## Calculation 1:

## In the segment <u>"Corporate"</u>, which <u>State</u> receives the most orders from the category <u>"Technology"</u>?

main()
runs the program and calls
the functions
No inputs
No outputs

load\_superstore\_df(csv\_file)
reads file and converts into
data structures
input: csv\_file (string)
output: data (list of
dictionaries)

organize\_to\_corporate(data)
this is will to focus on rows
that involve segment
"Corporate"
input: data (list of
dictionaries)
out: corporate\_data (list of
dictionaries)

organize\_to\_technology(corporate\_data)
this will focus on rows that only involve
the category "technology"
input: corporate\_data (list of
dictionaries)
output: corporate\_tech (list of
dictionaries)



max\_state(num\_states)
this will find the state that
recieves the most orders
from the dictionaries
input: num\_states
(dictionary)
output: max\_state(string)

number\_of\_states(corporate\_tech)
this will iterate over the data and count
the number for each state
input: corporate\_tech (list of
dictionaries)
output: number of states with the
corporate and technology (dictionaries,
keys are states)

## **Calculation 2:**

## In the category "<u>Furniture</u>", what is the most requested or used <u>Ship Mode</u> for orders placed by the "<u>Consumer</u>" Segment?

main()
runs the program and calls
the functions
No inputs
No outputs

load\_superstore\_df(csv\_file)
reads file and converts into
data structures
input: csv\_file (string)
output: data (list of
dictionaries)

organize\_to\_consumer(data)
in this function, it will edit the
rows to focus on on the
segment "consumer"
input: data (list of dictionaries)
output: consumer\_data (list of
dictionaries)

organize\_to\_furniture(consumer\_data)
this will focus on shorting the rows that
only involve categroy "Furniture"
input: consumer\_data (list of
dictionaries)
output: consumer\_furniture (list of
dictionaries)

