ASSIGNMENT-1

```
class Department:
    dept_count = 0
    def _init_(self, id, name, loc, hod):
        self.id = id
        self.name = name
        self.loc = loc
        self.hod = hod
        Department.dept_count += 1
    def display_dept_info(self):
        print("\nDepartment Information:")
        print("------------------------")
        print(f"ID: {self.id}")
        print(f"Name: {self.name}")
        print(f"Location: {self.loc}")
        print(f"Head of Dept: {self.hod}")
    @classmethod
    def get_total_depts(cls):
        return cls.dept_count
    @staticmethod
    def search_by_id(departments, search_id):
        for dept in departments:
            if dept.id == search_id:
                print("\n--- Department Found by ID ---")
                dept.display_dept_info()
                return
        print("Department with given ID not found.")
    @staticmethod
    def search_by_name(departments, search_name):
```

```
        for dept in departments:
            if dept.name.lower() == search_name.lower():
                print("\n--- Department Found by Name ---")
                dept.display_dept_info()
                return
        print("Department with given Name not found.")
num = int(input("Enter the number of departments: "))
departments = []
for i in range(num):
    print(f"\nEnter details for Department {i+1}:")
    id = int(input("Enter Dept ID: "))
    name = input("Enter Dept Name: ")
    loc = input("Enter Dept Location: ")
    hod = input("Enter Head of Department: ")
    dept = Department(id, name, loc, hod)
    departments.append(dept)
for dept in departments:
    dept.display_dept_info()
print(f"\nTotal Departments: {Department.get_total_depts()}")
search_id = int(input("\nEnter Department ID to search: "))
Department.search_by_id(departments, search_id)
search_name = input("\nEnter Department Name to search: ")
Department.search_by_name(departments, search_name)
```