**Data structures and Algorithms**

**Exercise 2: E-commerce Platform Search Function**

1. package collect;

public interface **Find** {

void show();

}

1. **public class** SoftToy **implements** Find {

@Override

public void show() {

System.*out*.println("This is a soft toy.");

}

}

1. **public** **class** HardToy **implements** Find {

@Override

**public** **void** show() {

System.***out***.println("This is a hard toy.");

}

}

1. **public** **class** Factory {

**public** Find getToy(String type) {

**if** (type == **null** || type.isEmpty()) **return** **null**;

**switch** (type.toUpperCase()) {

**case** "SOFT":

**return** **new** SoftToy();

**case** "HARD":

**return** **new** HardToy();

**default**:

**return** **null**;

}

}

}

1. **public** **class** Main {

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

Factory f = **new** Factory();

Find soft = f.getToy("SOFT");

**if** (soft != **null**) soft.show();

Find hard = f.getToy("HARD");

**if** (hard != **null**) hard.show();

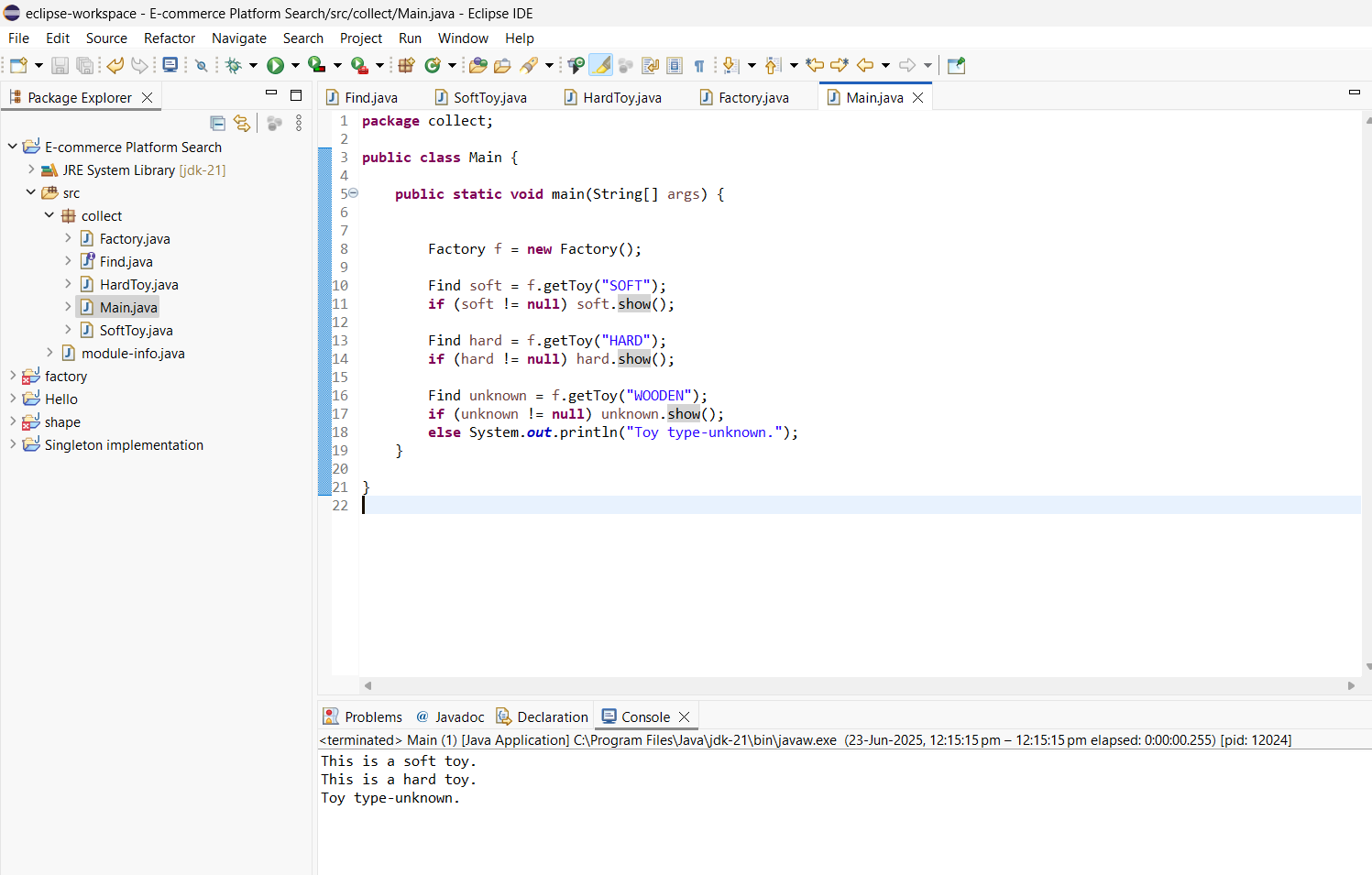
Find unknown = f.getToy("WOODEN");

**if** (unknown != **null**) unknown.show();

**else** System.***out***.println("Toy type-unknown.");

}

}



**Exercise 7: Financial Forecasting:**

* package collect;

import java.util.\*;

public interface Forecast {

void predict(double baseAmount);

}

* **public** **class** Construction **implements** Forecast{

**public** **void** predict(**double** baseAmt) {

**double** estimatedCost = baseAmt \* 1.25;

System.***out***.println("Estimated construction cost: ₹" + estimatedCost);

}

}

* **public** **class** Factory {

**public** Forecast getForecast(String type) {

**if** (type == **null** || type.isEmpty()) **return** **null**;

**switch** (type.toUpperCase()) {

**case** "CONSTRUCTION":

**return** **new** Construction();

**default**:

**return** **null**;

}

}

}

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

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

Scanner sc = **new** Scanner(System.***in***);

Factory f = **new** Factory();

System.***out***.println("Enter forecast type:");

String type = sc.nextLine();

System.***out***.println("Enter base amount in ₹:");

**double** amount = sc.nextDouble();

Forecast forecast = f.getForecast(type);

**if** (forecast != **null**) {

forecast.predict(amount);

} **else** {

System.***out***.println("Invalid forecast type.");

}

sc.close();

}

}

