Assignment 2 DFA and FST

1. Construct a deterministic finite automaton (DFA) that recognizes valid simplified English words. An English word, for this problem, is defined as a string that:
● Starts with an lowercase letter
● Followed by zero or more lowercase letters

   That is, the string must:

● Not start with a digit or punctuation
● Not contain uppercase letters after the first character
● Not contain spaces, digits, or special characters

   Output Format:

   If a string is accepted, output Accepted else Not Accepted.

● Accepted: cat, dog, a, zebra
   Not Accepted: dog1, 1dog, DogHouse, Dog_house, cats (starts with a space)

For visualizing the DFA, you can use https://pypi.org/project/visual-automata/ or https://github.com/rohaquinlop/automathon

2. You are given all the nouns from the brown corpus (brown_nouns.txt). You need to design a finite state transducer (FST) to generate the morph/grammatical features for every word in the corpus. Your output should look like the following:

   foxes = fox+N+PL (can be generalized as root+category+number)

   fox = fox+N+SG [where SG->Singular, PL->Plural, N->Noun]

   You need to take the following properties:

Name	Rule Description	Example
E insertion	e is added after -s, -z, -x, -ch, -sh before -s is added	watch/watches, fox/foxes
Y replacement	-y changes to -ie before -s	try/tries
S addition	-s is added at the end	bag/bags

   You need to ensure that incorrect words are not generated. You return an output "Invalid Word" in that case. Example: foxs = "Invalid Word"

There are techniques to decrease the number of states, try implementing them. Bonus marks for such techniques. [Don't try to code the rules, need to design an FST with transition table, input and output alphabets]