Q1. Explain the difference between greedy and non-greedy syntax with visual terms in as few words as possible. What is the bare minimum effort required to transform a greedy pattern into a non-greedy one? What characters or characters can you introduce or change?

Ans:

**Greedy match tries to match as many repetitions as possible and non-greedy match tries to match as few repetitions as possible. For conversion a quantifier such as ? can be removed to obtain a non-greedy pattern from a greedy pattern.**

Q2. When exactly does greedy versus non-greedy make a difference?  What if you're looking for a non-greedy match but the only one available is greedy?

Ans:  **greedy is preferable when you want to get as many matches as possible and non-greedy can be used in the opposite scenario. Greedy can be converted into non-greedy with the elimination of a quantifier ‘?’**

Q3. In a simple match of a string, which looks only for one match and does not do any replacement, is the use of a nontagged group likely to make any practical difference?

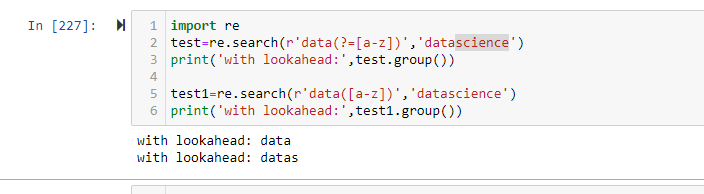
Ans: **yes it may make a different as non tagged group doesn’t get saved in memory hence it saves the memory occupation.**

Q4. Describe a scenario in which using a nontagged category would have a significant impact on the program's outcomes.

Ans: **the reason for using the nontagged group is to save memory as the regex engine doesn’t need to store the groups in the buffer**

Q5. Unlike a normal regex pattern, a look-ahead condition does not consume the characters it examines. Describe a situation in which this could make a difference in the results of your programme.

Ans: **you can use lookahead when you don’t want the output to return lookahead portion present in search strung but want to use it to match pattern which is followed by a particular section.**



Q6. In standard expressions, what is the difference between positive look-ahead and negative look-ahead?

Ans: **positive lookahead when a search string is followed by <lookahead\_regex> and negative lookahead when a search string is not followed by <lookahead\_regex>**

Q7. What is the benefit of referring to groups by name rather than by number in a standard expression?

Ans: **it improves the readability and understandability as you can see which part of a regular expression is being matched.**

Q8. Can you identify repeated items within a target string using named groups, as in "The cow jumped over the the moon"?

Ans:

Graphical user interface, text

Description automatically generated

Q9. When parsing a string, what is at least one thing that the Scanner interface does for you that the re.findall feature does not?

Ans: **upon doing painstaking research, I have ascertained that there is no scanner interface in python, rather there is an input function which is used to take user’s inputs in the form of string and can be type casted and saved in a variable.**

**Additionally, re.findall is used to find all the occurrences of a pattern in a text.**

Q10. Does a scanner object have to be named scanner?

Ans: **there is no scanner method in python like Java, instead it has an equivalent method called input() which can accept input and store it in a variable.**