Unit-2: Chellenge 2.1

```
# Python program to create Bankaccount class

# with both a deposit() and a withdraw() function


class Bank_Account:


    def __init__(self):


        self.balance=0


        print("Hello!!! Welcome to the Deposit & Withdrawal Machine")



    def deposit(self):


        amount=float(input("Enter amount to be Deposited: "))


        self.balance += amount


        print("\n Amount Deposited:",amount)



    def withdraw(self):
```

```
        amount = float(input("Enter amount to be Withdrawn: "))

        if self.balance>=amount:

            self.balance-=amount

            print("\n You Withdrew:", amount)

        else:

            print("\n Insufficient balance  ")



    def display(self):

        print("\n Net Available Balance=",self.balance)

# Driver code



# creating an object of class

s = Bank_Account()
```

```
# Calling functions with that class object

s.deposit()

s.withdraw()

s.display()
```

Output:

```
Hello !!! Welcome to Deposit&Withdrawal Machine

Enter amount to be deposited:

 Amount Deposited: 1000.0

Enter amount to be withdrawn:

 You Withdrew: 500.0


 Net Available Balance = 500.0
```

Unit-2:Chellenge 2.2

```
# A Python program to demonstrate inheritance

# Base or Super class. Note object in bracket.

# (Generally, object is made ancestor of all classes)

# In Python 3.x "class Person" is

# equivalent to "class Person(object)"

class Person(object):

    # Constructor

    def __init__(self, name):

        self.name = name
```

```
    # To get name
    def getName(self):
        return self.name
    # To check if this person is employee
    def isEmployee(self):
        return False
# Inherited or Sub class (Note Person in bracket)
class Employee(Person):
    # Here we return true
    def isEmployee(self):
        return True
# Driver code
emp = Person("Geek1")  # An Object of Person
print(emp.getName(), emp.isEmployee())
emp = Employee("Geek2") # An Object of Employee
print(emp.getName(), emp.isEmployee())
```

Output:

```
('Geek1', False)
('Geek2', True)
```