



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

Experiment No. 6

Aim: Serialization in python using Pickel

Objective: To introduce basic concept of Pickel 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 Emp:
```

```
    def __init__(self, id, name, sal):
```

```
        self.id = id
```

```
        self.name = name
```

```
        self.sal = sal
```

```
    def display(self):
```

```
        print("{:5d} {:20s} {:10.2f}".format(self.id, self.name, self.sal))
```

```
# Writing employee data to file
```

```
with open("emp.dat", "wb") as f:
```

```
    n = int(input("How many employees: "))
```

```
    for _ in range(n):
```

```
        id = int(input("Enter id: "))
```

```
        name = input("Enter name: ")
```

```
        sal = int(input("Enter sal: "))
```

```
        e = Emp(id, name, sal)
```

```
        pickle.dump(e, f)
```

```
# Reading employee data from file
```

```
with open("emp.dat", "rb") as f:
```

```
    emp_objects = []
```

```
    try:
```

```
        while True:
```

```
            emp_obj = pickle.load(f)
```

```
            emp_objects.append(emp_obj)
```

```
    except EOFError:
```

```
        pass
```



```
# Displaying employee data
for emp_obj in emp_objects:
    print("ID:", emp_obj.id)
    print("Name:", emp_obj.name)
    print("Salary:", emp_obj.sal)
    print()
```

Output:-

```
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.

D:\>python Serial1.py
How many employees: 1
Enter id: 98
Enter name: Suvidhi Pareek
Enter sal: 10000000
ID: 98
Name: Suvidhi Pareek
Salary: 10000000

D:\>
```

Conclusion: Serialization in Python using Pickle has been demonstrated.

Python pickle module is used for serializing and de-serializing a Python object structure. Any object in Python can be pickled so that it can be saved on disk.