and quotes. For example:

`find(“john”,myFile)`
Finds all instances of john in the file

`find(“john smith”,myFile)`
Finds all instances of john smith in the file

`find(““john” smith”,myFile)`
Finds all instances of “john” smith in the file

Use the category-partition method to identify a pool of valid test specifications.

1. Identify the **choices** that you control when testing.
    a. For each input variable, what aspects should you consider when testing?
    b. Are there any environmental factors that you can control that will also impact the outcome of executing this service?
2. For each **choice**, identify a set of **representative values** that could lead to different outcomes of executing this service.
3. Impose **constraints** on the choices to reduce the pool of test cases.
    a. error constraints identify values that should result in an error no matter what other values they are paired with.
    b. single constraints identity values that should result in normal execution, but should be tried once because they have the potential for error or strange behavior.
    c. if-constraints identify values that should only be used if other choices are set to particular values (“if choice X = THIS, then choice Y = THAT)