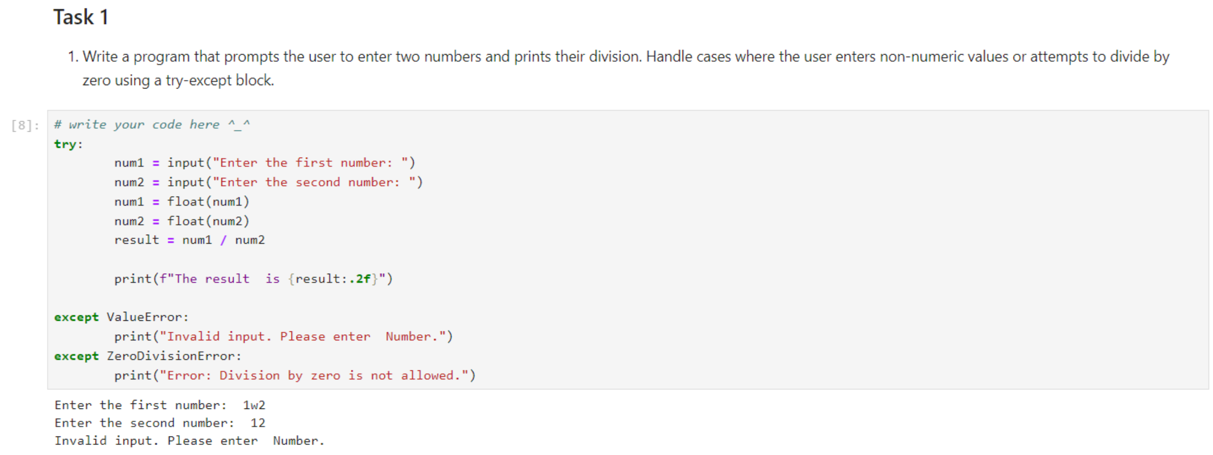
Lab_python_2.2

Task 1 :



Task 1

1. Write a program that prompts the user to enter two numbers and prints their division. Handle cases where the user enters non-numeric values or attempts to divide by zero using a try-except block.

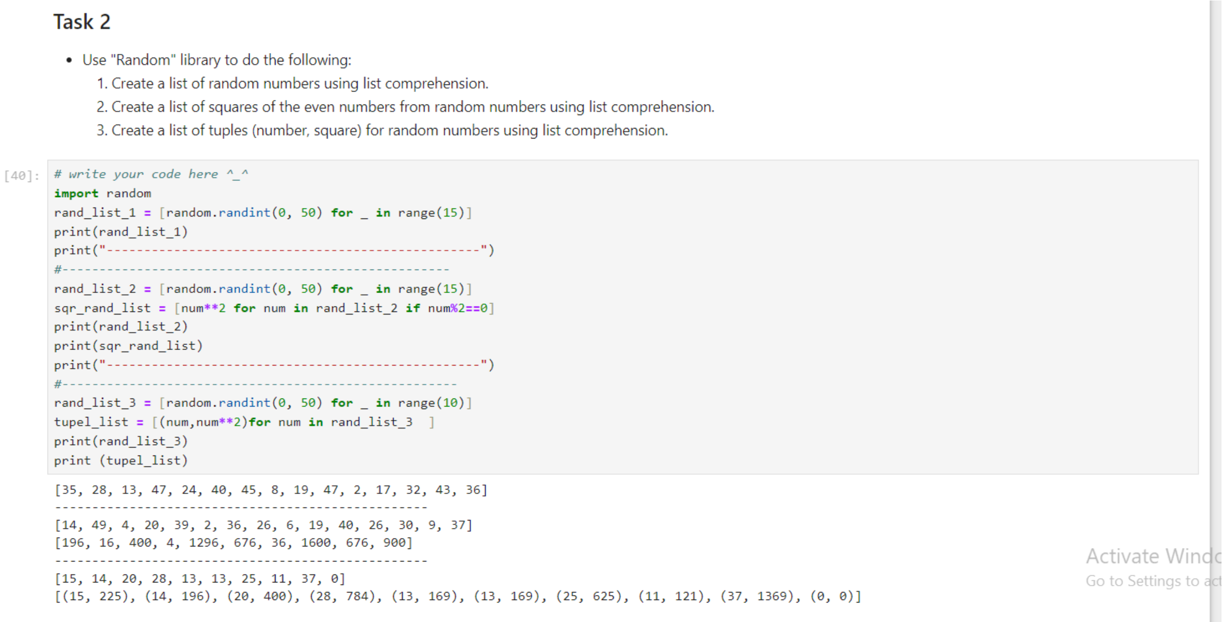
```
[8]: # write your code here ^_^
try:
        num1 = input("Enter the first number: ")
        num2 = input("Enter the second number: ")
        num1 = float(num1)
        num2 = float(num2)
        result = num1 / num2

        print(f"The result  is {result:.2f}")

except ValueError:
        print("Invalid input. Please enter  Number.")
except ZeroDivisionError:
        print("Error: Division by zero is not allowed.")
```

```
Enter the first number:  1w2
Enter the second number:  12
Invalid input. Please enter  Number.
```

Task 2 :



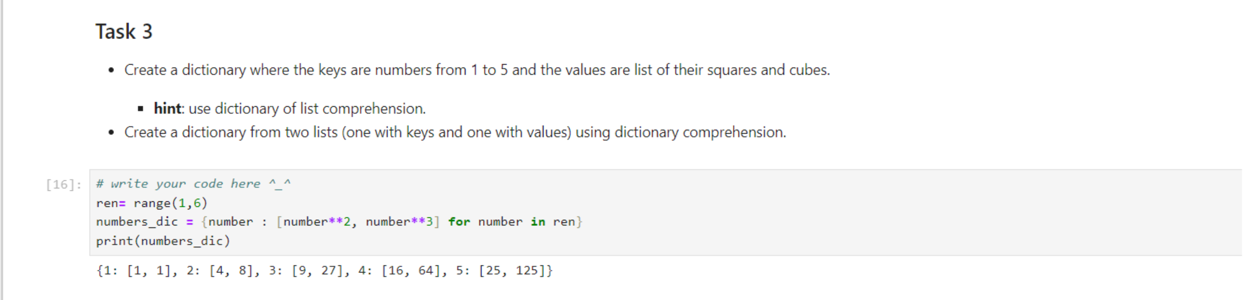
Task 2

- Use "Random" library to do the following:
  1. Create a list of random numbers using list comprehension.
  2. Create a list of squares of the even numbers from random numbers using list comprehension.
  3. Create a list of tuples (number, square) for random numbers using list comprehension.

```
[40]: # write your code here ^_^
import random
rand_list_1 = [random.randint(0, 50) for _ in range(15)]
print(rand_list_1)
print("--------------------------------------------------")
#-----------------------------------------------------
rand_list_2 = [random.randint(0, 50) for _ in range(15)]
sqr_rand_list = [num**2 for num in rand_list_2 if num%2==0]
print(rand_list_2)
print(sqr_rand_list)
print("--------------------------------------------------")
#------------------------------------------------------
rand_list_3 = [random.randint(0, 50) for _ in range(10)]
tupel_list = [(num,num**2)for num in rand_list_3  ]
print(rand_list_3)
print (tupel_list)
```

```
[35, 28, 13, 47, 24, 40, 45, 8, 19, 47, 2, 17, 32, 43, 36]
--------------------------------------------------
[14, 49, 4, 20, 39, 2, 36, 26, 6, 19, 40, 26, 30, 9, 37]
[196, 16, 400, 4, 1296, 676, 36, 1600, 676, 900]
--------------------------------------------------
[15, 14, 20, 28, 13, 13, 25, 11, 37, 0]
[(15, 225), (14, 196), (20, 400), (28, 784), (13, 169), (13, 169), (25, 625), (11, 121), (37, 1369), (0, 0)]
```

Task 3 :



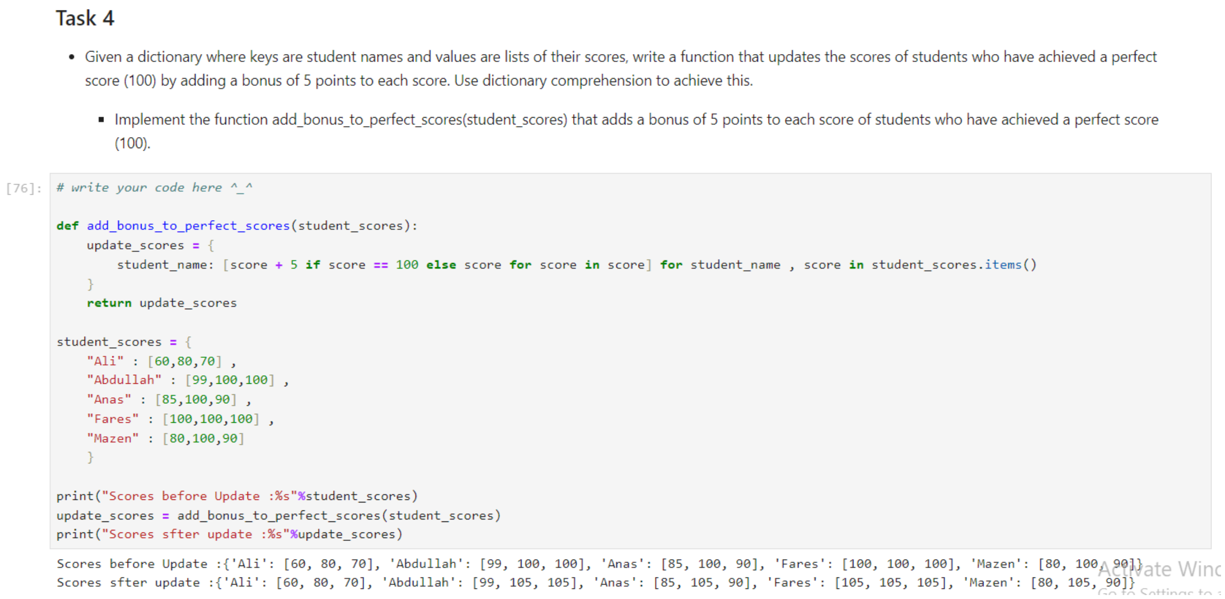
Task 3

- Create a dictionary where the keys are numbers from 1 to 5 and the values are list of their squares and cubes.
  - **hint**: use dictionary of list comprehension.
- Create a dictionary from two lists (one with keys and one with values) using dictionary comprehension.

```
[16]: # write your code here ^_^
ren= range(1,6)
numbers_dic = {number : [number**2, number**3] for number in ren}
print(numbers_dic)
```

```
{1: [1, 1], 2: [4, 8], 3: [9, 27], 4: [16, 64], 5: [25, 125]}
```

Task 4 :



Task 4

- Given a dictionary where keys are student names and values are lists of their scores, write a function that updates the scores of students who have achieved a perfect score (100) by adding a bonus of 5 points to each score. Use dictionary comprehension to achieve this.
  - Implement the function add_bonus_to_perfect_scores(student_scores) that adds a bonus of 5 points to each score of students who have achieved a perfect score (100).

[76]:

```
# write your code here ^_^

def add_bonus_to_perfect_scores(student_scores):
    update_scores = {
        student_name: [score + 5 if score == 100 else score for score in score] for student_name , score in student_scores.items()
    }
    return update_scores

student_scores = {
    "Ali" : [60,80,70] ,
    "Abdullah" : [99,100,100] ,
    "Anas" : [85,100,90] ,
    "Fares" : [100,100,100] ,
    "Mazen" : [80,100,90]
    }

print("Scores before Update :%s"%student_scores)
update_scores = add_bonus_to_perfect_scores(student_scores)
print("Scores sfter update :%s"%update_scores)
```

```
Scores before Update :{'Ali': [60, 80, 70], 'Abdullah': [99, 100, 100], 'Anas': [85, 100, 90], 'Fares': [100, 100, 100], 'Mazen': [80, 100, 90]}
Scores sfter update :{'Ali': [60, 80, 70], 'Abdullah': [99, 105, 105], 'Anas': [85, 105, 90], 'Fares': [105, 105, 105], 'Mazen': [80, 105, 90]}
```