

АБСТРАКТТЫ КЛАСТАР МЕН ӘДІСТЕР. МҰРАГЕРЛІК

АРАЛАС САБАҚ



МҰРАГЕРЛІК

- Мұрагерлік объектіге бағытталған бағдарламалаудың негізгі аспектілерінің бірі болып табылады. Мұрагерлік классқа бұрыннан бар класстардың функционалдығын алуға және қажет болған жағдайда оны қайта анықтауға мүмкіндік береді. Егер бізде бірнеше функция жетіспейтін класс болса, онда кодты қайта жазу арқылы нөлден жаңасын жасаудан гөрі, бірнеше жолды жазу арқылы бар классты қайта анықтау әлдеқайда оңай.
- Бір классты екіншісінен мұра ету үшін біз `extends` операторын қолдануымыз керек. Айта кету керек, PHP-де біз классты тек бір класстан мұра ете аламыз. Бірнеше мұрагерлік сақталмайды.

```
<?php
class Person
{
    public $name;
    function __construct($name)
    {
        $this->name = $name;
    }
    function displayInfo()
    {
        echo "Имя: $this->name<br>";
    }
}
class Employee extends Person
{
    public $company;
    function __construct($name, $company)
    {
        $this->name = $name;
        $this->company = $company;
    }
    function displayInfo()
    {
        echo "Имя: $this->name<br>";
        echo "Работает в $this->company<br>";
    }
}

$tom = new Employee("Tom", "Microsoft");
$tom -> displayInfo();
?>
```

МЫСАЛ:

Бұл жағдайда Person класы адамды тұтастай білдіреді, ал Employee класы белгілі бір кәсіпорынның қызметкері деп есептейік. Осыған байланысты әр қызметкер адамды білдіреді. Бұл жұпта Person класы ата - ана немесе базалық класс деп те аталады, ал Employee класы туынды класс немесе мұрагер класс деп аталады.

```
<?php
//Базалық класс
class Employee{
public $aty;
public $otdelenie;//модификаторлар
function __construct($aty,$otd)
{
$this->aty = $aty;
$this->otd = $otd;
}
}//мұрагерлік класс
class Executive extends Employee{

//айнымалылар
public $dolzhnost;
public $salary;

function __construct($aty,$dolzhnost,$otd,$salary)
{
//базалық класс объектісі
parent::__construct($aty,$otd);
$this->dolzhnost = $dolzhnost;
$this->salary = $salary;
}

//Печать сведений о сотруднике
public function shygary()
{
echo "<b>Қызметкер туралы ақпарат:</b><br/>";
echo "Аты: ".$this->aty."<br/>";
echo "Должность: ".$this->dolzhnost."<br/>";
echo "Отделение: ".$this->otd."<br/>";
echo "Зарплата: $".$this->salary."<br/>";
}
}

//мұрагерлік класс объектісі
$objExecutive=new Executive('Aidar','Programmist','Bank HALYK BANK',150000);
//мұрагерлік класс әдісі
$objExecutive->shygary();

?>
```

- Абстракттілі класстардың негізгі ерекшеліктерінің бірі - біз оның конструкторын шақыру арқылы абстракттілі класс объектілерін тікелей жасай алмаймыз:
-

- **abstract class Messenger { }**
- **\$telegram = new Messenger();** \\ бұл жол жұмыс істемейді

```
abstract class Person {  
  
    private $firstName = "";  
    private $lastName = "";  
  
    public function setName( $firstName, $lastName ) {  
        $this->firstName = $firstName;  
        $this->lastName = $lastName;  
    }  
  
    public function getName() {  
        return "$this->firstName $this->lastName";  
    }  
  
    abstract public function showWelcomeMessage();  
}
```

-
- Абстракттілі класс класс атауының алдында abstract модификатормен анықталады:

-

```
abstract class Messenger { }
```

-
- Абстракттілі класстар, әдеттегі класстар сияқты, айнымалылар мен тұрақтыларды, әдістер мен конструкторларды анықтай алады. Бірақ абстракттілі сыныптардың тағы бір ерекшелігі-оларда абстракттілі әдістер болуы мүмкін. Бұл іске асырылмаған әдістер. Олар үшін мұрагер класстар жүзеге асырады. Мысалы:
 - `abstract class Messenger`
 - `{`
 - `abstract function send($message); // абстракт әдіс`
 - `}`
 - Абстракттілі әдісті анықтау кезінде `abstract` модификатор `function` сөзінің алдына қойылады. Ал әдіс параметрлерінің тізімінен кейін – нүктелі үтір қойылады. Абстракттілі әдістерді тек абстракттілі класстарда ғана орналастыруға болады. Кәдімгі абстракттілі класста абстракттілі әдістер болуы мүмкін емес.

ЕСЕПТЕР ШЫҒАРУ

Студент классын(аты, адрес) және Студент бөлімі, мұрагерлік класс (тобы , курсы) құрамыз .

Persons классынан мұра болатын бағдарламашы классын жасаңыз. Жаңа класста бағдарламашы білетін тілдер массиві сақталатын langs парамаетрі болсын.