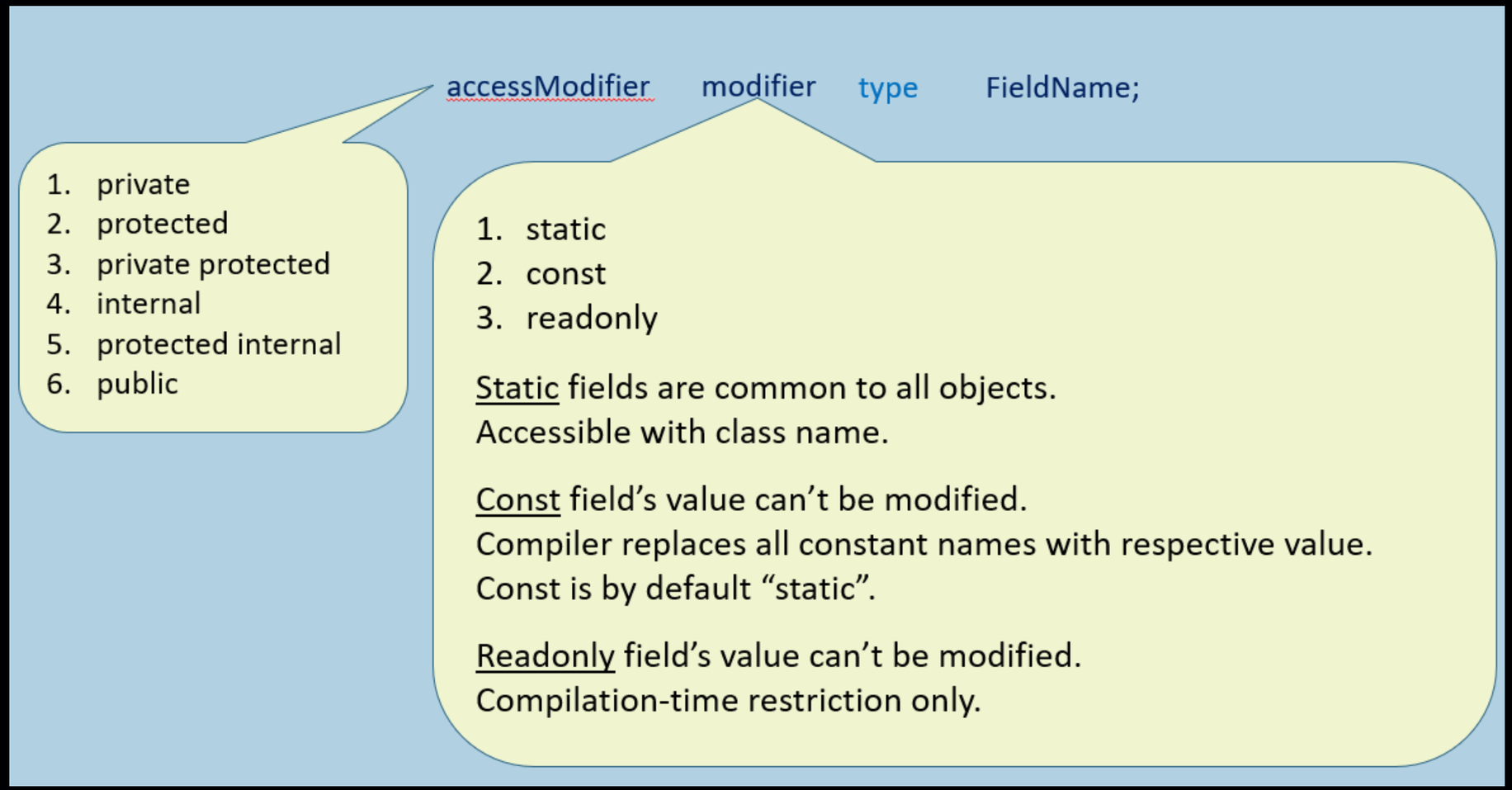
FIELDS

Variables that are declared in the class; are stored in the objects. solated for each object.



ACCESS MODIFIERS OF FIELDS

Access Modifier (a.k.a. "Access Specifier" or "Visibility Modifier) specifies the accessibility of fields, where the fields can be accessible; they provide security for the fields.

Access Modifier	In the same class	In the child classes at the same assembly	In the other classes at the same assembly	Child classes at other assembly	Other classes at other assembly
private	Yes	No	No	No	No
protected	Yes	Yes	No	Yes	No
private protected	Yes	Yes	No	No	No
internal	Yes	Yes	Yes	No	No
protected internal	Yes	Yes	Yes	Yes	No
public	Yes	Yes	Yes	Yes	Yes

MODIFIERS OF FIELDS

Static Fields

- Static fields are stored outside the object.
- Static fields are common to all objects of a class.

MODIFIERS OF FIELDS

Instance Fields (vs) Static Fields

Topic	**Instance Fields**	**Static Fields**
Storage	Stored in Objects	Stored in Class's memory.
Related to	Represents data related to objects.	Represents common data that belongs to all objects.
Declaration	Declared without "static" keyword. Syntax: type fieldName;	Declared with "static" keyword. Syntax: static type fieldName;
Accessible with	Accessible with object (through reference variable).	Accessible with class name only (not with object).
When memory gets allocated	Allocated separately for each object, because instance fields are stored "inside" the objects.	Allocated only once for the entire program; i.e. when the class is used for the first time while executing the program.

MODIFIERS OF FIELDS

Constant Fields

- Constant Fields are like static fields, that are common to all objects of the class
- We can't change the value of the constant field.
- Constant Fields are accessible with class name [not with object].
- Constant Fields are not stored in the object; will not be stored anywhere.
- Constant Fields will be replaced with their value, while compilation; so it will not be stored anywhere in memory.
- Constant Fields must be initialized, in line with declaration (with a literal value only).
- Constants can also be declared as 'local constants' (in a method).
- **Syntax:** AccessModifier const type FieldName = value;

MODIFIERS OF FIELDS

Readonly Fields

- Readonly Fields are like instance fields, that are stored in every object, individually.
- We can't change the value of readonly field.
- Readonly Fields are accessible with reference variable [with object].
- Readonly Fields must be initialized, either "in-line with the declaration" [or] "in the constructor".
- Syntax: AccessModifier readonly DataType FieldName = value;

MODIFIERS OF FIELDS

Key Points to Remember

- Fields are variables that are declared in the class; but stored in objects.
- Access modifiers of fields: private, protected, private protected, internal, protected internal, public
- Modifiers of fields: static, const, readonly
- Instance fields are individual for each object; Static fields are common (one-time) for all objects.
- Constants must be initialized along with declaration; Readonly fields must be initialized either 'along with declaration' or in 'instance constructor'.

METHODS

- Method is a function (group of statements), to do some process based on fields.
- Methods are parts of the class.
- Methods can receive one or more input values as "parameters" and return a value as "return".