

The background is a dark blue gradient with a subtle pattern of white dots. Overlaid on the left side are several concentric circles and arcs in a lighter blue color. Some of these arcs have degree markings, such as 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, and 260. There are also small arrows pointing in various directions, suggesting a sense of rotation or movement.

MODIFIERS AND DECLARATIONS

Ram Sharma

public, <default>, protected, private

- Applicable for class, interface, method and variable.
- If something is public, we can access it from anywhere.
- If something is <default>, we can access it within the package.
- If something is protected, we can access it within the package as well as in the child class.
- If something is private, we can access it within the class only.

Access Modifier	within class	within package	outside package by subclass only	outside package
private	Y	N	N	N
<default>	Y	Y	N	N
protected	Y	Y	Y	N
public	Y	Y	Y	Y

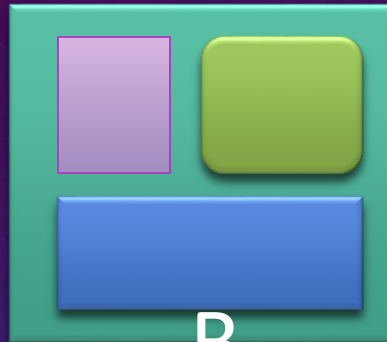
static

- Only applicable for variables and methods but not for classes.
- Only one copy is created for the class and shared by all its objects.
- You need not create an object of a class in order to use its static members. They should be accessed with class name.
- static members can be directly accessed from both instance and static areas where as instance members cannot be directly access in static areas.
- Let us understand: `System.out.println()`
- `System` – It is a class of `java.lang` package. Not required to import.
- `out` – It is a static reference of type `PrintStream` inside `System` class.
- `println()` – It is a method of class `PrintStream`.

Class And Objects With Static Members



A



B



C



D



E



F



G



S

1. clean my room
2. clean gym
3. persons in colony
4. persons in a room
5. address

interface and abstract class

- Any service requirement Specification(SRS) or any contract between client and service provider or it is 100% pure abstract class.
- Advantages:
 - We achieve security as internal implementation is not provided.
 - Enhancement becomes easy.
- we can declare interface by using keyword *interface*.
- All the methods inside interface are by default public and abstract.
- All the variables inside interface are by default public, static and final.

Criteria	class	abstract class	interface
implementation	known	partially known	unknown
method	concrete	can have abstract	public and abstract
variable	no restriction	no restriction	public, static and final
Instantiation	yes	no	no
Constructor	yes	yes	no
Abstraction	0%	0-100%	100%