

Dylan Liesenfelt

Z23688417

Question 1

```
In [ ]: # A) Write code that creates a list of the squares of integers from 0 to 9 using a list comprehension.
x = [i*i for i in range(0,10)]
print('Answer A:', x)

# B) Write code that creates a list of the cubes of even integers from 0 to 20 using a list comprehension.
y = [i*i*i for i in range(0,21)]
print('Answer B:', y)

# Given the following code that converts a number from binary to decimal, create the same output using a list comprehension
num = '111101011011'
decimal = sum([int(d) * 2**i for i, d in enumerate(num[::-1])])
print(f'{num} in binary is equal to {decimal} in decimal')
```

Answer A: [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]

Answer B: [0, 1, 8, 27, 64, 125, 216, 343, 512, 729, 1000, 1331, 1728, 2197, 2744, 3375, 4096, 4913, 5832, 6859, 8000]

111101011011 in binary is equal to 3931 in decimal

Question 2

```
In [ ]: # Using a dictionary comprehension, count the occurrences of each vowel in the given text. Consider both uppercase and lowercase vowels.
text = 'Life is like riding a bicycle. To keep your balance, you must keep moving.'
vowels = 'AEIOU'
x = {vowel : text.upper().count(vowel) for vowel in vowels}
print('Answer:',x)
```

Answer: {'A': 3, 'E': 8, 'I': 7, 'O': 4, 'U': 3}

Question 3

In []: *# Convert the following conditional structure to a ternary operation. Print and test both versions.*

```
num = 7

# Given Code
if num > 0:
    result = "Positive"
elif num == 0:
    result = "Zero"
else:
    result = "Negative"
print('Original results: ', result)

# My Code (Ternary)
myResult = 'Positive' if num > 0 else 'Zero' if num == 0 else 'Negative'
print('Ternary conversion result:', myResult)
```

Original results: Positive

Ternary conversion result: Positive

Question 4

In []: *# Given the following function that returns the List of factors for an integer:*

```
n = 100
def factors(n):
    factors = []
    for i in range(1,n+1):
        #print(n,i,('Factor' if n%i == 0 else 'Not a factor'))
        factors.append(i) if n%i == 0 else None
    return factors
```

In []: *# A) Write a function isPrime(n) that returns True or False if the number n is prime, using the factors(n) function.*

```
def isPrime(n):
    return len(factors(n)) == 2
```

```
# B) Write a function primes(n) that returns a list of all the primes up to the number n. Use the isPrime(n) function
def primes(n):
    return [i for i in range(2, n+1) if isPrime(i)]

# C) Use the functions to return a report
print(f'The Number {n} is prime: {isPrime(n)}')
print(f'Factors: {factors(n)}')
print(f'Primes up to {n}: {primes(n)}')
print(f'Total number of primes up to {n}: {len(primes(n))}')
```

The Number 100 is prime: False

Factors: [1, 2, 4, 5, 10, 20, 25, 50, 100]

Primes up to 100: [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97]

Total number of primes up to 100: 25

Question 5

```
In [ ]: # Convert the following functions into Lambda expressions. Test both the original and Lambda versions and show your r
num, a, str1, str2 = 25, 'Dylan', 'Hello', ' World'

# A)
square = lambda num: num ** 2
print('Result A:', square(num))

# B)
concatenate = lambda str1, str2: str1 + str2
print('Result B:', concatenate(str1, str2))

# C)
is_even = lambda num: num % 2 == 0
print('Result C:', is_even(num))

# D)
reverse = lambda a : a[::-1]
print('Result D:', reverse(a))
```

Result A: 625

Result B: Hello World

Result C: False

Result D: nalyD

Question 6 (Bonus)

```
In [ ]: # Write a function to calculate the area of a rectangle given its width and height. Demonstrate how to call the funct
height, width = 25, 16

def areaRect(h,w):
    return h * w

print(f'The area of a rectangle of {height} by {width} is {areaRect(height, width)}')
```

The area of a rectangle of 25 by 16 is 400