**What is the difference between abstract class and interface**

|  |  |
| --- | --- |
| **Abstract** | **Interface** |
| An abstract class is declared using the abstract keyword, and it can have both abstract and concrete methods. Abstract methods are declared without implementation and must be implemented by any concrete subclass. | An interface is declared using the interface keyword, and it consists only of method signatures. All methods declared in an interface must be implemented by the classes that implement the interface. |
| PHP supports single inheritance for classes, meaning a class can extend only one other class. | However, PHP supports multiple interface inheritance, allowing a class to implement multiple interfaces. |
| Abstract classes can have constructors, and they are invoked when a concrete subclass is instantiated. | Interfaces cannot have constructors. |
| PHP does not support default method implementations in interfaces as of PHP 7.4 and earlier. Each method declared in an interface must be implemented by the implementing class. | PHP 8.0 introduced the ability to include default method implementations in interfaces using the default keyword. |
| In PHP, abstract class methods can have various visibility modifiers (public, protected, private). | In PHP versions before 7.1, interface methods are implicitly public. In PHP 7.1 and later, interface methods can have explicit visibility modifiers (public, protected, private). |