**Why not the outer class be other than public in java?**

If private 🡪 you can’t extend the class and thus it is scope is hidden for the outside classes. When something is not accessible for other class, then there is no point in creating one. So private is not allowed

If protected 🡪 when something is declared as protected, then it can be accessed only by extending it in child class but not by all other classes. Hence it is not allowed

**Why an abstract class can’t be final and static?**

Final 🡪 if something is declared as Final, then we can’t change it. The problem is in abstract classes, we just declare the method and not define it, bcoz the implementation will be carried in the class which extracts it. So if we made abstract methods are final, then we can’t implement them in the child class

Static 🡪 Static is common to all but as per abstract’s class requirement, it should be available only when we extend it. So static is not allowed in abstract class since if we declare abstract class as static then the abstract class will become common to all though they are not extending it.

**Why “this” keyword is not allowed in static?**

When something is declared as static, which is common to all. Also it can access non-static member only by using “Object reference variable”. The keyword “this” is limited to “Specific” member and not to all.

**Can an outer class be static?**

No the outer class can’t be static but the inner / nested class be static.

**What is checked and unchecked exception?**

**Checked 🡪** When you java standalone application deals with external resources like File reading, DB connection with SQL etc., compiler itself will make the user to handle the exception using try – catch block, to avoid unexpected loss of data… Compare this scenario with continuous hard shutdown of a pc without closing the running files properly. Later sometime it will display that boot up issue. Eg., FileNotFound, IO Exception

**Unchecked 🡪** When java application does not deal with external resource eg., **Arithmetic, ArrayIndexOutOfBound, NullPointer exception**

**Why static block executed before main method?**

In layman language, executing a program is nothing but interacting with the app. Assume when you walk into the ATM, what-if you could not see the home screen to select language, which makes us to assume the ATM is not working. So before you interact with the app, there are few info which needs to be displayed to the end user and using which we start the interaction. So to achieve this we need the static block and must execute before main method

**Why a main method is public and static?**

Unless a main method is declared as not static(Common), it is available only to specific members of a class and hence not all can access the application. To overcome this we declare this as static as well as public is to make it available to across the packages

**Why can’t we override a constructor?**

Override will come into picture only when we talk about parent / child class i.e., the method declared in parent class should be defined in child class with **same name and signature** to achieve override. Now constructor is special method where it will not have return type and same name as class name. So we can’t do constructor override since in child class you can’t declare the constructor with parent class name as per the definition of constructor. Even if you declare it will be treated as a method and not as constructor

**Boxing** 🡪 Converting primitive data types into objects is called Boxing and achieved using

Integer.valueOf(a);

Where a is of type int.