

TEST-1

BSC.IT (3rd SEMESTER)

Sub: Object Oriented Development Using Java

Sub Code: CT038-3-2

NAME: SHIVAM RANABHAT

Intake Code: NPI2F1909IT

Student ID: NPI000047

Question 1:

ANS: C

Question 2:

ANS: D

Question 3:

ANS: B

Question 4:

ANS: B

QUESTION 5:

ANS: C

6.

a. Creating two appropriate class for account and customer using constructor

```
class Account{
    int id;
    double balance;
    Account(int id, double balance)
    {
        this.id = id;
        this.balance = balance;
    }
}

class Customer{
    Account myAccount;
    String name;
    Customer(String name, Account ma)
    {
        this.name = name;
        this.myAccount = ma;
    }
}
```

b.

```
public class Testing{
    public static void main(String[] args){
```

```
String name = "Joseph";  
Account acc = new Account(12345, 500.0);  
Customer c;  
c = new Customer (name, acc);  
}  
}
```

7. To use encapsulation we should make private instance variable in class Account so that they cannot be accessed by directly from outside the class TestAccount and we should use getter and setter method. Using modularity we can modify the balance including new balance, deposit and withdraw.

```
class Account{  
    private double balance;  
    Account(double x){  
        this.balance = x;  
    }  
    //Using Setter method  
    public void setBalance(double x){  
        balance = x;  
    }  
    //Using Getter method  
    public double getBalance(){  
        return balance;  
    }  
}
```

```
}
```

```
}
```

```
public class TestAccount{
```

```
    public static void main(String[] args){
```

```
        Account x = new Account(500.00);
```

```
        x.setBalance(550.00); // modifying the balance
```

```
        x.setBalance(x.getBalance() - 100.00); //withdraw 100.00 from the account
```

```
        x.setBalance(x.getBalance() + 250.00); //deposit 250.00 form the account
```

```
        System.out.println(x.getBalance()); //access the final balance after withdraw a  
nd deposit
```

```
    }
```

```
}
```