## Теоретическая часть

* Наследование в PHP, абстрактные классы, методы
* Интерфейсы и наследование в PHP
* ООП в PHP: пространства имен **namespace, use**
* ООП в PHP: автозагрузка классов, функции **\_\_autoload(имяКласса)**, **spl\_autoload\_register(имяФункции)**, **spl\_autoload()**
* Сериализация скалярных переменных, массивов, объектов в строковое представление
* Десериализация скалярных переменных, массивов, объектов из строкового представления
* Магические методы **\_\_sleep()**, **\_\_wakeup()** для сериализации и десериализации объектов

## Практическая часть

Разработайте приложение PHP. На главной странице разместите это задание, на других страницах – решение задач. Доступ к функционалу приложения открывается только после создания сессии (клик по кнопке «Войти»). По клику на кнопку «Выйти» удаляется сессия, запрещается доступ к функционалу.

В файле формата CSV хранить данные о моменте начала и завершения сессии (начало сессии – момент нажатия кнопки «Вход», окончание сессии – момент нажатия кнопки «Выход»). Логин и пароль не использовать. Предусмотрите страницу для просмотра файла-журнала сессий.

В сеттерах классов выбрасывать исключения при некорректны параметрах. Обработка исключений – на страницах с таблицей товаров, таблицей транспортных средств.

Формы обрабатывать в тех же файлах, где они определены, при некорректных значениях выбрасывать исключения.

**Задача 1.** Создать класс **Goods** (товар). В классе должны быть представлены поля: наименование товара, дата оформления (прихода), цена единицы товара, количество единиц товара, номер накладной, по которой товар поступил на склад.

Реализовать методы изменения цены единицы товара, изменения количества товара (увеличения и уменьшения), вычисления стоимости товара. Разработать форму добавления/редактирования товара. Использовать \_\_toString().

Реализовать массив товаров, добавление в массив, удаление из массива. Данные по товарам сохранять в файле в формате **CSV**. Также требуется выводить таблицу товаров, итоговую сумму хранимых товаров.

**Задача 2.** Разработайте иерархию: Интерфейс ТранспортноеСредство --> абстрактный класс ОбщественныйТранспорт --> класс Трамвай.

Данные по трамваю вводить в форме, трамвай добавляем в массив, массив трамваев сохранять на сервере, в CSV-файле. Отображение массива трамваев – в таблице. Предусмотрите возможность добавления, удаления, изменения данных о конкретном трамвае.

Некоторые рекомендуемые свойства трамвая: маршрут, пассажировместимость, фактическое количество пассажиров, текущая скорость. Некоторые рекомендуемые методы: начало движения, завершение движения, посадка пассажиров, высадка пассажиров.

Очевидно, что посадка и высадка пассажиров не выполняются в процессе движения трамвая.

Примените magic-методы \_\_get(), \_\_set(), \_\_toString() в классе Трамвай. По отдельным командам продемонстрируйте сериализацию и десериализацию трамвая, массива трамваев.

**Задача 3.** Обработка файла объектов в формате **CSV**. Объект – класс Планета (Солнечной системы) с закрытыми свойствами название, радиус, масса, количество спутников, *тип планеты (каменная, газовый гигант, ледяной гигант)*, расстояние до Солнца в а.е., фотография. По кликам на кнопки типа «submit» реализуйте обработки:

* Вывод данных из файла на страницу с упорядочиванием по расстоянию
* Вывод данных из файла на страницу с упорядочиванием по алфавиту
* Вывод данных из файла на страницу с упорядочиванием по массе
* Выборка планет по массе
* Выборка планет по типу
* Удаление сведений о планете

## Дополнительно

Запись занятия можно скачать [**по этой ссылке**](https://cloud.mail.ru/public/vhnY/K741ziKmS), материалы занятия – в этом же архиве.