Abstract class:-

i) Supports single inheritence only.

ii) Contain abstract methods.

iii) Can contain concrete methods.

iv) Can contain instance variables and constructors.

Limitation: Do not support multiple inheritence

Interface:

i) Supports multiple inheritence.

ii) Contains abstract methods.

iii) Contains constants.

iv) Also contains default and static methods.

v) No constructors or instance variables.

Advantage: Allows a class to inherit behaviour from multiple sources, enabling flexible design.

□ When to use abstract class:-

i> When classes are closely related.

ii> When you need to share code, state or constructors.

iii> When partial implementation is required.

□ When to use interface:-

i> When multiple inheritance is needed.

ii> When defining capabilities or roles.

iii> When classes are unrelated but share common behaviour.

iv> When loose coupling is required.