

DIFFERENCE B/W INTERFACES & ABSTRACT CLASS.

Syntax level

1. In Interface D.M. are public and static
Not in Abstract class.
2. In Interface to give ^{body to} non-static fn, 'default' is used. Not No keyword needed in Abstract class.
3. In Abstract class, Abstract D.M. are inherited
Not in interface.
4. In Interfaces Functions are public, Not in Abstract class.
5. More than one Interface can be implemented
but only one class can be extended.

Conceptual level.

1. If more than one parent has to perform same task we use interfaces.
2. Interfaces increases performance as
Whenever we load any class into memory
lots of meta-data also come into memory.
Like: info. of parent class
info of associated class
This degrades performance of child class.

3. For \rightarrow Object + Abstraction : Go for Abstract class

Only Abstraction : Go for interface

4. Abstraction + Reusability : Go for Abstract class.

Only Abstraction : Go for interfaces.