

ЛАБОРАТОРНА РОБОТА № 4

Тема: Об'єктно-орієнтоване програмування в PHP

Мета роботи: навчитися працювати з класами

Завдання 1. (Організація класів по каталогах в проєкті)

- Створіть пустий проєкт PHP.
- Створіть каталоги: "Models", "Controllers", "Views".
- У кожному каталозі створіть по одному класу, наприклад, "UserModel", "UserController", "UIView".
- В кожному класі реалізуйте просту функціональність, наприклад, виведення повідомлення чи повернення значень.

Завдання 2. (Автопідключення класів за допомогою spl_autoload_register. PHPDoc)

- Додайте PHPDoc коментарі до всіх класів, вказавши їх призначення та властивості.
- Створіть файл autoload.php, який буде містити функцію для автопідключення класів.
- Використайте spl_autoload_register для автоматичного підключення класів на основі їхніх імен та розташування.

Завдання 3. (Неймспейси)

- Додайте неймспейси до класів у попередньому завданні. Наприклад, "namespace Models;" для "UserModel".
- Змініть файл autoload.php так, щоб він також враховував неймспейси при підключенні класів

Завдання 4. (Автопідключення класів з неймспейсами)

- Використовуйте аналогічний підхід до підключення класів, але тепер з урахуванням неймспейсів.
- Переконайтеся, що класи виводять повідомлення чи результати виклику.

					ДУ«Житомирська політехніка».25.121.27.000 – Лр4			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Семенчук О.А.			Звіт з лабораторної роботи		Літ.	Арк.
Перевір.		Дмитренко І.А.						Аркушів
Керівник								1
Н. контр.								13
Зав. каф.							ФІКТ Гр. ІПЗ-23-1[2]	

Лістинг класу UserController:

```
<?php

namespace Controllers;

/*
 * Example class
 *
 * @package Lab 4
 * @subpackage Controllers
 * @author Олексій Семенчук
 */
class UserController
{
    /*
     * Displays the given Model
     *
     * @access public
     * @return void
     * @param UserView $View
     * @param UserModel $Model
     */
    public function GetController($View, $Model): void
    {
        echo $View->GetView($Model->GetModel()->name) . "\n";
    }
}
```

Лістинг класу UserModel:

```
<?php

namespace Models;

/*
 * Example class
 *
 * @package Lab 4
 * @subpackage Models
 * @author Семенчук Олексій
 */
class UserModel
{
    /*
     * Returns a Person model
     *
     * @access public
     * @return object
     */
    public function GetModel(): object
    {
        $Person = (object)array(
```

		Семенчук О.А.			ДУ «Житомирська політехніка».25.121.27.000 – Лр4	Арк.
		Дмитренко І. А.				
Змн.	Арк.	№ докум.	Підпис	Дата		2

```

        "name" => "FearessAtom",
        "age" => "18",
    );

    return $Person;
}
}

```

Лістинг класу UserView:

```

<?php

namespace Views;

/*
 * Example class
 *
 * @package Lab 4
 * @subpackage Views
 * @author Олексій Семенчук
 */
class UserView
{
    /*
     * Wrapps a given string in a HTML <h1> tag
     *
     * @access public
     * @return void
     * @param string $Title
     */
    public function GetView(string $Title): string
    {
        return "<h1>" . $Title . "</h1>";
    }
}

```

Лістинг файлу autoload.php:

```

<?php

spl_autoload_register(function ($class)
{
    $class = str_replace("\\", "/", $class);

    $ext = ".php";

    $path = "";

    $file_full_path = $path . $class . $ext;

    if (!file_exists($file_full_path))
    {
        return false;
    }
}

```

		Семенчук О.А.			ДУ «Житомирська політехніка».25.121.27.000 – Лр4	Арк.
		Дмитренко І. А.				
Змн.	Арк.	№ докум.	Підпис	Дата		3

```

        include_once $file_full_path;
    });

    Лістинг файлу autoload.php:
<?php

include_once "autoload.php";

$Controller = new Controllers\UserController();
$Model = new Models\UserModel();
$View = new Views\UserView();

$Person = $Model->GetModel();
echo $Person->name . "\n";

echo $View->GetView("Title") . "\n";

$Controller->GetController($View, $Model);

```

Результат:

```

[/var/www/labs/task1-4] -> php index.php
FearessAtom
<h1>Title</h1>
<h1>FearessAtom</h1>

```

Рис. 1 Результат завдань 1-4

Завдання 5. (Створення класу. Методи GET і SET)

1. Створіть клас Circle з полями: координати центру і радіус кола
2. Створіть конструктор, що приймає значення для 3-х полів
3. Створіть метод __toString(), що повертає рядок в форматі: «Коло з центром в (x, y) і радіусом radius»
4. Створіть методи GET і SET для всіх 3-х полів
5. Створіть об'єкт та перевірте всі його методи

Завдання 6. (Модифікатори доступу)

1. В класі з попереднього завдання зробіть всі поля private.
2. Створіть метод, що приймає об'єкт коло, і повертає true, якщо дані кол перетинаються, і false, якщо вони не перетинаються.

Лістинг класу Circle:

```

<?php

class Circle
{
    private float $X;
    private float $Y;
    private float $Radius;

```

		Семенчук О.А.			ДУ «Житомирська політехніка».25.121.27.000 – Лр4	Арк.
		Дмитренко І. А.				
Змн.	Арк.	№ докум.	Підпис	Дата		4

```

function __construct(float $X, float $Y, float $Radius)
{
    $this->SetX($X);
    $this->SetY($Y);
    $this->SetRadius($Radius);
}

function Intersects(Circle $Circle): bool
{
    $Distance = sqrt(pow($Circle->GetX() - $this->GetX(), 2) +
pow($Circle->GetY() - $this->GetY(), 2));

    return $Distance <= sqrt($this->Radius) + sqrt($Circle-
>GetRadius());
}

function __toString(): string
{
    return "A circle at ({ $this->X}, { $this->Y}) with radius of
{ $this->Radius}";
}

public function GetX(): float
{
    return $this->X;
}

public function SetX(float $value): void
{
    $this->X = $value;
}

public function GetY(): float
{
    return $this->Y;
}

public function SetY(float $value): void
{
    $this->Y = $value;
}

public function GetRadius(): float
{
    return $this->Radius;
}

public function SetRadius(float $value): void
{
    $this->Radius = abs($value);
}
}

```

		Семенчук О.А.			ДУ «Житомирська політехніка».25.121.27.000 – Лр4	Арк.
		Дмитренко І. А.				
Змн.	Арк.	№ докум.	Підпис	Дата		5

Лістинг файлу index.php:

```
<?php

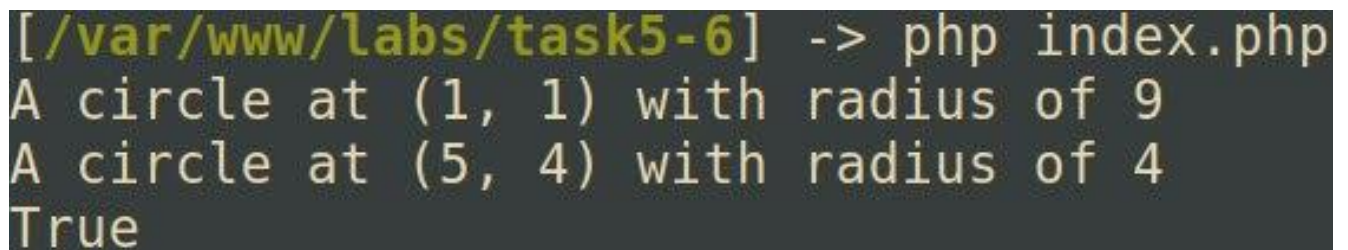
include_once "../Circle.php";

$a = new Circle(1, 1, 9);
$b = new Circle(5, 4, 4);

echo $a . "\n";
echo $b . "\n";

echo ($a->Intersects($b) ? "True" : "False") . "\n";
```

Результат:



```
[/var/www/labs/task5-6] -> php index.php
A circle at (1, 1) with radius of 9
A circle at (5, 4) with radius of 4
True
```

Рис. 2 Результат завдань 5-6

Завдання 7. (Статичні властивості і методи)

1. Створіть директорію text, а в ній 3 текстових файла
 2. Створіть клас зі статичним полем dir="text"
 3. Створіть 2 статичних методи в класі: на читання та запис в файл:
 - Ім'я файлу передається як параметр метода.
 - В метод «на запис в файл» передається ще й рядок, який потрібно дописати в файл.
 - Директорія береться зі статичного поля
 4. Створіть метод, що дозволяє стерти вміст файлу
- Перевірте роботу всіх методів

Лістинг класу File:

```
<?php

class File
{
    public static string $directory = "../text/";

    public static function Write(string $file_name, string $content): bool
    {
        $file_path = self::$directory . $file_name;

        if (!file_exists($file_path))
        {
            throw new Exception("File " . $file_path . " does not exist!");
        }
    }
}
```

		Семенчук О.А.			ДУ «Житомирська політехніка».25.121.27.000 – Лр4	Арк.
		Дмитренко І. А.				6
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    }

    $file = fopen($file_path, "a");

    if(!fwrite($file, $content))
    {
        return false;
    }

    fclose($file);

    return true;
}

public static function Read(string $file_name): string
{
    $file_path = self::$directory . $file_name;

    if (!file_exists($file_path))
    {
        throw new Exception("File " . $file_path . " does not
exist!");
    }

    $file_size = filesize($file_path);

    if ($file_size == 0)
    {
        return "";
    }

    $file = fopen($file_path, "r");
    $content = fread($file, filesize($file_path));

    fclose($file);

    return $content;
}

public static function Erase(string $file_name): bool
{
    $file_path = self::$directory . $file_name;

    if (!file_exists($file_name))
    {
        if (!file_exists($file_path))
        {
            throw new Exception("File " . $file_path . " does
not exist!");
        }
    }

    if(!$file = fopen($file_path, "w"))

```

		Семенчук О.А.			ДУ «Житомирська політехніка».25.121.27.000 – Лр4	Арк.
		Дмитренко І. А.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        {
            return false;
        }

        fclose($file);
        return true;
    }
}

Лістинг файлу index.php:
<?php

include_once "../File.php";

echo File::Read("text1.txt");

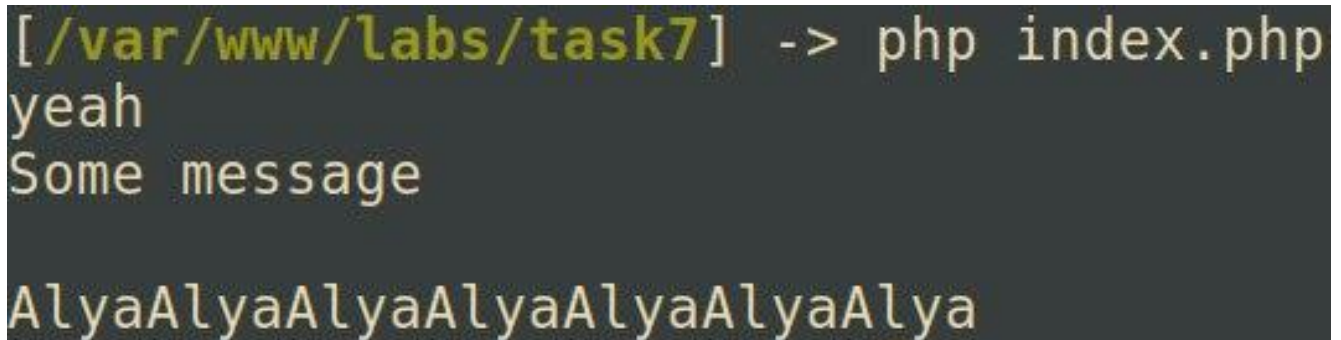
File::Write("text2.txt", "Some message");
echo File::Read("text2.txt") . "\n";

File::Erase("text2.txt");
echo File::Read("text2.txt") . "\n";

File::Write("text3.txt", "Alya");
echo File::Read("text3.txt") . "\n";

```

Результат:



```

[/var/www/labs/task7] -> php index.php
yeah
Some message
AlyaAlyaAlyaAlyaAlyaAlyaAlyaAlyaAlyaAlya

```

Рис. 3 Результат завдання 7

Завдання 8. (Наслідування)

- Створіть клас Human з властивостями, що характеризують людину (зріст, маса, вік...). Створіть методи GET і SET для кожної властивості
- Створіть клас Student, який успадковуватиметься від класу Human:
 - Додайте властивості, специфічні тільки для студента (назва ВНЗ, курс...)
 - Додайте в клас методи GET і SET для всіх нових властивостей.
 - Реалізуйте метод, який буде переводити студента на новий курс (тобто просто збільшувати значення поля «курс» на 1)
- Створіть клас Programmer, який успадковуватиметься від класу Human:
 - Додайте властивості, специфічні тільки для програміста (масив з мовами програмування, які він знає, досвід роботи...).
 - Додайте в клас методи GET і SET для всіх нових властивостей.

		Семенчук О.А.			ДУ «Житомирська політехніка».25.121.27.000 – Лр4	Арк.
		Дмитренко І. А.				8
Змн.	Арк.	№ докум.	Підпис	Дата		

- Реалізуйте метод, який буде додавати в масив з мовами ще одну мову. Перевірте роботу всіх класів і всіх методів. Не забудьте змінити зріст і масу у студентів і програмістів, скориставшись методами з батьківського класу Human

Завдання 9. (Абстрактні класи)

1. Зробіть клас Human абстрактним.
 2. Напишіть метод «Народження дитини» в класі Human, що викликає метод «Повідомлення при народженні дитини» (не забудьте поставити модифікатор protected), який буде абстрактним
 3. Реалізуйте «Повідомлення при народженні дитини» у класів Student та Programmer
- Перевірте роботу методів «народження»

Завдання 10. (Інтерфейси)

1. Створіть інтерфейс «Прибирання будинку», в якому опишіть 2 методи: «Прибирання кімнати» і «Прибирання кухні»
2. Додайте створений інтерфейс в клас Human
3. Реалізуйте у кожному класі-спадкоємці (Student та Programmer) обидва методи
4. Реалізація повинна бути у вигляді одного з рядків: «Студент прибирає кімнату», «Студент прибирає кухню», «Програміст прибирає кімнату», «Програміст прибирає кухню»,
5. Перевірте роботу методів прибирання в обох класах

Лістинг класу Human:

```
<?php
```

```
abstract class Human implements IHouseCleaning
{
    protected int $Height;
    protected int $Weight;
    protected int $Age;

    function __construct(int $Height, int $Weight, int $Age)
    {
        $this->SetHeight($Height);
        $this->SetWeight($Weight);
        $this->SetAge($Age);
    }

    abstract public function HaveBirth(): void;

    protected function GetHeight(): int
    {
        return $this->Height;
    }

    public function SetHeight(int $Height): void
    {

```

		Семенчук О.А.			ДУ «Житомирська політехніка».25.121.27.000 – Пр4	Арк.
		Дмитренко І. А.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        $this->Height = $Height;
    }

    public function GetWeight(): int
    {
        return $this->Weight;
    }

    public function SetWeight(int $Weight): void
    {
        $this->Weight = $Weight;
    }

    public function GetAge(): int
    {
        return $this->Age;
    }

    public function SetAge(int $Age): void
    {
        $this->Age = $Age;
    }
}

```

Лістинг класу Programmer:

```

<?php

include_once "../Interfaces/IHouseCleaning.php";

class Programmer extends Human
{
    private array $ProgrammingLanguages = array();
    private int $ExperienceYears = 0;

    /*
     * @param string[] $ProgrammingLanguages
     * @param int $ExperienceYears
     */
    function __construct(int $Height, int $Weight, int $Age, array
$ProgrammingLanguages = array(), int $ExperienceYears = 0)
    {
        parent::__construct($Height, $Weight, $Age);

        $this->ProgrammingLanguages = $ProgrammingLanguages;
        $this->ExperienceYears = $ExperienceYears;
    }

    public function CleanRoom(): void
    {
        echo "Student is cleaning his room!" . "\n";
    }

    public function CleanKitchen(): void

```

		Семенчук О.А.			ДУ «Житомирська політехніка».25.121.27.000 – Лр4	Арк.
		Дмитренко І. А.				10
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    {
        echo "Student is cleaning the kitchen!" . "\n";
    }

    function HaveBirth(): void
    {
        echo "A programmer was born!" . "\n";
    }

    public function AddProgrammingLanguage(string $Programming-
Language): void
    {
        $this->ProgrammingLanguages = array_merge($this-
>ProgrammingLanguages, array($ProgrammingLanguage));
    }

    public function GetProgrammingLanguages(): array
    {
        return $this->ProgrammingLanguages;
    }

    /*
    * @param array $ProgrammingLanguages
    */
    public function SetProgrammingLanguages(array $Programming-
Languages): void
    {
        $this->ProgrammingLanguages = $ProgrammingLanguages;
    }

    public function GetExperienceYears(): int
    {
        return $this->GetExperienceYears;
    }

    public function SetExperienceYears(int $ExperienceYears): void
    {
        $this->ExperienceYears = $ExperienceYears;
    }
}

```

Лістинг класу Student:

```

<?php

include_once "../Interfaces/IHouseCleaning.php";

class Student extends Human
{
    private string $EstablishmentName;
    private int $Course;

    function __construct(int $Height, int $Weight, int $Age, string
$EstablishmentName = "", int $Course = 1)

```

		Семенчук О.А.			ДУ «Житомирська політехніка».25.121.27.000 – Пр4	Арк.
		Дмитренко І. А.				11
Змн.	Арк.	№ докум.	Підпис	Дата		

```

{
    parent::__construct($Height, $Weight, $Age);

    $this->SetEstablishmentName($EstablishmentName);
    $this->SetCourse($Course);
}

public function CleanRoom(): void
{
    echo "Student is cleaning his room!" . "\n";
}

public function CleanKitchen(): void
{
    echo "Student is cleaning the kitchen!" . "\n";
}

public function HaveBirth(): void
{
    echo "A student was born!" . "\n";
}

public function IncrementCourse(): void
{
    $this->SetCourse($this->Course + 1);
}

public function GetEstablishmentName(): string
{
    return $this->EstablishmentName;
}

public function SetEstablishmentName(string $EstablishmentName):
void
{
    $this->EstablishmentName = $EstablishmentName;
}

public function GetCourse(): int
{
    return $this->Course;
}

public function SetCourse(int $Course): void
{
    $this->Course = $Course;
}
}

```

Лістинг інтерфейсу IHouseCleaning:

<?php

| | | | | | | |
|------|------|-----------------|--------|------|--|------|
| | | Семенчук О.А. | | | ДУ «Житомирська політехніка».25.121.27.000 – Лр4 | Арк. |
| | | Дмитренко І. А. | | | | 12 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```
interface IHouseCleaning
{
    public function CleanRoom(): void;
    public function CleanKitchen(): void;
}
```

Лістинг файлу index.php:

```
<?php

spl_autoload_register(function ($class)
{
    include_once "../Classes/" . $class . ".php";
});

$Programmer = new Programmer(183, 80, 18);

$Programmer->AddProgrammingLanguage("C#");
$Programmer->AddProgrammingLanguage("JavaScript");
$Programmer->AddProgrammingLanguage("Python");
$Programmer->AddProgrammingLanguage("PHP");

$Programmer->HaveBirth();

$Programmer->CleanRoom();

$Student = new Student(180, 74, 19);

$Student->SetEstablishmentName("ZTU");
$Student->SetCourse(2);

$Student->HaveBirth();

$Student->CleanKitchen();
```

Результат:

```
[/var/www/labs/task8-10] -> php index.php
A programmer was born!
Student is cleaning his room!
A student was born!
Student is cleaning the kitchen!
```

Рис. 4 Результат завдань 8-10

Git repository: <https://git.ztu.edu.ua/ipz/2023-2027/ipz-23-1/semenchuk-oleksiy/backend/-/tree/lab4>

Висновок: У ході виконання лабораторної роботи було розглянуто основи об'єктно-орієнтованого програмування в PHP. Було створено класи, реалізовано їх властивості та методи, а також застосовано принципи наслідування.

| | | | | | | |
|------|------|-----------------|--------|------|--|------|
| | | Семенчук О.А. | | | ДУ «Житомирська політехніка».25.121.27.000 – Лр4 | Арк. |
| | | Дмитренко І. А. | | | | 13 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |