SETTING UP THE PROBLEM

→ Import pandas
→ We are defining a function called calculate_demographic_data
  → calculate the demographic data of this input data
  → then read in the data
    - from a "comma separated value" (CSV) file ~Excel
    - the data we are performing calculations on is elsewhere → in an external CSV file
    → this imports it into a Pandas dataframe (df) ← called a variable
→ So - by the start of question 1, we have
  → a function called calculate_demographic_data
  → CSV data which has been imported into this in a numpy dataframe → equal to a variable called "df"

QUESTION 1

```
# QUESTION 1:
# "How many of each race are represented in this dataset?
# This should be a Pandas series with race names as the index labels."
```

[..., ..., ..., ...]
↑ number of British people
← number of Indian people

→ From the 19... census data we just imported
→ we count the number of people of each nationality
→ they get their own element in an array
→ so, we store that array in a variable - and set it equal to → the race columns of the database we just imported

→ we run the value_counts method on that column

QUESTION 2

QUESTION 2: "What is the average age of men?"

- -> we want just the men
- -> and then the average value for the ages of those men
- -> you can't calculate the average age, and then extract only the men from it, you have to extract the men first
- -> and then to calculate their average ages
- -> you could look at the dataset, take all of the ages, and then only the men
- -> or you could take only the men, and then look at the ages <- use this one
- -> so we define a variable which extracts the sex column from the data frame
- -> and then we do the same with the ages
- -> then we calculate the mean of the sex column where the sex is male
- -> it's the process of looking at the question, and being able to convert it into a Python expression
- -> this is statistics -> it's like Venn diagrams, where we have the age and the sex and we are taking the mean of the age column which intersects with the population of males
- -> the first part of this is reading the question and extracting the columns with the data which we want in their own variables
- -> and then combining them into one expression which calculates the value we want

QUESTION 3

QUESTION 3: "What is the percentage of people who have a Bachelor's degree?"

- -> paying attention to the language in the question
- -> it's the entire approach of, "how do we get the words in this question into an equation / expressions in Python?"
- -> we want a percentage
- e-> of all the people
- -> only the ones who have a bachelors degree
- -> so it's the number of all people where the degree is bachelors
- -> over the total count of all of the people
- -> we first want to extract the number of people with a BSc
- -> and then we want to divide it by the total number of people
- -> to calculate the number of people with a BSc
  - -> we extract the 'education' column
  - -> for which the value of an element is 'Bachelors'
  - -> and then we sum it up
  - -> this gives us the total number of people with a bachelors degree, which we store int he variable x
- -> then we calculate the amount of those people as a percentage of all of them
  - -> we store this in the variable y
  - -> this is as a percentage of all of the values in the database for education
- -> then we round the number to one decimal place -> the question asked for this during testing

"What percentage of people with advanced education" QUESTION 4

(`Bachelors`, `Masters`, or `Doctorate`) make more than 50K?"

- ***-> initial thoughts***
  - -> we want the columns in the data frame for which the person's education is a BSc, MSc or PhD
  - -> and then we want the subset of those people who earn above 50k
  - -> so we first want to extract the people with higher education
  - -> and then we want the population out of those people -> the count, for which the salary is > 50k
  - -> then we want one number as a percentage of the other
  - -> and we want it rounded to the nearest 10th
  - -> we first want to extract the columns of people with higher education
  - -> then the count from that for which the salary is above a certain expectation
  - -> and then one number as a percentage of the other
  - -> then to found it

- ***-> approach used***
  - ***-> to extract the people with higher education***
    - -> we define an entirely new data frame -> just for the people who have higher degrees
    - -> this gets rid of all of the people we don't want
    - -> without having to loose information that would be lost if we just counted the number of entries in a column, for example
    - -> so now we have a data frame which is just for the people with higher duration

  - ***-> we want to extract the number of people who are earning above a certain salary***
    - -> we store this number in a new variable
    - -> we take the data frame which just involves the people who have higher education (degrees)
    - -> and then we take the salary column, the values for which the salary is >=50k
    - -> so now we have the people with degrees that earn more than this salary requirement
    - -> now wee want to count them -> so we apply the sum method to it
    - -> then set the entire thing equal to a variable

  - ***-> then for the percentage***
    - -> so we have the number of people with higher education above a certain salary range
    - -> then we take that as a percentage of all of the numbers in the data frame which just contains the number of people with higher educations
    - -> we store this in another variable, y
    - -> then we round the entire thing to a tenth of a decimal place

QUESTION 5

"What percentage of people without advanced education make more than 50K?" (lots of adp.

- ***-> thoughts***
  - -> this is the same as the previous question -> just slightly different
    - -> case of spot the difference between this and the last question
  - -> same as the last question
    - -> we want a percentage
    - -> to a 10th of a decimal place
    - -> we are taking a subset of the entire data frame <- people without an advanced education in this

case
        - -> the condition is the same -> in the sense that we want more than 50k
    - -> different to the last question
        - -> the previous case was people without one

- ***-> approach used***
    - -> we can reuse the solution to the previous question
    - -> but just change the data in the sub-frame we are selecting
    - -> from the people with higher education, to the people without it (!= compared to =)
    - -> and change the names of the variables, because we are going to want to return this data and avoid overwriting the values which the previous variables store

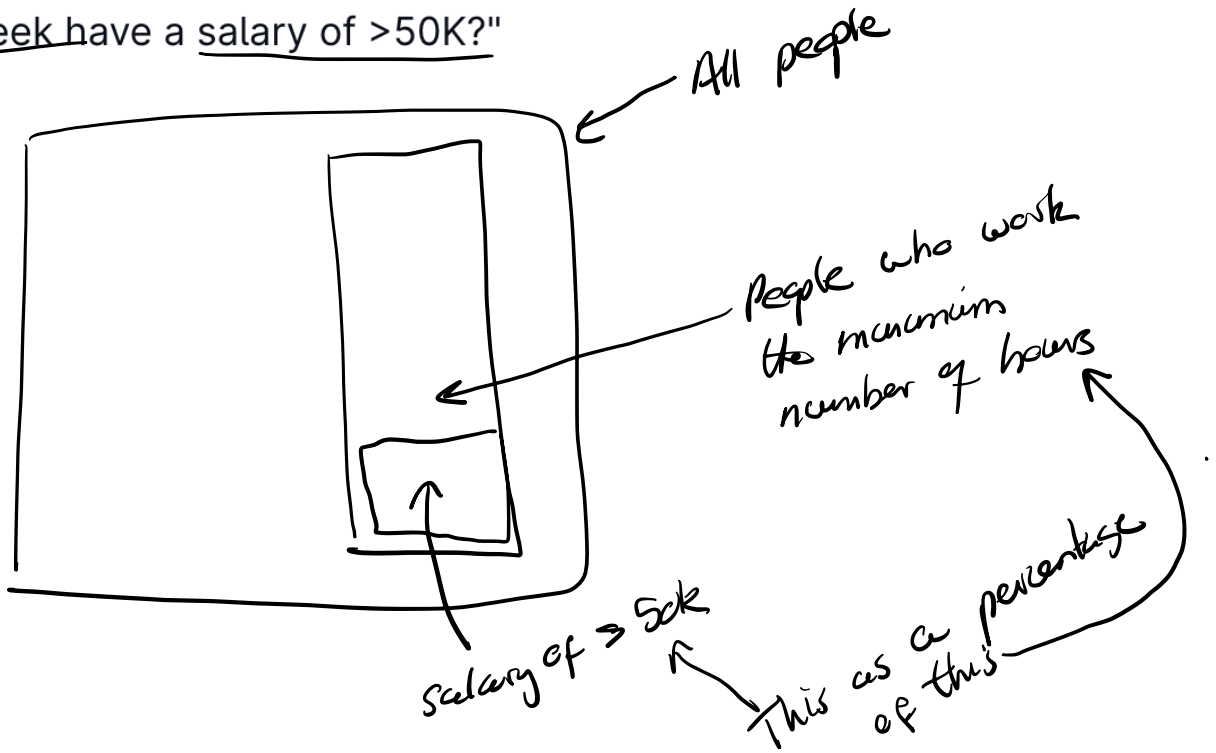
QUESTION 6

"What is the minimum number of hours a person works per week (hours-per-week feature)?"

- -> we want to take the entire data frame, extract the values in the hours-per-week column, and then take the minimum
- -> we want everything to be to a 10th of.a decimal place -> but if it's hours per week we should be fine
- -> we we set it equal to a variable
- -> take the entire data frame, then just the column which stores the amount of hours people worked per week
- -> then jut take the minimum from that
- -> we are using the minimum method on it

QUESTION 7

"What percentage of the people who work the minimum number of hours per
week have a salary of >50K?"



percentage = $\frac{small}{large} \times 100$

large = people who work min. number of hours

→ answer to previous question was the maximum number of hours people worked / week

→ so total dataframe, for which the number of hours per week worked (the value this is equal to) equals the answer to the previous question

→ we can set this equal to a new variable – which just contains the entries where people worked the maximum number of hours

→ count / sum up the number of those entries ← then this is "large"

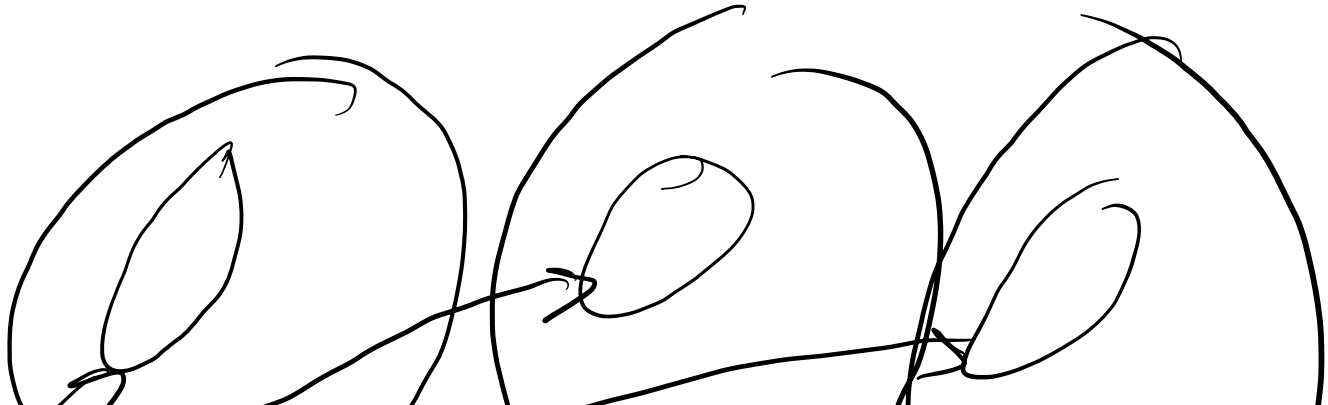
small = → the sum of the number of entries in this dataframe for which the salary >= 50k

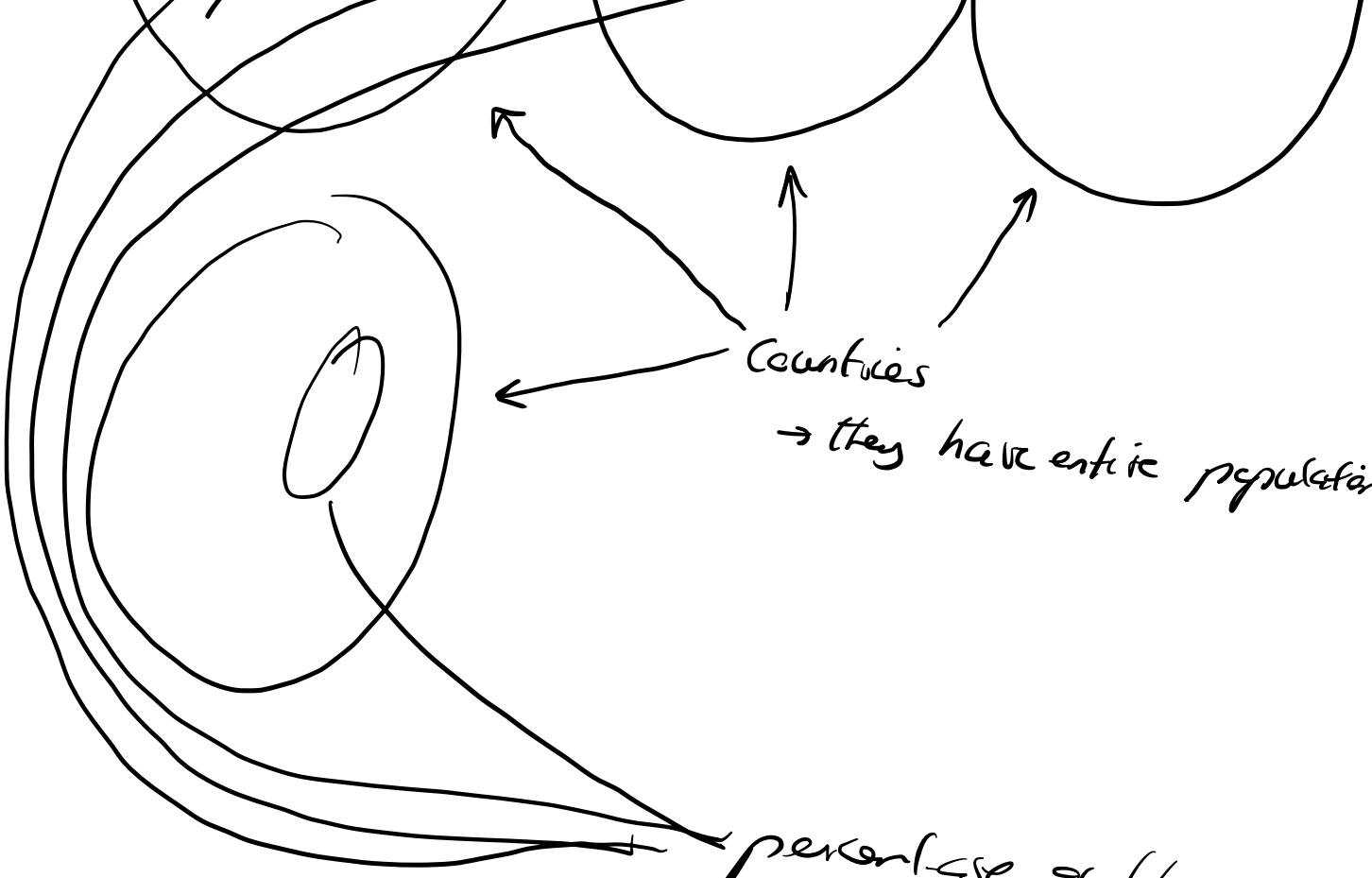
then we do $\frac{small}{large} \times 100$, store it in a variable

→ then round the value of that variable to a 6th of a decimal place

QUESTION 8

"What country has the highest percentage of people that earn >50K and what is that percentage?"

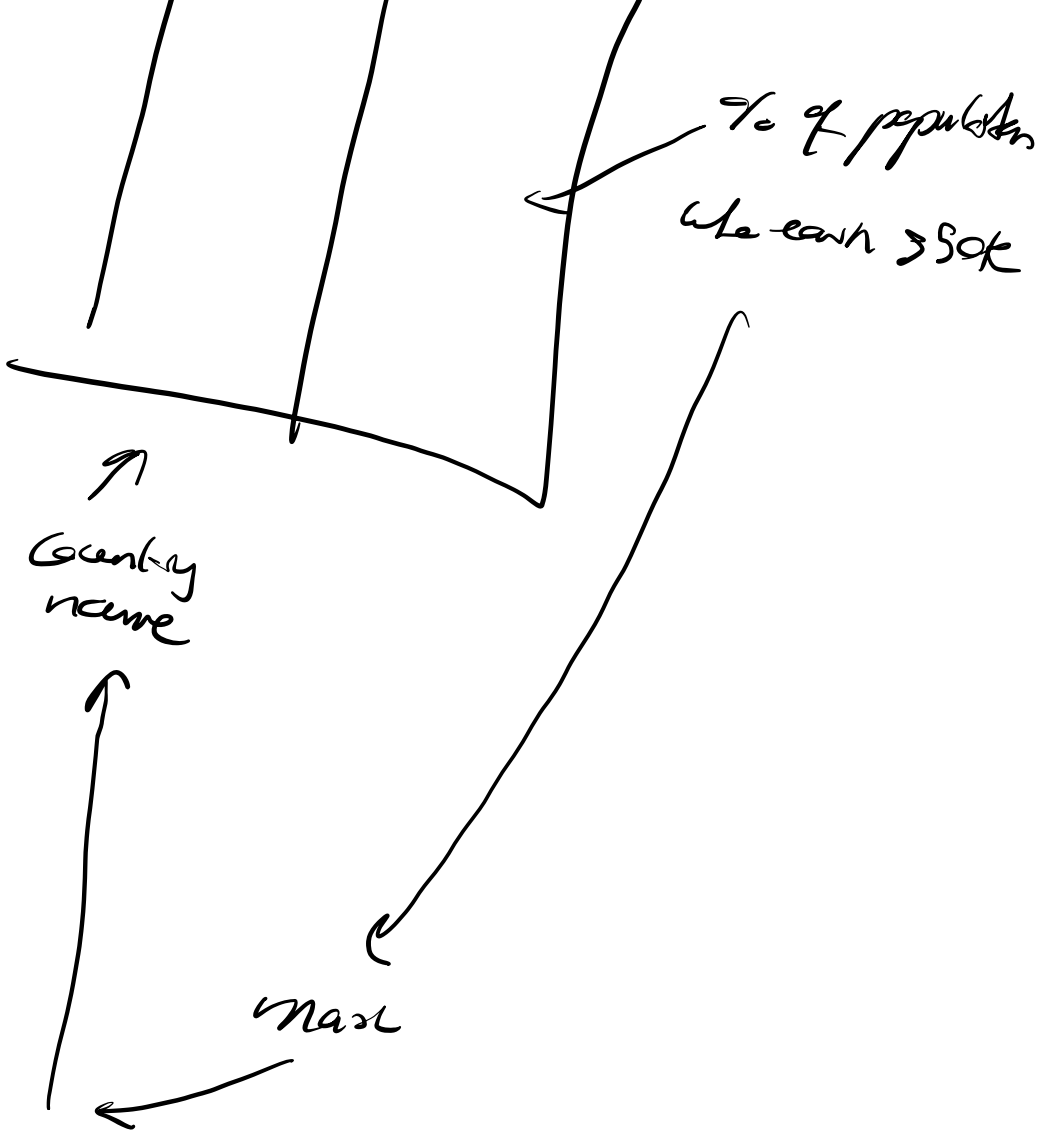




[... , ... , ... , ... , ...]

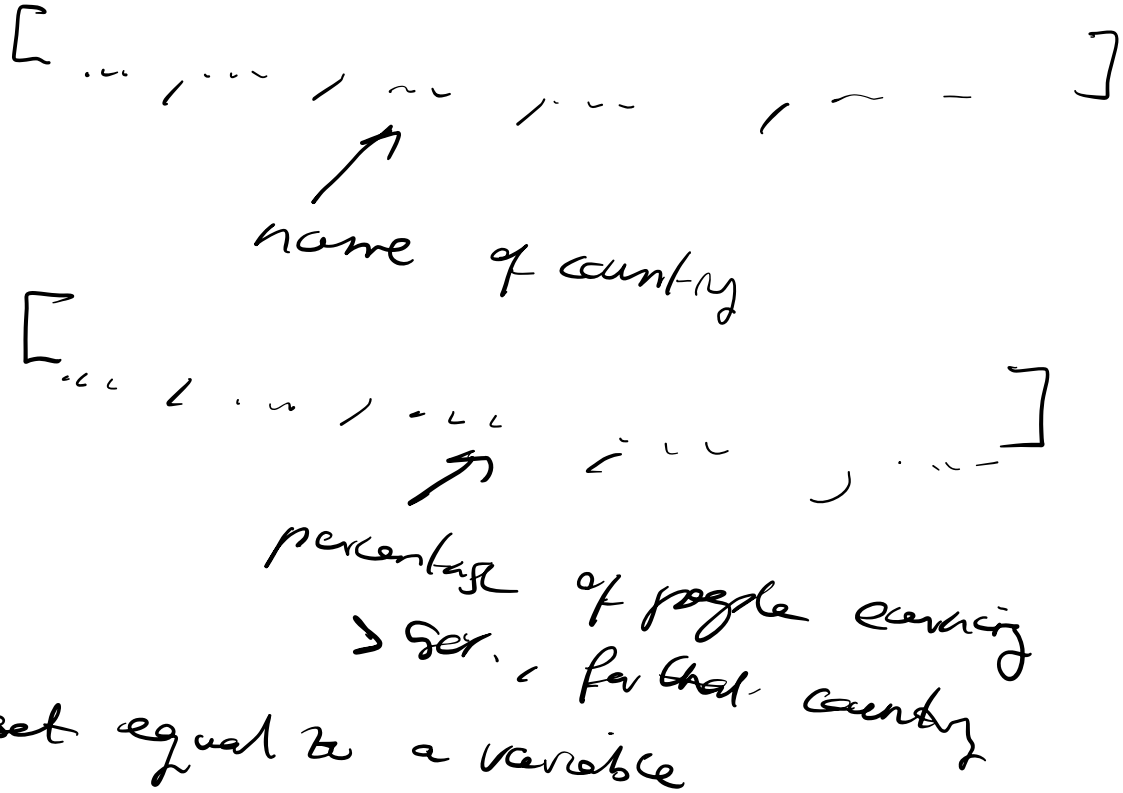
→ you can take them, put them into an array
→ set that array equal to a variable
→ calculate its maximum

or another dataframe



→ You could

- Make an array which contains the list of countries
- for i in this list of countries, give me the sum of the number of people who earn > 50k
- calculate that as a percentage of the total population)
- do that for each country → and as we go populate this second array with those values



→ set equal to a variable
→ then we return the index of that array for which the percentage is maximal
→ we extract that percentage, round it to the nearest 1 or 2 decimal place and return the element in the array of country names which has that index

QUESTION 9

"Identify the most popular occupation for those who earn >50K in India."

→ define a new variable
→ set it equal to a new dataframe
- those who earn > 50k and are in India
→ we are selecting these x2 columns from the bigger database (we have controlled for this condition, and this condition)

② → we take the previous database ↑ and set it equal to a variable
→ then, we count the number of people per occupation → we are returning the most frequently occurring one in that set, storing it in a variable and returning it

→ Then running unit tests on the results until they pass, pushing the project files to GitHub and submitting a link to that repository at the project brief URL