RegdNo. 322201297

Assignment No. 1

Course Code: ECAP776 **Registration Number:** 322201297

Instructions:

a. Attempt all questions given below in your own handwriting. Assignment in typed format will not be considered for evaluation.

b. The student has to complete the assignment in the allocated pages only. Any other page in case utilized shall not be considered.

Q1. What is Object Oriented Programming? Explain it with their concepts and merits.
[10 Marks] [CO2, L1]

Obeject Oriented Programming (OOP) is a programming paradigm that uses "objects" to design applications and computer programs. The fundamental idea behind OOP is to combine data and functions.

Concepts of OOP:

1. class: A blueprint for creating objects. It defines a datatype by bundling data and methods that work on the data into single unit.
2. Object: An instance of a class. It is a self-contained component that contains properties and methods needed to a certain type of data useful.
3. method: A function defined within a class. Methods define the behaviors of the objects created from the class.
4. Inheritance: The mechanism by which one class inherits the properties and methods of another class. This promotes code reusability.
5. Polymorphism: The ability to present same interface for different data types. It allows methods to do different things based on object.
6. Encapsulation: The bundling of the data and the methods.
7. Abstraction: The concept of hiding the complex implementations and showing only necessary features of an object.

Merits of OOP:

1. Modularity: The source code of an object can be written independently of the source code for other objects. This makes it easier to manage and modify.
2. Reusability: Objects and classes can be reused across programs.
3. Scalability: New objects can be created with small modifications to existing ones.
4. Maintainability: OOP makes it easier to manage and maintain code. Changes to a part of the system can be made with minimal impact on other parts.
5. Security: Encapsulation and data hiding provides a way to protect data from unintended interface and misuse.
6. Flexibility: Polymorphism and inheritance provide a flexible and dynamic way to handle different data types and operations.

Signature of the Student

Note:-
CO: is the Course Outcome as per your course syllabus.
L1-L6: Learning level objectives as per Revised Bloom Taxonomy (RBT).

Assignment No. 1

Course Code: ECAP776 **Registration Number:** 322201297

Instructions:

a. Attempt all questions given below in your own handwriting. Assignment in typed format will not be considered for evaluation.

b. The student has to complete the assignment in the allocated pages only. Any other page in case utilized shall not be considered.

Q2. Explain the different ways of indexing and slicing operations in NumPy package.

[10 Marks] [CO3, L2]

Indexing in NumPy

1. Basic Indexing: Similas to python lists, you can acces elements using sq. brackets [ ]
Example:
```
import numpy as np
arr = np.array([1,2,3,4,5])
print(arr[0]) # output 1
```

2. Multi-dimensional Indexing: For multi-dimensional arrays you can use a comma-separated list of endices.
Example:
```
arr = np.arr([1,2,3], [4,5,6])
print(arr[1,2]) # output 6.
```

3. Boolean Indexing: You can use boolean arrays to index another array.
Example:
```
arr = np.array([1,2,3,4,5])
bool-idx = arr > 3
print(arr[bool-idx]) # Output: [4,5]
```

Slicing in NumPy

1. Basic Slicing: Slicing Syntax is start: stop: step
Example:
```
arr = np.array([1,2,3,4,5])
print(arr[1:4]) # Output: [2,3,4]
```

2. Multi-dimensional Sliing: You can slice multidimensional arrays by specifying slices for each dimension.
Example:
```
arr = np.array([[1,2,3], [4,5,6], [7,8,9]])
print(arr[0:2, 1:3]) # output: [[2,3], [5,6]]
```

3. Ellipsis (...): The ellipsis can be used to represent multiple colons for slicing
Example:
```
arr = np.array([[[1,2],[3,4]],[[5,6],[7,8]]])
print(arr[..., 1]) # Output: [[2,4],[6,8]]
```

4. Advanced Slicing: You can indexing and slicing
Example:
```
arr = np.array([[1,2,3],[4,5,6],[7,8,9]])
print(arr[1:, 1:]) # output [[5,6], [8,9]]
```

Indexing: It allows you to access individual element or subsets of elements.
Sliing: It allows you to acces range of elements.

Signature of the Student Tariq

Note:-

CO: is the Course Outcome as per your course syllabus.

L1-L6: Learning level objectives as per Revised Bloom Taxonomy (RBT).