* Which keyword is to be used when class A inherits from class B?

*extends*

* Which keyword is to be used when class A implements an interface B?

*Implements*

* Can class A inherit from multiple classes?

*No*

* Can class A inherit multiple interfaces?

*Yes*

* Can class A inherit from class B and implement interface C at the same time?

*Yes*

* Which keyword is to be used when interface A inherits from interface B?

*Implements*

* Can interface A inherit multiple interfaces?

*Yes*

* Can interface A inherit from class B?

*No*

* What does the term “concrete class” imply?

*That this class is extended by another class.*

* What does the term “parent class” imply?

*That this class extends another class.*

* What is the difference between a regular class and an abstract class?

*An abstract class does usually not have any or only partial implementations.*

* What is the difference between a regular method and an abstract method?

A*n abstract method does not have any method body. The only essentials are the declaration and method signature.*

* Can an abstract class hold a constructor?

*Yes*

* Can an abstract class hold multiple constructors?

*Yes*

* Can a regular class hold abstract methods?

*No*

* Can an abstract class hold regular and abstract methods at the same time?

*Yes*

* Can an abstract class hold no abstract methods at all?

*Yes*

* Can a concrete class A override a regular method of its parent class B?

*Yes*

* Can a concrete class A override an abstract method of its parent class B?

*Yes*

* Is it possible that a concrete class A inherits from an abstract class B and does not provide logic for all abstract methods?

*Yes*

* Must an abstract class A implement all methods from an Interface B it uses?

*Yes*

* Can you assign other access modifiers to interface methods apart from public?

*No*

* If an interface A inherits from Interface B does it have to implement all methods from interface B?

*No*

* What is the purpose of the method super()?

*Super() is used to call the immediate parent of the concrete class.*

* How can a method super() of a concrete class A talk to a specific constructor of its parent class B?

*Through the right parameters.*

* What is the difference between super() and this?

Super() refers an instance of the immediate parent class while this refers to an instance of the current class.

* Can an interface or a class be generic? What does the term “generic” imply?

*Yes, generics are a way to achieve code reusability by defining generic classes.*

* How can a generic character be imposed on an interface?

*With diamond brackets <>.*

* How can a generic character be imposed on a class?

*With diamond brackets <>.*

* Can you find any naming parameter conventions such as T-Type online?

*E: Element, N: Number, K: Key, V: Value*