# Теоретическая часть

* Понятие о трейтах в PHP
* Создание трейта, ключевое слово **trait**
* Подключение трейта к классу, ключевое слово **use**
* Разрешение конфликтов свойств и методов, ключевые слова **insteadof**, **as**
* Понятие о генераторах в PHP, ключевое слово **yield**
* Понятие о классе **PDO** (PHP Data Object) для доступа к базам данных из PHP
* Включение поддержки **PDO** для MySQL, SQLite в конфигурационном файле php.ini
* Подключение к базе данных при помощи **PDO**
* Реализация CRUD-операций при помощи класса **PDO**, методы query(), execute()
* Получение результатов в виде ассоциативного и обычного массивов
* Получение скалярного результата запроса
* Подготовленные запросы в PDO
* Передача неименованных параметров в запросы – символы «?» в запросе и передача массива параметров
* Передача именованных параметров в запросы – :имяПараметра и привязка параметра

# Практическая часть

Разработайте веб-приложение на **PHP** для решения задач с использованием трейтов, базы данных **SQLite3** и **PDO**.

**Задача 1.** Применение трейтов. Реализуйте обработку **прямоугольных** матриц по GET-запросу. Размеры матриц формируйте при помощи генератора случайных чисел, заполнение матриц – целыми случайными числами.

При помощи исключений обрабатывать ошибки времени исполнения.

* + **Вариант 12.** Найти номер первой из строк, содержащих хотя бы один положительный элемент (0 считаем положительным числом). Уплотнить заданную матрицу, удаляя из нее строки и столбцы, заполненные нулями.
  + **Вариант 15.** Определить номер первого из столбцов, содержащих хотя бы один нулевой элемент.Характеристикой строки целочисленной матрицы назовем сумму ее отрицательных четных элементов. Переставляя строки заданной матрицы, расположить их в соответствии с убыванием характеристик.

Решение по каждому варианту надо разместить в отдельном классе (Variant12, Variant15). Работу с матрицей (создание, вывод в таблицу, заполнение матрицы значениями), реализовать в трейте. Добавлять трейт к каждому из классов решения (Variant12, Variant15).

**Задача** **2.** Применение PDO. Разработайте базу данных SQLite3. Разработайте код для выполнения запросов с параметрами. Параметры запросов вводить из форм

|  |  |  |
| --- | --- | --- |
| *База данных****«Платный прием в поликлинике»*** | | |
| **Описание предметной области**  Платный прием пациентов (консультации специалистов) проводится врачами разных специальностей (хирург, терапевт, кардиолог, офтальмолог и т.д.). Несколько врачей могут иметь одну и ту же специальность. При оформлении приема должна быть сформирована квитанция об оплате приема, в которой указывается информация о пациенте, о враче, который консультирует пациента, о стоимости приема, о дате приема.  Пациент оплачивает за прием некоторую сумму, которая устанавливается персонально для каждого врача. За каждый прием врачу отчисляется фиксированный процент от стоимости приема. Процент отчисления от стоимости приема на зарплату врача также устанавливается персонально для каждого врача.  Размер начисляемой врачу заработной платы за каждый прием вычисляется по формуле: **Стоимость приема \* Процент отчисления от стоимости приема на зарплату врача**. Из этой суммы вычитается подоходный налог, составляющий 13% от суммы. | | |
| ***База данных должна включать как минимум таблицы ВРАЧИ, ПАЦИЕНТЫ, ПРИЕМ, содержащие следующую информацию:*** | | |
| Фамилия врача | | |
| Имя врача | | |
| Отчество врача | | |
| Специальность врача | | |
| Стоимость приема | | |
| Процент отчисления от стоимости приема на зарплату врача | | |
| Фамилия пациента | | |
| Имя пациента | | |
| Отчество пациента | | |
| Дата рождения пациента | | |
| Адрес пациента | | |
| Дата приема | | |
| **ЗАПРОСЫ** | | |
| **Номер запроса** | **Тип запроса** | **Какую задачу решает запрос** |
| 1 | Запрос с параметрами | Выбирает информацию о пациентах с фамилиями, начинающимися на заданную последовательность символов |
| 2 | Запрос с параметрами | Выбирает информацию о врачах, для которых значение в поле **Процент отчисления на зарплату**, больше заданного |
| 3 | Запрос с параметрами | Выбирает информацию о приемах за некоторый период |
| 4 | Запрос с параметрами | Выбирает из таблицы информацию о врачах с заданной **специальностью** |
| 5 | Запрос с вычисляемыми полями | Вычисляет размер заработной платы врача за каждый прием. Включает поля **Фамилия врача, Имя врача, Отчество врача, Специальность врача, Стоимость приема, Зарплата**. Сортировка по полю **Специальность врача** |
|  |  |  |
| 6 | Итоговый запрос | Выполняет группировку по полю **Дата приема**. Для каждой даты вычисляет максимальную стоимость приема |
| 7 | Итоговый запрос | Выполняет группировку по полю **Специальность**. Для каждой специальности вычисляет средний Процент отчисления на зарплату от стоимости приема |

# Дополнительно

Материалы занятия и задачник – в этом же архиве. Запись занятия можно скачать [**по этой** **ссылке**](https://cloud.mail.ru/public/JKGi/gEJxGWTNh).