// Interface que representa uma Galinha

interface *Galinha* {

    cacarejar(): *void*;

  }

  // Classe que representa uma implementação concreta de Galinha

  class GalinhaComum implements *Galinha* {

    cacarejar(): *void* {

      console.log("Cocoricó!");

    }

  }

  // Interface que representa um Pato

  interface *Pato* {

    grasnar(): *void*;

  }

  // Classe que representa uma implementação concreta de Pato

  class PatoComum implements *Pato* {

    grasnar(): *void* {

      console.log("Quack!");

    }

  }

  // Adaptador que permite que um objeto Pato seja usado como uma Galinha

  class AdaptadorPato implements *Galinha* {

    private pato: *Pato*;

    constructor(*pato*: *Pato*) {

*this*.pato = *pato*;

    }

    cacarejar(): *void* {

      // Adapta o método cacarejar para chamar o método grasnar do Pato

*this*.pato.grasnar();

    }

  }

  // Exemplo de uso

  const galinhaComum: *Galinha* = **new** GalinhaComum();

  const patoComum: *Pato* = **new** PatoComum();

  console.log("Comportamento normal da Galinha:");

  galinhaComum.cacarejar();

  console.log("\nComportamento normal do Pato:");

  patoComum.grasnar();

  // Usando o Adaptador para permitir que o Pato seja usado como Galinha

  const adaptadorPato: *Galinha* = **new** AdaptadorPato(patoComum);

  console.log("\nUsando o Adaptador para o Pato como se fosse uma Galinha:");

  adaptadorPato.cacarejar();