
Разработка многостраничного сайта на PHP

ЗАНЯТИЕ № 7,8,9
ВВЕДЕНИЕ В ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

Тема занятия – введение в объектно-ориентированное программирование

Цель занятия –

Продолжить изучать объектно-ориентированное программирования на практике

Актуализация

На прошлом занятии мы учились на практике :

- создавать классы,
- вычисляли факториалы,
- вычисляли разницу между датами, используя методы ООП

Практическое занятие

Сегодня на занятии мы продолжим составлять классы, используя методы ООП в PHP.

Задача № 1

- Сделайте класс **Worker**, в котором будут следующие **public** поля - name (имя), age (возраст), salary (зарплата).
- Создайте объект этого класса, затем установите поля в следующие значения (не в __construct, а для созданного объекта) - имя 'Иван', возраст 25, зарплата 1000. Создайте второй объект этого класса, установите поля в следующие значения - имя 'Вася', возраст 26, зарплата 2000.
- Выведите на экран сумму зарплат Ивана и Васи. Выведите на экран сумму возрастов Ивана и Васи.

Задача № 2

- Сделайте класс **Worker**, в котором будут следующие **private** поля - name (имя), age (возраст), salary (зарплата) и следующие **public** методы setName, getName, setAge, getAge, setSalary, getSalary.
- Создайте 2 объекта этого класса: 'Иван', возраст 25, зарплата 1000 и 'Вася', возраст 26, зарплата 2000.
- Выведите на экран сумму зарплат Ивана и Васи. Выведите на экран сумму возрастов Ивана и Васи.

Задача № 3

- Дополните класс **Worker** из предыдущей задачи **private** методом `checkAge`, который будет проверять возраст на корректность (от 1 до 100 лет). Этот метод должен использовать метод `setAge` перед установкой нового возраста (если возраст не корректный - он не должен меняться).

Задача № 4

- Сделайте класс **Worker**, в котором будут следующие **private** поля - name (имя), salary (зарплата). Сделайте так, чтобы эти свойства заполнялись в методе `__construct` при создании объекта (вот так: `new Worker(имя, возраст)`). Сделайте также **public** методы `getName`, `getSalary`.
- Создайте объект этого класса 'Дима', возраст 25, зарплата 1000. Выведите на экран произведение его возраста и зарплаты.

Задача № 5

- Сделайте класс **User**, в котором будут следующие **protected** поля - name (имя), age (возраст), **public** методы setName, getName, setAge, getAge.
- Сделайте класс **Worker**, который наследует от класса User и вносит дополнительное **private** поле salary (зарплата), а также методы **public** getSalary и setSalary.
- Создайте объект этого класса 'Иван', возраст 25, зарплата 1000. Создайте второй объект этого класса 'Вася', возраст 26, зарплата 2000. Найдите сумму зарплата Ивана и Васи.
- Сделайте класс **Student**, который наследует от класса User и вносит дополнительные **private** поля стипендия, курс, а также геттеры и сеттеры для них.

Задача № 6

- Сделайте класс **Driver** (Водитель), который будет наследоваться от класса **Worker** из предыдущей задачи. Этот метод должен вносить следующие private поля: водительский стаж, категория вождения (А, В, С).

Рефлексия

Сегодня на практическом занятии мы отработали те теоретические навыки, которые мы получили на прошлом занятии.

Ответьте, пожалуйста, на несколько вопросов:

1. Какая задача была самая интересная?
2. Какая задача вам показалась наиболее легкой и почему?
3. Какая задача вам показалась наиболее тяжелой и почему?
4. Остались ли у вас какие-нибудь вопросы после нашего занятия?



**Спасибо
за
внимание**