Experiment No.6
Serialization in python using Pickle
Date of Performance:
Date of Submission:

Experiment No. 6

Aim: Serialization in python using Pickle

Objective: To introduce basic concept of Pickle module

Theory:

- What is Serialization?
- Serialization is the process of converting a Python object into a byte stream that can be stored in a file or transmitted over a network.
- What is Pickle?
- Pickle is a Python module used for serializing and deserializing Python objects.
- Why Pickle?
- Pickle provides a convenient way to save Python objects to disk and load them back into memory later.
- How to use Pickle?
- The pickle module provides two main functions: dump() for serialization and load() for deserialization.

1) **pickle.dump(obj, file)**:
   - The **pickle.dump()** function is used to serialize a Python object **obj** and write it to a file specified by the file object **file**.
   - This function takes two parameters:
     - **obj**: The Python object to be serialized.
     - **file**: A file object opened in binary write mode ('wb') where the serialized data will be written.

2) **pickle.load(file)**:
   - The **pickle.load()** function is used to deserialize data from a file specified by the file object **file** and reconstruct the original Python object.
   - This function takes one parameter:
     - **file**: A file object opened in binary read mode ('rb') from which the serialized data will be read and deserialized.

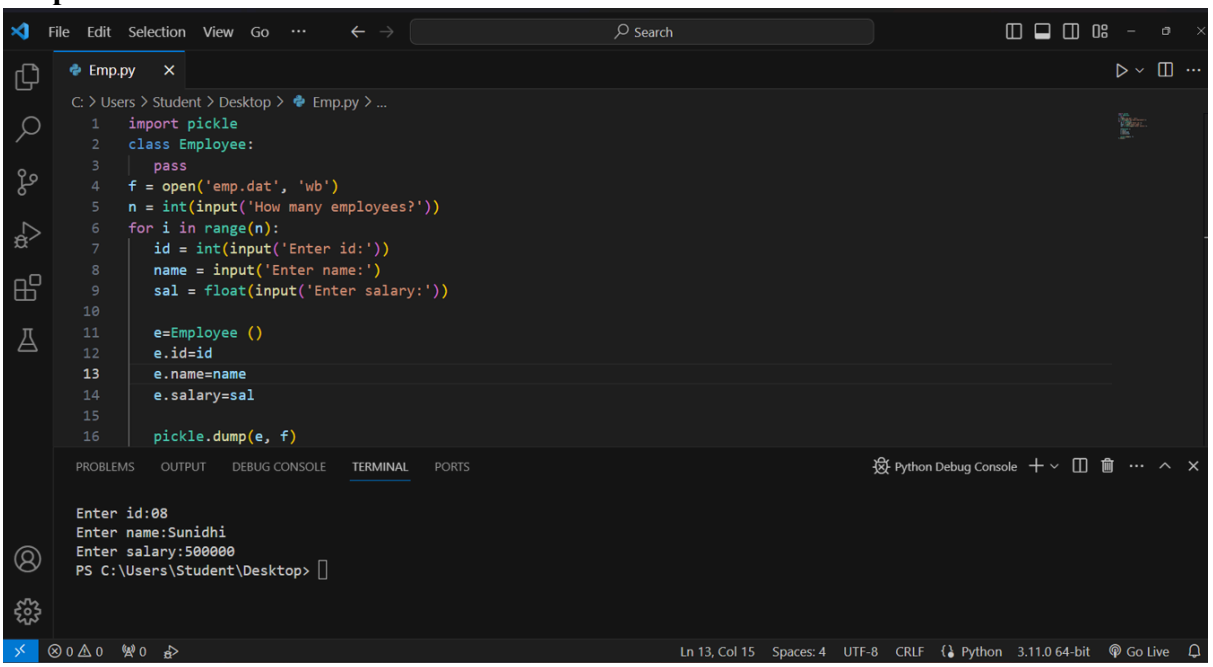
Code:-

```
import pickle
```

```
class Employee:
  pass
f = open('emp.dat', 'wb')
n = int(input('How many employees?'))
for i in range(n):
   id = int(input('Enter id:'))
   name = input('Enter name:')
   sal = float(input('Enter salary:'))

   e = Employee ()
   e.id =id
   e.name =name
   e.salary=sal
   pickle.dump(e, f)
f.close()
```

Output:-



Conclusion: Serialization in Python using Pickle has been demonstrated.