

SECTION 7: MILESTONE PROJECT - 1, 1 hour 40 minutes, 9 parts

• 4/9 Validating User Input

○ -> using the input() function

- -> converting data from it from one type to another
- -> **a while loop which keeps on asking the user for input until the right datatype is entered**
- -> a while loop along with checking the datatype, asking the user for the correct input
- -> def user_choice():
 - choice = input("Please enter a number (0-10): ")
 - return int(choice)

○ -> user_choice()

- -> **she's put input into a() function**
- -> **calling the function now asks the user for an input**
- -> **then in the input function, they're checking against the expected inputs**
 - i.e in the function which asks the users for inputs
 - -> **she's adding in a while loop to make sure that the input is an integer (2 not two, e.g)**

• -> **there is a test to do this, which is isdigit, and she's looking through the documentation for it**

- it can have errors for negative numbers
- -> so we're assuming the user enters a positive input
 - **edge cases are things the users could enter which we would not normally expect**
- -> the user can enter something which isn't a digit, or which is outside of the range that we want
- -> some_value = '100'

○ -> **some_value shift tab <- and then we can see all of the different options there**

- the .isdigit method
- **-> then shift tab and it tells you the documentation**
- -> so she's found a method which tells the user if the thing is a digit or not
- -> we're checking if the choice is a digit
- -> **then she's testing the function with the new method in it -> testing the code etc**
- -> then while choice.isdigit() == False:
 - choice = input("Please enter a number (1-10): ")
 - **she's put the code asking for the user input in a while loop -> as long as the input isn't a digit then it will ask for the input again and again**
 - -> adding an error message so the user knows what happens
 - if choice.isdigit() == False:
 - (tab) print("Sorry that is not a digit!")
 - -> **so making different case scenarios -> of different things that the user could enter**

○ -> **checking across multiple different values**

- **checking the input from the user with a list of acceptable values**

- -> the in operator
- -> result in acceptable_values <- it returns False (a boolean)
- -> another one is result not in acceptable_values <- a boolean, to see if we have what we want or not
- -> in the code, she's initialised the values of the variables and then is testing the code
- -> range(0,10)
- -> then is populating them and testing the code
- -> while the choice.isdigit() == False or within_range==False:
 - -> then a digit check
 - -> and a range check
 - -> running different tests to check the user enters an input in the correct format
 - -> == is a boolean and = is a variable assignment
 - -> not being within the range
 - -> review the notebook and the logic will make sense
 - -> then she's testing the function with the edge cases again -> we are checking if the value is within range and the right datatype