0. Logic

In [9]:

```
test_strs = ['y', 'o']
comp_str = 'Hello World!'

(test_strs[1] or test_strs[0]) in comp_str
```

Out[9]:

```
True
```

1. Age Category:

In [2]:

```
#a

ages = int(input("What is your age?"))
infant = list(range(0,3))
child = list(range(3,13))
teenager = list(range(13,20))
adult = list(range(20,65))
senior = [65]



if ages in infant:
    print("You are an infant.")
elif ages in child:
    print("You are a child.")
elif ages in teenager:
    print("You are a teen.")
elif ages in adult:
    print("You are an adult.")
elif ages <0:
    print("Well that's kinda odd... Maybe try again.")
else :
    print("Jeez you're old.")
```

```
Well that's kinda odd... Maybe try again.
```

2. Number Classifier:

In [15]:

```
entered_number = input("Enter a whole number.")
entered_number = int(entered_number)
evenodd = entered_number % 2

#using modulus to determine even or odd
#using greater than or equal to as basis for pos/neg/zero

if entered_number >0:
    num = ("positive")
elif entered_number <0:
    num=("negative")
else:
    num=("zero")

if evenodd ==0:
    sign=("even")
elif evenodd==1:
    sign=("odd")


print("Your number,",entered_number, ",","is",num,"and",sign,".")
```

```
Your number, 430 , is positive and even .
```

3. Simple Calculator:

In [19]:

```
num1 = int(input("Enter the first number for your calculation"))
num2 = int(input("Enter the second number for your calculation"))
op = input("Enter the operation you would like to perform (*, +, -, /)")


if (num2 == 0) and (op =="/"):
        print("Oof, you can't divide by zero. You should know this...")

if op == "*":
    solution = num1 * num2
elif op == "+":
    solution = num1 + num2
elif op == "-":
    solution = num1 - num2
elif op == "/":
    solution = num1 / num2


print(solution)
```

```
1
```

4. Guessing Game:

In [44]:

```
#a-e
import random
answer = random.randint(1,10)
attempt = 0
max_attempt = 5

while attempt < 5: #limits number of guesses possible
    guess = int(input("Try guessing the number!"))
    attempt += 1 #adds 1 to attempt variable after each guess
    if guess>answer:
        print("Too high")
    elif guess<answer:
        print("Too low")
    else:
        print("Woohoo, you got it! The answer was", answer, "! :-)")
        break #stops the loop when you get it right
```

```
if attempt == max_attempt:
    print("Reached maximum number of attempts. The correct answer was", answer,
```

```
Too low

Too low

Woohoo, you got it! The answer was 9 ! :)
```

5. Counter Contrast:

In [6]:

```
n = int(input("Give me a positve integer."))

#a:

if n <=0:
    print("Nope. Positive integer, please.")

for i in range(1,n+1): #loops printing range until reaches n( + 1 to account fo
        print(i)
```

```
Nope. Positive integer, please.
```

In [7]:

```
#b
n = int(input("Give me a positve integer."))

if n <=0:
    print("Nope. Positive integer, please.")
else:
    i=1 #replaces the range function from the for loop

while i <= n:
    print(i)
    i=i+1 #keep adding to i (1) until you reach n
```

```
1
2
```

I much preferred a for loop *for* this HW problem. I think it was because it was easier for me to understand the problem in the context of a for loop, rather than a while loop where I had to understand what conditions I needed to establish.

6. Multiplication Table:

In [8]:

```
n = int(input("Give me a number"))
num2 = list(range(1,11))

#a
for i in num2:
    print(n, "x", i, "=", i * n)
```

```
4 x 1 = 4
4 x 2 = 8
4 x 3 = 12
4 x 4 = 16
4 x 5 = 20
4 x 6 = 24
4 x 7 = 28
4 x 8 = 32
4 x 9 = 36
4 x 10 = 40
```

In [49]:

```
#b
n = int(input("Give me a number"))
i = 1 #same as the counter contrast. gives us a condition to use for a while lo

while i <= 10:
    result = i * n
    print(n, "x", i, "=", i * n)
    i += 1 #add 1 to i each time (replaces the range function)
```

```
10 x 1 = 10
10 x 2 = 20
10 x 3 = 30
10 x 4 = 40
10 x 5 = 50
10 x 6 = 60
10 x 7 = 70
10 x 8 = 80
10 x 9 = 90
10 x 10 = 100
```

I still prefer a for loop. Again, I think its easier for me to understand programs like these in the context of a for loop.

For a program that requires a positive number and continues to prompt for a positive number until entered, a while loop would be easier. I can see what conditions aren't being met much more easily.