Create a script **Adventure.py**

1. (5 pts) Print a welcome message for your player
2. (5 pts) Ask player for their name and store it in a variable called *user_name*
3. (5 pts) Ask them "Which direction do you want to go, [*their name here*]? " and give them the option of **North, South, East, and West**, asking them to put in "N" for North, "S" for South, "E" for East, and "W" for West.
4. (10 pts) If they give you a string that is not N, S, E, or W, tell them that's not a valid response and keep asking them till they give you a correct response
5. (20 pts) For each direction ask them a different question of your choice (like "what's your favorite color" or "What's the airspeed velocity of an unladen swallow") and prompt them for a response. Ask them to enter their answer in all *lowercase.* This means you should have 4 questions. One for each direction they could choose.
6. (5 pts) Based on their response, print some kind of appropriate response. Something like "You got it right!", or "I like that color too", or "Wow, that was an easy one"
7. (5 pts) Ask them AGAIN "Which direction do you want to go for your second step, [*their name here*]? " and give them the option of **N,S,E,W** again (*remember, this is their second step, after their first step).*
8. (5 pts) If they give you a string that is not N,S,E or W, tell them that's not a valid response and *keep* asking them
9. (5 pts) If they go **South**, **East**, or **West**, Tell them that "Sorry, [their name here], you died" and add an additional death message.
10. (30 pts) If they choose North, they get to play a game/solve a puzzle. You'll create 4 puzzles (one for when their choices were **North** for the first step and then **North** for the second step, one for if they Chose **South** and then **North**, one for if they chose **West** and then **North**, one for if they chose **East** and then **North**). Below is a diagram to help with the logic.

First step → Second Step ↓	**NORTH**	**SOUTH**	**EAST**	**WEST**
NORTH	Puzzle 1	Puzzle 2	Puzzle 3	Puzzle 4

SOUTH	*DIE*	*DIE*	*DIE*	*DIE*
EAST	*DIE*	*DIE*	*DIE*	*DIE*
WEST	*DIE*	*DIE*	*DIE*	*DIE*

a. Puzzle 1: **GUESSING GAME**: Create a variable that stores a list of the names of the 7 dwarves. Ask them to name one (using all lowercase). Check if it's correct. Ask them for another name (using all lower case). Check that it's correct AND that it's different from the first one they names.Print out some response telling them if they got them right or wrong.

b. Puzzle 2: **DICE**: Roll a die and show them the value they rolled. Ask them if it's even or odd. If they get it right, tell them they're smart. If they get it wrong, tell them they got the **wrong** answer. A random roll can be simulated with a call to `random.randint(1,6)` which generates a uniform random integer in [1,6]. Make sure to import the random module (ie. `import random`) before you use `random.randint(1,6)`. DO NOT hardcode which dice rolls are even and which are odd, use python to figure it out.

c. Puzzle 3: **BACKWARDS**: Give them a word and ask them to type it in backwards. CASE MATTERS. If they get it correct, tell them good job. Otherwise show them the correct answer.

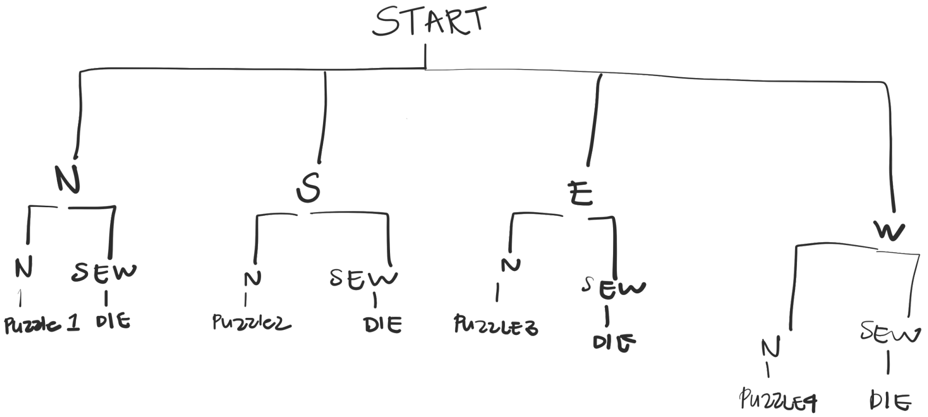
d. Puzzle 4: **PRIME**: ask them for a prime number. Check if they're correct, and let them know if they're right or wrong.

11. (5 pts) If they get it right, tell them they made it out alive and congratulate them. If they didn't get it right, Tell them that "Sorry, [their name here], you died" and add an additional death message.
12. **Extra Credit** (5 pts): Instead of asking the player questions in #5, ask them to enter a single word in all lowercase, and then ask them to enter that word (in all lower case) in pig latin. Check if they are correct. Instead of #6, print out a message telling them whether they're correct, or incorrect. If they're incorrect, print the correct version.

Pig latin is a fun, kids language where you insert the phoneme "ay" into English words. If a word starts with a vowel, you should add an "yay" at the end. For example, `apple` in pig latin would be appleyay, and open would become openyay. If a word starts with a consonant, then you take the first letter of the word, remove it and add it to the end of the word, then add an "ay". For example, bed would become edbay, sand would become andsay, car would become arcay. *HOWEVER, if the first two letters are in the list lst, then you take the first two letters of the word, remove them, and add it to the end of the word, then add an "ay".* For example, chelsea would become elseachay, plant would become antplay, brick would become ickbray. You can check if a string is an item in a list by using the in operator (e.g. the code "sh" in lst will return True because "sh" is one of the items in the list lst).

```
lst = ['sh', 'gl', 'ch', 'ph', 'tr', 'br', 'fr', 'bl', 'gr', 'st', 'sl', 'cl', 'pl', 'fl']
```

SEE BELOW DIAGRAM for more game logic help:


START
N
S
E
W
N
SEW
Puzzle 1
DIE
N
SEW
Puzzle2
DIE
N
SEW
PUZZLE3
DIE
N
SEW
PUZZLE4
DIE