



**TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS**

LAB 3

By:

Sinjal Dahal (081/BEL/080)

**DEPARTMENT OF COMPUTER ENGINEERING
LALITPUR, NEPAL**

1. Write a Python function named `greet_user` that takes a user's name and prints.

Source Code:

```
def greet_user():  
    user = input("Enter your name : ")  
    print(f"Hello , {user}")  
    greet_user()
```

Output:

```
Enter your name : Sinjal Dahal  
Hello , Sinjal Dahal
```

2. Hello, <name>! Welcome to Python. Call the function with a sample name.

Source Code:

```
def greet(name):  
    print(f"Hello, {name}")  
  
greet("Sinjal Dahal")
```

Output:

```
Hello, Sinjal Dahal
```

3. Create a function `power(base, exponent=2)` that returns the result of base raised to the power of exponent. Demonstrate it with and without the exponent argument.

Source Code:

```
import math  
  
def power(base , exponent=2):  
    print(math.pow(base , exponent))
```

```
power(2,8)
```

Output:

```
256.0
```

Source Code:

```
import math  
  
def power(base , exponent=2):  
    print(math.pow(base , exponent))
```

```
power(2)
```

Output:

```
4.0
```

4. Write a function `book_info(title, author, year)` that prints book details.Call the function using keyword arguments in different orders.

Source Code:

```
def book_info(title, author, year):  
    print(f"Title: {title}")  
    print(f"Author: {author}")  
    print(f"Year: {year}")  
  
book_info(title="1984", author="Doraemon", year=1949)  
  
print()  
  
book_info(author="J.K. Rowling", year=1997, title="Harry Potter")
```

```
print()

book_info(year=1813, title="P and P", author="Jane")
```

Output:

```
Title: 1984
Author: Doraemon
Year: 1949

Title: Harry Potter
Author: J.K. Rowling
Year: 1997

Title: P and P
Author: Jane
Year: 1813
```

5. Create a function `sum_numbers(*args)` that accepts any number of numeric arguments and returns their sum. Test it with 2, 3, and 5 numbers.

Source Code:

```
def sum_numbers(*args):
    print(sum(args))

sum_numbers(2,4)
sum_numbers(2,4,8)
sum_numbers(5,6,8,2,9)
```

Output:

```
6
14
30
```

6. Write a function `student_profile(kwargs)` that prints the key-value pairs passed (e.g., name, age, grade). Call it with at least three named arguments.**

Source Code:

```
def student_profile(**kwargs):  
    for key,value in kwargs.items():  
        print(f"{key} : {value}")  
  
student_profile(name="Sinjal",age=10,marks=56)  
student_profile(name="Ram",age=20,marks=96)  
student_profile(name="Shyam",age=30,marks=26)
```

Output:

```
name : Sinjal  
age : 10  
marks : 56  
name : Ram  
age : 20  
marks : 96  
name : Shyam  
age : 30  
marks : 26
```

7. Write a lambda function to compute the square of a number. Use it to compute the square of 5 and 12.

Source Code:

```
square = lambda x: x**2  
print(square(5))
```

```
print(square(12))
```

Output:

```
25
144
```

8. Given a list of numbers [1, 2, 3, 4, 5], use map() and a lambda function to return a new list with each number doubled.

Source Code:

```
numbers = [1, 2, 3, 4, 5]
double = list(map(lambda x: 2*x ,numbers))
print(numbers)
print(double)
```

Output:

```
[1, 2, 3, 4, 5]
[2, 4, 6, 8, 10]
```

9. Given a list [10, 15, 20, 25, 30], use filter() and a lambda function to extract numbers divisible by 10.

Source Code:

```
list1 = [10, 15, 20, 25, 30]
divisible_by_ten = list(filter(lambda x : x%10 == 0,list1))
print(divisible_by_ten)
```

Output:

```
[10, 20, 30]
```

10. Given a list of temperatures in Celsius [36.5, 37.0, 39.2, 35.6, 38.7], convert them to Fahrenheit using map(), Filter out those above 100°F using filter().

Source Code:

```
temperatures_in_celsius = [36.5, 37.0, 39.2, 35.6, 38.7]
temperatures_in_fahrenheit = list(map(lambda x: (x * 9/5) + 32, temperatures_in_celsius))
filtered_temperatures = list(filter(lambda x: x <= 100, temperatures_in_fahrenheit))
print("Temperatures in Fahrenheit:", temperatures_in_fahrenheit)
print("Temperatures ≤ 100°F:", filtered_temperatures)
```

Output:

```
Temperatures in Fahrenheit: [97.7, 98.6, 102.56, 96.08000000000001, 101.66]
Temperatures ≤ 100°F: [97.7, 98.6, 96.08000000000001]
```