الفرق بين primary key و unique

1. Primary key:

- primary key must continue auniqe value for each row of data

- يجب أن يكون القيم في ال primary key فريدة من نوعها ولا يمكن تكرارها notnull

- يسمح ال primary key بقبول قيم (NULL) .

2. Unique:

- يستخدم ال unique لضمان أن القيم في العمود مختلفة ولا يمكن تكرارها.

- يمكن أن يحتوي ال unique على قيم فارغة (NULL) ولكن يسمح بقيمة واحدة فقط بأن تكون NULL.

باختصار، ال primary key يستخدم لتحديد قيمه فريده للجدول بينما يستخدم ال unique لضمان عدم تكرار القيم في العمود.

Foriegn key

Akey or value references a primary key in another table

Better to be declared as same as its references primarykey

Magic methods

are special methods that are predefined in classes and are invoked automatically under certain circumstances.

These methods are prefixed with two underscores () to distinguish them from regular methods.

Some of the most commonly used magic methods in PHP are:

1. construct(): This method is called when an object is created and allows you to initialize the object's properties.

2. destruct(): This method is called when an object is destroyed and allows you to perform any cleanup tasks.

3. get(): This method is called when trying to access a property that is not accessible or does not exist.

4. set(): This method is called when trying to assign a value to a property that is not accessible or does not exist.

5. call(): This method is called when trying to call a method that is not accessible or does not exist.

6. toString(): This method is called when trying to convert an object to a string.

By using magic methods, you can add functionality to your classes without explicitly defining all the methods. This can make your code more concise and easier to maintain.

1. Definition:

- Abstract class: An abstract class is a class that cannot be instantiated on its own and is designed to be a base class for other classes to inherit from. It can contain both abstract and non-abstract methods

- -

- Interface: An interface is a blueprint of a class that defines a set of methods that a class must implement. It does not contain any implementation of methods, only method signatures.

2. Inheritance:

- Abstract class: A class can only inherit from one abstract class due to the single inheritance limitation in most programming languages.

- Interface: A class can implement multiple interfaces, allowing for more flexibility in defining the behavior of a class.

3. Implementation:

- Abstract class: An abstract class can have both abstract methods (methods without implementation) and concrete methods (methods with implementation).

-

- Interface: An interface can only have method signatures and does not contain any implementation. The class implementing the interface must provide the implementation for all the methods defined in the interface.

4. Access modifiers:

- Abstract class: Abstract classes can have a mix of access modifiers for their methods, including public, protected, and private.

-

- Interface: All methods in an interface are public by default and cannot have any other access modifiers.

5. Multiple inheritance:

- Abstract class: Some programming languages do not support multiple inheritance for abstract classes, while others allow it with certain restrictions.

-

- Interface: Interfaces are a way to achieve multiple inheritance in languages that do not support it for classes. A class can implement multiple interfaces to inherit

- polymorphism

قبل مانعرف اي ال polymorphism

overriding

تخيل كده عندنا function من 200 سطر كود وعايز استخدم نفس الفانكشن ده ف كلاس تاني هتقولي اعمل inheritance واستخدم الفانكشن ده هقولك حلو بس انا عايز ازود كام سطر كود تاني ف الفانكشن ده

بدل مااقعد اكتب ال ٢٠٠ سطر كود من الاول عشان ازود الكام سطر اللي انا عايزاهم جه ال oop قالك اعمل overriding يعني اي

يعني اعمل فانكشن بنفس الاسم الفانكشن اللي فيها ٢٠٠ سطر كود واكتب كل الجديد اللي انت عايز تضيفه بس

تاني حاجه لازم نعرفها هيا ال overloading

مثلاً عايزه اقسم رقمين ٣و ٦ هبعتهم لفانكشن اسمها divide -> divide(6,3) هترجعلي 2 ده integer طب لو بعت 5 و 2 -> divide(5,2) هترجع 2 وده الصح لان الفانكشن ده بترجع integer بس طب انا عايزاها ترجعلي القيم الحقيقة كده هعمل فانكشن تانيه بترجعلي float اسميها مثلاً divide\_float طب ماكده هزود عدد الفانكشن اللي مفروض احفظها والموضوع هيبقا صعب ومش هتبقا حاجه كويسه ان يبقا عندي كذا فانكش عشان اعمل نفس العمليه اللي هيا القسمه جه ال oop قالك اعمل overloading يعني اي يعني اعمل فانكشن تانيه اسمها divide بردو بس ب return float وكده تقدر تستخدم نفس اسم الفانكش وتبعتلها اي حاجه وانا هعملك اللي انت عايزه وهبعتلك الناتج الصح

بعد مافهمنا ال overriding و ال overloading كده نبقا حققنا مبدأ ال polymorphism يعني يكون عندي interface واحده مثلاً فانكشن واحده واقدر استخدمها كذا مرة ب implementation

The instanceof keyword in PHP is used to determine whether an object is an instance of a specific class or implements a specific interface. It returns true if the object is an instance of the specified class or implements the specified interface, and false otherwise.

Here is an example of how the instanceof keyword is used in PHP:

class Animal {

// Class definition

}

class Dog extends Animal {

// Class definition

}

$animal = new Animal();

$dog = new Dog();

if ($animal instanceof Animal) {

echo "animal is an instance of Animal class";

}

if ($dog instanceof Animal) {

echo "dog is an instance of Animal class";

}

if ($dog instanceof Dog) {

echo "dog is an instance of Dog class";

}

Constructor promotion:

instead of doing this:

class CustomerDTO

{

public string $name;

public string $email;

public DateTimeImmutable $birth\_date;

public function \_\_construct(

string $name,

string $email,

DateTimeImmutable $birth\_date

) {

$this->name = $name;

$this->email = $email;

$this->birth\_date = $birth\_date;

}

}

You would write this:

class CustomerDTO

{

public function \_\_construct(

public string $name,

public string $email,

public DateTimeImmutable $birth\_date,

) {}

}

can we define a constructor as a private constructor? if it yes when could we do this?

No, PHP does not support method overloading like some other programming languages such as Java. In PHP, you cannot have multiple methods with the same name but different parameters.

If you want to achieve similar functionality, you can use variable-length argument lists or default parameter values in PHP functions. This way, you can create a single function that can handle different numbers of arguments or different types of arguments.

You can also use the funcnumargs() and funcgetargs() functions to get the number and values of arguments passed to a function dynamically.

does php has multiple constructor or not and how to reach out?

PHP does not support multiple constructors in the same way that some other programming languages do. However, you can achieve similar functionality by using default parameter values in the constructor.

For example, you can define a constructor with default parameter values like this:

class MyClass {

public function \_\_construct($param1 = null, $param2 = null) {

// Constructor logic here

}

}

Then, you can create instances of the class with different sets of parameters like this:

$obj1 = new MyClass(); // Uses default parameter values

$obj2 = new MyClass('value1'); // Sets param1 to 'value1' and uses default value for param2

$obj3 = new MyClass('value1', 'value2'); // Sets both param1 and param2

.