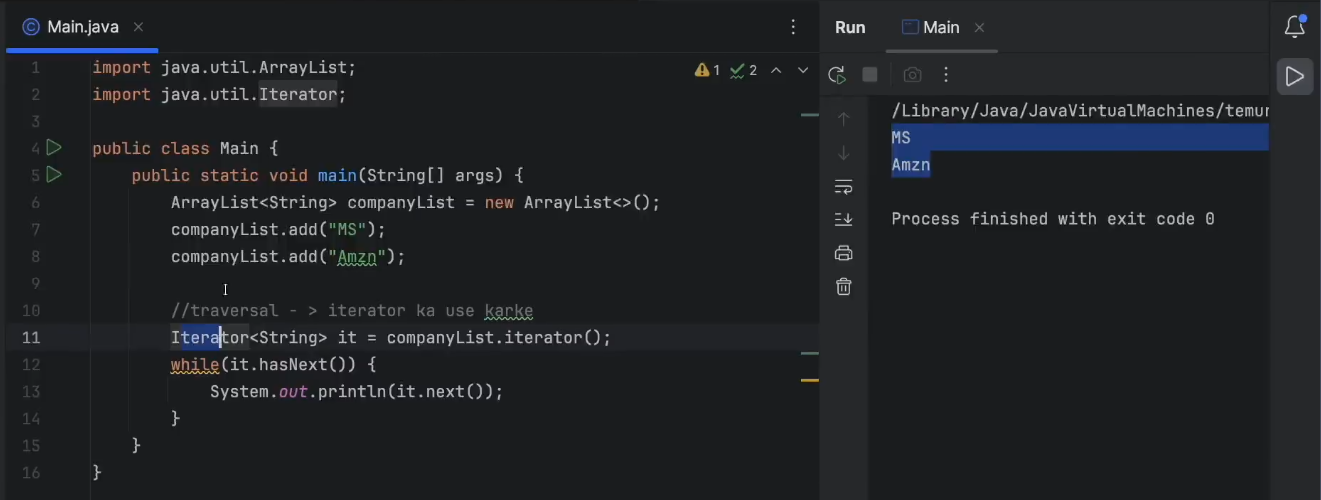
3.0 Let's learn Iterator && Visitor Pattern



Double dispatch

public class Main {

    public static void main(String[] args) {

        Scorpio scorpio = new Scorpio();

        Scorpio advScorpio = new AdvancedScorpio();

        StandardBreak standardBreak = new StandardBreak();

        StandardBreak advBreak = new AdvancedBreak();

//        scorpio.applyBreak(standardBreak);

//        advScorpio.applyBreak(standardBreak);

        scorpio.applyBreak(advBreak);

        //yaha fatega

        advScorpio.applyBreak(advBreak);

    }

}

class StandardBreak {

    public String playSound() {

        return " normal break ki sound";

    }

}

class AdvancedBreak extends StandardBreak {

    @Override

    public String playSound() {

        return "adv break ki sound";

    }

}

class AdvancedScorpio extends Scorpio{

    @Override

    public String whatsMyName() {

        return "Advanced Scorpio";

    }

}

class Scorpio {

    public String whatsMyName() {

        return "Scorpio";

    }

    public void applyBreak(StandardBreak normalBreak) {

        System.out.println(whatsMyName() + " apply normal Break " + normalBreak.playSound());

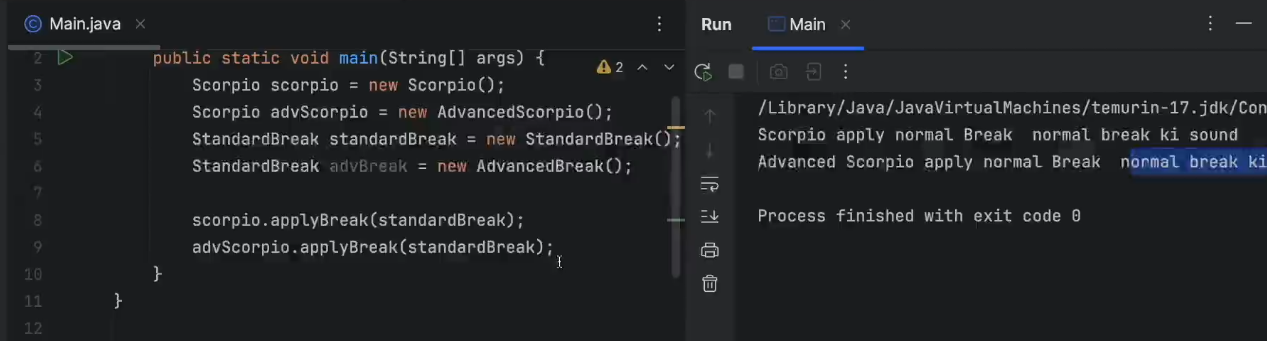
    }

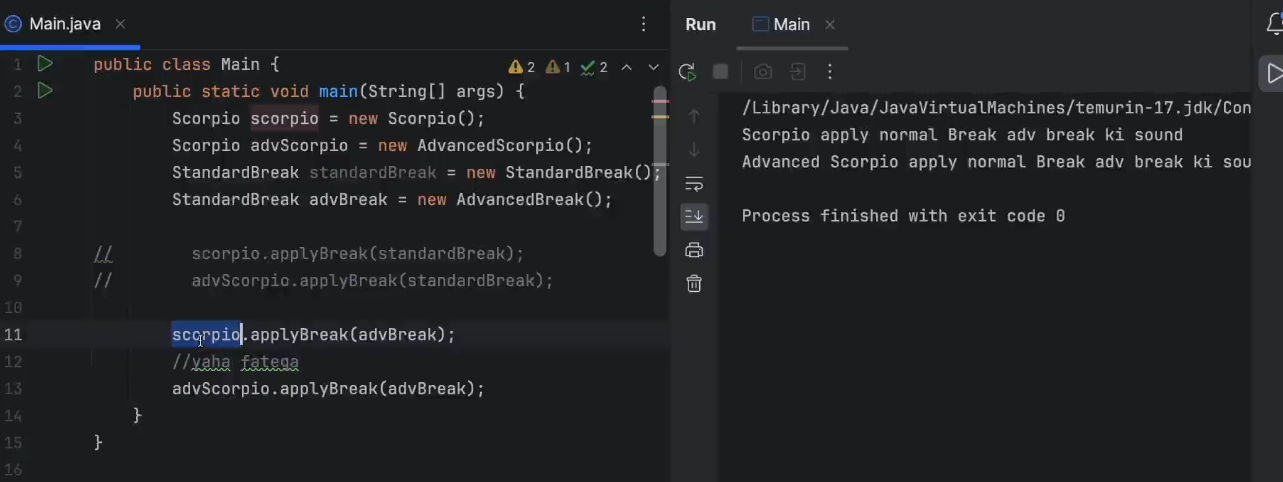
    public void applyBreak(AdvancedBreak advBreak) {

        System.out.println(whatsMyName() + " apply adv break " + advBreak.playSound());

    }

}





Visitor Pattern for solve double dispatch

