**DAY-5 PRACTICE EXERCISE**

**BASICS OF JAVA**

**Problem-1**

BillingService Class

**package** org.software.com;

**public** **class** BillingService {

**private** **static** BillingService *billing*;

**private** BillingService(){

}

**public** **static** **synchronized** BillingService getInstance() {

**if**(*billing*==**null**) {

*billing*=**new** BillingService();

}

**return** *billing*;

}

**public** **void** processPayment(String payment) {

System.***out***.println("Payment Handled ");

}

**public** **void** generateInvoice(String order) {

System.***out***.println("Order Details are Displayed ");

}

}

TestBillingService Class

**package** org.software.com;

**public** **class** TestBillingService {

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

BillingService b1 = BillingService.*getInstance*();

BillingService b2 = BillingService.*getInstance*();

b1.processPayment("");

b2.generateInvoice("");

}

}

**Problem-2**

Vehicle Interface

**package** org.software.com;

**public** **interface** Vehicle {

**void** start();

**void** accelerate();

**void** brake();

}

Car Class

**package** org.software.com;

**class** Car **implements** Vehicle {

@Override

**public** **void** start() {

System.***out***.println("Car Started");

}

@Override

**public** **void** accelerate() {

System.***out***.println("Car Accelerated");

}

@Override

**public** **void** brake() {

System.***out***.println("Car Stopped");

}

}

MotorCycle Class

**package** org.software.com;

**class** MotorCycle **implements** Vehicle {

@Override

**public** **void** start() {

System.***out***.println("MotorCycle Started");

}

@Override

**public** **void** accelerate() {

System.***out***.println("Motorcycle Accelerated");

}

@Override

**public** **void** brake() {

System.***out***.println("MoterCycle Stopped");

}

}

Truck Class

**package** org.software.com;

**class** Truck **implements** Vehicle {

@Override

**public** **void** start() {

System.***out***.println("Truck Started");

}

@Override

**public** **void** accelerate() {

System.***out***.println("Truck Accelerated");

}

@Override

**public** **void** brake() {

System.***out***.println("Truck Stopped");

}

}

VehicleFactory Class

**package** org.software.com;

**public** **class** VehicleFactory {

**public** **static** Vehicle createVehicle(String type) {

**if**(type.equalsIgnoreCase("Car"))

**return** **new** Car();

**else** **if**(type.equalsIgnoreCase("Truck"))

**return** **new** Truck();

**else** **if**(type.equalsIgnoreCase("MotorCycle"))

**return** **new** MotorCycle();

**else**

**return null;**

}

}

Main Class

**package** org.software.com;

**public** **class** Main {

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

Vehicle car=VehicleFactory.*createVehicle*("Car");

car.start();

car.accelerate();

car.brake();

System.***out***.println("------------------------------------");

Vehicle motorCycle=VehicleFactory.*createVehicle*("MotorCycle");

motorCycle.start();

motorCycle.accelerate();

motorCycle.brake();

System.***out***.println("------------------------------------");

Vehicle truck=VehicleFactory.*createVehicle*("Truck");

truck.start();

truck.accelerate();

truck.brake();

}

}

**Problem-3**

Shape Class

**package** org.software.com;

**interface** Shape {

**abstract** **void** draw();

}

Circle Class

**package** org.software.com;

**public** **class** Circle **implements** Shape {

@Override

**public** **void** draw() {

System.***out***.println("Circle Shape Designed");

}

}

Rectangle Class

**package** org.software.com;

**public** **class** Rectangle **implements** Shape {

@Override

**public** **void** draw() {

System.***out***.println("Rectangle Shape Designed");

}

}

Square Class

**package** org.software.com;

**public** **class** Square **implements** Shape{

@Override

**public** **void** draw() {

System.***out***.println("Square Shape Designed");

}

}

AbstractFactory Class

**package** org.software.com;

**abstract** **class** AbstractFactory {

**abstract** Shape getShape(String shapeType);

}

ShapeFactory Class

**package** org.software.com;

**public** **class** ShapeFactory **extends** AbstractFactory{

@Override

**public** Shape getShape(String shapeType) {

**if**(shapeType.equalsIgnoreCase("Circle"))

**return** **new** Circle();

**else** **if**(shapeType.equalsIgnoreCase("Rectangle"))

**return** **new** Rectangle();

**else** **if**(shapeType.equalsIgnoreCase("Square"))

**return** **new** Square();

**else**

**return** **null**;

}

}

FactoryProducer Class

**package** org.software.com;

**public** **class** FactoryProducer{

**public** **static** AbstractFactory getShapeFactory() {

**return** **new** ShapeFactory();

}

}

AbstractFactoryDesighPattern Class

**package** org.software.com;

**public** **class** AbstractFactoryDesighPattern {

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

AbstractFactory shapeFactory = FactoryProducer.*getShapeFactory*();

Shape circle = shapeFactory.getShape("CIRCLE");

circle.draw();

Shape rectangle = shapeFactory.getShape("RECTANGLE");

rectangle.draw();

Shape square = shapeFactory.getShape("SQUARE");

square.draw();

}

}

**Problem-4**

Employee Class

**package** org.software.com;

**public** **class** Employee {

**private** String firstName;

**private** String lastName;

**private** String dateOfbirth;

**private** **int** empId;

**private** String joiningDate;

**private** **double** salary;

**public** Employee(String firstName, String lastName, String dateOfbirth, **int** empId, String joiningDate,**double** salary) {

**super**();

**this**.firstName = firstName;

**this**.lastName = lastName;

**this**.dateOfbirth = dateOfbirth;

**this**.empId = empId;

**this**.joiningDate = joiningDate;

**this**.salary = salary;

}

**public** String getFirstName() {

**return** firstName;

}

**public** String getLastName() {

**return** lastName;

}

**public** **int** getEmpId() {

**return** empId;

}

**public** String getDateOfbirth() {

**return** dateOfbirth;

}

**public** String getJoiningDate() {

**return** joiningDate;

}

**public** **double** getSalary() {

**return** salary;

}

}

FindEmployee Class

**package** org.software.com;

**public** **class** FindEmployee {

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

Employee emp = **new** Employee("Sreekanth Reddy", "Vaddemani", "10-06-2001", 1234, "10-06-2024", 30000);

System.***out***.println("Employee Name : " + emp.getFirstName() + " " + emp.getLastName());

System.***out***.println("Employee Date Of Birth : " + emp.getDateOfbirth());

System.***out***.println("Employee Id : " + emp.getEmpId());

System.***out***.println("Employee Date Of Joining : " + emp.getJoiningDate());

System.***out***.println("Employee Salary : " + emp.getSalary());

}

}