```
public class Carre implements Forme {

    public boolean intersecte(Forme f) {
        if (f instanceof Cercle) {
            return intersecte((Cercle) f);
        } else if (f instanceof Carre) {
            return intersecte((Carre) f);
        }
        throw new IllegalArgumentException("intersection " + this.getClass().getName
() + " - " + f.getClass().getName() + " impossible");
    }

    public boolean intersecte(Cercle c) {
        System.out.println("intersection carre-cercle");
        return true; // au hasard !!!
    }

    public boolean intersecte(Carre c) {
        System.out.println("intersection carre-carre");
        return true; // au hasard !!!
    }

    public void dessine() {
        System.out.println("Je suis un carre");
    }
}
```



```
public class Cercle implements Forme {

    public boolean intersecte(Forme f) {
        if (f instanceof Cercle) {
            return intersecte((Cercle) f);
        } else if (f instanceof Carre) {
            return f.intersecte(this);
        }
        throw new IllegalArgumentException("intersection " + this.getClass().getName
() + " - " + f.getClass().getName() + " impossible");
    }

    public boolean intersecte(Cercle c) {
        System.out.println("intersection cercle-cercle");
        return true; // au hasard !!!
    }

    public void dessine() {
        System.out.println("Je suis un cercle");
    }
}
```

```
public interface Forme {
    public boolean intersecte(Forme f);
    public void dessine();
}
```



```
import java.lang.reflect.*;

public class Test {
    public static void main(String [] args)
        throws Exception
    {
        Forme ce = new Cercle();
        Forme ca = new Carre();

        ce.intersecte(ce);
        ce.intersecte(ca);
        ca.intersecte(ce);
        ca.intersecte(ca);
    }
}
```