**Prototype Pattern Example: Shape**

**Class: Shape**

**package** shape;

**public** **abstract** **class** Shape **implements** Cloneable {

**private** String id;

**protected** String type;

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

**public** String getType(){

**return** type;

}

**public** String getId() {

**return** id;

}

**public** **void** setId(String id) {

**this**.id = id;

}

**public** Object clone() {

Object clone = **null**;

**try** {

clone = **super**.clone();

} **catch** (CloneNotSupportedException e) {

e.printStackTrace();

}

**return** clone;

}

}

**Class: Rectangle**

**package** shape;

**public** **class** Rectangle **extends** Shape {

**public** Rectangle(){

type = "Rectangle";

}

@Override

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

System.***out***.println("Inside Rectangle::draw() method.");

}

}

**Interface: Square**

**package** shape;

**public** **class** Square **extends** Shape {

**public** Square(){

type = "Square";

}

@Override

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

System.***out***.println("Inside Square::draw() method.");

}

}

**Class: Circle**

**package** shape;

**public** **class** Circle **extends** Shape {

**public** Circle(){

type = "Circle";

}

@Override

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

System.***out***.println("Inside Circle::draw() method.");

}

}

**Class: ShapeCache**

**package** shape;

**import** java.util.Hashtable;

**public** **class** ShapeCache {

**private** **static** Hashtable<String, Shape> *shapeMap* = **new** Hashtable<String, Shape>();

**public** **static** Shape getShape(String shapeId) {

Shape cachedShape = *shapeMap*.get(shapeId);

**return** (Shape) cachedShape.clone();

}

// for each shape run database query and create shape

// shapeMap.put(shapeKey, shape);

// for example, we are adding three shapes

**public** **static** **void** loadCache() {

Circle circle = **new** Circle();

circle.setId("1");

*shapeMap*.put(circle.getId(),circle);

Square square = **new** Square();

square.setId("2");

*shapeMap*.put(square.getId(),square);

Rectangle rectangle = **new** Rectangle();

rectangle.setId("3");

*shapeMap*.put(rectangle.getId(), rectangle);

}

}

**Class: PrototypePatternDemo**

**package** shape;

**public** **class** PrototypePatternDemo {

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

ShapeCache.*loadCache*();

Shape clonedShape = (Shape) ShapeCache.*getShape*("1");

System.***out***.println("Shape : " + clonedShape.getType());

Shape clonedShape2 = (Shape) ShapeCache.*getShape*("2");

System.***out***.println("Shape : " + clonedShape2.getType());

Shape clonedShape3 = (Shape) ShapeCache.*getShape*("3");

System.***out***.println("Shape : " + clonedShape3.getType());

}

}

Result:

Shape : Circle

Shape : Square

Shape : Rectangle