Lab 2

# Topic 1 : While Loop

## Example Code :

# Python program to illustrate

# while loop

count = 0

while (count < 3):

count = count + 1

print("Pokemon")

#Output: Pokemon

# Topic : Single Statement While Block

## Example code:

# Python program to illustrate

# Single statement while block

count = 0

while (count == 0): print("Hello Geek")

#Output: Hello Geek

# Topic : for in loop

## Example Code :

# Python program to illustrate

# Iterating over a list

print("List Iteration")

l = ["geeks", "for", "geeks"]

for i in l:

print(i)

#Output : List Iteration

geeks

for

geeks

## Example Code 2 :

# Iterating over a tuple (immutable)

print("\nTuple Iteration")

t = ("geeks", "for", "geeks")

for i in t:

print(i)

#output:

Tuple Iteration

geeks

for

geeks

## Example Code 3 :

# Iterating over a String

print("\nString Iteration")

s = "Geeks"

for i in s :

print(i)

#output:

String Iteration

G

e

e

k

s

# Topic : iterating by index of sequence

## Example Code :

# Python program to illustrate

# Iterating by index

list = ["geeks", "for", "geeks"]

for index in range(len(list)):

print list[index]

#output :

geeks

for

geeks

Topic : loop Control Statements

## Continue Statement Code :

# Prints all letters except 'e' and 's'

for letter in 'geeksforgeeks':

if letter == 'e' or letter == 's':

continue

print('Current Letter :', letter)

#output:

Current Letter: g

Current Letter: k

Current Letter: f

Current Letter: o

Current Letter: r

Current Letter:g

Current Letter: k

## Break Statement Code:

# Prints letters until it encounters 'e' or 's', then breaks the loop

for letter in 'geeksforgeeks':

# break the loop as soon it sees 'e' or 's'

if letter == 'e' or letter == 's':

break

print('Current Letter :', letter)

#output:

Current Letter: g

Topic : Python Functions

## Example Code :

**Creating a function**

# Define a function named my\_function

def my\_function():

# Print a greeting message

print("Hello from a function")

**Calling a function:**

def my\_function():

print("Hello from a function")

my\_function()

#output : Hello from a function

Topic : Parameters

Example Code :

# Function definition with one parameter

def my\_function(fname):

print(fname + " Refsnes")

# Calling the function with different arguments

my\_function("Emil") # Output: Emil Refsnes

my\_function("Tobias") # Output: Tobias Refsnes

my\_function("Linus") # Output: Linus Refsnes

Topic : Default parameter value

## Example Code :

# Function definition with a default parameter value

def my\_function(country="Norway"):

print("I am from " + country)

# Calling the function with different arguments

my\_function("Sweden") # Output: I am from Sweden

my\_function("India") # Output: I am from India

my\_function() # Output: I am from Norway

my\_function("Brazil") # Output: I am from Brazil

Topic: Passing a List as a Parameter

## Example Code :

# Function definition that takes a list as a parameter

def my\_function(food):

for x in food:

print(x)

# Defining a list

fruits = ["apple", "banana", "cherry"]

# Calling the function with a list

my\_function(fruits)

# Output:

# apple

# banana

# cherry

Topic : Return Values

## Example Code :

# Function definition that returns a value

def my\_function(x):

return 5 \* x

# Calling the function and printing the result

print(my\_function(3)) # Output: 15

print(my\_function(5)) # Output: 25

print(my\_function(9)) # Output: 45

# 

# 

# Topic : Keyword Arguments

## Example Code :

# Function definition with keyword arguments

def my\_function(child3, child2, child1):

print("The youngest child is " + child3)

# Calling the function with keyword arguments

my\_function(child1="Emil", child2="Tobias", child3="Linus")

# Output: The youngest child is Linus

Topic : Python Classes/Objects

## Example Code :

# Class definition

class MyClass:

x = 5

# Creating an object of the class

p1 = MyClass()

# Printing the value of the property x

print(p1.x) # Output: 5

Topic : The\_init\_() Function

## Example Code :

# Class definition with \_\_init\_\_ function

class Person:

def \_\_init\_\_(self, name, age):

self.name = name

self.age = age

# Creating an object of the class

p1 = Person("John", 36)

# Printing the properties of the object

print(p1.name) # Output: John

print(p1.age) # Output: 36

Topic : Object Method

## Example Code :

# Class definition with a method

class Person:

def \_\_init\_\_(self, name, age):

self.name = name

self.age = age

def myfunc(self):

print("Hello my name is " + self.name)

# Creating an object of the class and calling the method

p1 = Person("John", 36)

p1.myfunc()

# Output: Hello my name is John