

SCD-Lab

Lab#4

Name: Anas-Altaf

Roll.No: 22F-3639

Java Codes:

Task -1:

```
package Task_1;
class NotificationService {
    public void sendNotification(String message) {
        System.out.println(message);
    }
}
class TransactionService {
    private NotificationService notificationService = new
NotificationService();
    public TransactionService() {
    }
    public void processTransaction(double amount) {
        System.out.println("Transaction Processed of Amount: "
+ amount);
        notificationService.sendNotification("Notification :
Transaction Succesful, Amount : Rs: " + amount);
    }
}
class AccountService {
    private TransactionService transactionService;
    private NotificationService notificationService;
    public AccountService(TransactionService ts) {
        this.transactionService = ts;
    }
    public void processAccountTransaction(double amount) {
        System.out.println("Account Service Running...");
        transactionService.processTransaction(amount);
    }
}
```

```

    }
    public void setNotificationService(NotificationService ns)
{
    notificationService = ns;
    System.out.println("NotificationService is Injected
using Setter");
}
    public void useNotificationService(String msg) {
        notificationService.sendNotification(msg);
    }
}
public class DriverClass {
    public static void main(String[] args) {
        TransactionService ts = new TransactionService();
        NotificationService ns = new NotificationService();
        AccountService accService = new AccountService(ts);
        accService.processAccountTransaction(1200);
        // Setter Injection
        accService.setNotificationService(ns);
        accService.useNotificationService("Notification
Service is Used..");
    }
}

```

Output:

```

Account Service Running...
Transaction Processed of Amount: 1200.0
Notification : Transaction Succesful, Amount : Rs: 1200.0
NotificationService is Injected using Setter
Notification Service is Used..

```

Task-2:

```

package Task_2;

import java.util.List;

abstract class Item {
    protected String name;
}

```

```

    Item(String name) {
        this.name = name;
    }

    String getName() {
        return name;
    }

    public abstract void getDetails();
}

class NonPerishableItem extends Item {

    String expDate;

    NonPerishableItem(String name) {
        super(name);
    }

    @Override
    public void getDetails() {
        System.out.println("Details : " + getName());
    }
}

class PerishableItem extends Item {

    String expDate;

    PerishableItem(String name, String expDate) {
        super(name);
        this.expDate = expDate;
    }

    @Override
    public void getDetails() {
        System.out.println("Details : " + getName() + "\nExpiry Date: " +
expDate);
    }
}

class Inventory<T> extends Item {
    List<T> items;

    public void addItem(T item) {
        items.add(item);
    }
}

```

```

    }

    public void removeItem(T item) {
        items.remove(item);
    }

    public void displayItems() {
        System.out.println("Displaying Items:");
        for (T item : items) {
            item.getDetails();
        }
    }

    public void filterByType(Class<?> itemType) {
        System.out.println("Filtered By Item : " + itemType.getSimpleName());
        for (T item : items) {
            //if(item == items.getClass())
            item.getDetails();
        }
    }
}

public class InventoryDriver {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

    }

}

```

Output: