

Язык программирования Java и современные Java-технологии

Краткая информация о курсе:

Продолжительность курса: 12 месяцев, 184 пары, 92 встречи.

Формат обучения: 2 раза в неделю по 2 пары

Чему научим:

Модуль 1: Язык программирования Java

По окончании модуля слушатель будет:

- Понимать фундаментальные принципы создания программ с использованием Java
- Уметь создавать, компилировать, и отлаживать проекты в IDE Eclipse / IDE IntelliJ idea
- Уметь проектировать и реализовывать различные алгоритмы
- Использовать механизмы условий и циклов
- Применять массивы для хранения данных
- Уметь использовать алгоритмы сортировки и поиска данных
- Разбираться в принципах ООП
- Уметь проектировать классы различной степени сложности
- Создавать иерархии классов для решения практических задач
- Использовать механизмы generics для построения шаблонных классов
- Уметь порождать и обрабатывать исключительные ситуации
- Выбирать и использовать классы JCF
- Сохранять и читать информацию из файлов
- Понимать механизмы многопоточности Java
- Понимать и уметь применять паттерны проектирования

Модуль 2: Front-end Technologies

По окончании модуля HTML слушатель будет:

- Понимать, где и как используются основные возможности HTML5/CSS3
- Использовать принципы блочной верстки создания и корректировки HTML страниц
- Создавать адаптивную верстку
- Знать основы HTML5/CSS3

По окончании модуля JS слушатель будет:

- Знаком с ООП и основными его принципами
- Уметь изменять содержимое страницы по средства JS
- Применять основные возможности DOM
- Отличать DOM и BOM
- Разбираться в понятиях событие, обработчик события.
- Создавать функции-обработчики различных событий
- Создавать и валидировать формы
- Уметь сериализовать и парсить данные используя JSON
- Знаком с основами асинхронных запросов при помощи Ajax

Модуль 3: Theory of Database

По окончании курса слушатель будет:

- Разбираться в языке структурированных запросов SQL
- Уметь создавать многотабличные запросы
- Понимать принципы работы подзапросов и функций агрегирования
- Уметь производить нормализацию баз данных
- Использовать хранимые процедуры, триггеры, виды, пользовательские функции

Модуль 4: Java Technologies for Web Applications

По окончании курса слушатель будет:

- Понимать фундаментальные принципы создания серверных решений с использованием Java
- Уметь создавать, компилировать, и отлаживать веб-приложения
- Уметь взаимодействовать с источниками данных
- Использовать сетевые механизмы
- Уметь создавать сервлеты
- Понимать и применять паттерн MVC
- Уметь создавать JSP решения
- Применять механизмы cookies и сессий
- Понимать и применять фреймворки: Spring & Hibernate

Программа обучения:

МОДУЛЬ 1 - Java

Введение в язык программирования “Java” (4 пары)

1. Вступление
 1. История и этапы развития языка “Java”
 2. Сравнительный анализ языка “Java” с другими языками программирования
 3. Что такое виртуальная машина?
 4. Что такое байт-код?
2. Алгоритм
 1. Понятие алгоритма
 2. Примеры использования алгоритмов в реальной жизни
 3. Типы алгоритмов. Линейный, разветвлённый, циклический
3. Понятие блок-схемы
 1. Базовые обозначения в блок-схемах
 2. Блок начала алгоритма
 3. Блок завершения алгоритма
 4. Блок ввода данных
 5. Блок вывода данных
 6. Блок вычислений
 7. Простейшие примеры использования блок-схем
4. Программная среда “Eclipse”
 1. Инсталляция
 2. Основы работы с IDE Eclipse
 3. Создание проекта
 4. Добавление файла к проекту
 5. Обзор альтернативных средств разработки
 6. Запуск простейшего приложения

Переменные, типы данных, операторы (6 пар)

1. Типы данных
 1. Понятие типа данных. Размер, диапазон значений
 2. Целые типы данных
 3. Типы данных для работы с дробными числами
 4. Символьный тип данных
 5. Логический тип данных
 6. Перечислимый тип данных
2. Переменные
 1. Необходимость использования переменных
 2. Идентификаторы
 3. Ключевые слова
 4. Синтаксис объявления переменных
3. Константы и литералы
 1. Необходимость применения
 2. Синтаксис объявления
4. Операторы
 1. Понятие оператор
 2. Типы операторов
 1. Арифметические операторы
 2. Логические операторы
 3. Операторы ветвлений
 4. Унарные операторы
 5. Бинарные операторы

6. Тернарный оператор
3. Оператор присваивания
4. Арифметические операторы
 1. Оператор сложения
 2. Оператор вычитания
 3. Оператор умножения
 4. Оператор деления
 5. Оператор деления по модулю
 6. Инкремент. Постфиксная и префиксная форма
 7. Декремент. Постфиксная и префиксная форма
 8. Сокращенные формы
5. Примеры построения программ с использованием блок-схем

Логические операторы, операторы ветвлений, побитовые операторы (6 пар)

1. Преобразование типов данных
 1. Необходимость использования
 2. Неявное преобразование типов
 3. Явное преобразование типов
2. Логические операторы
 1. Знакомство с логическими операциями
 2. Таблица результатов применения логических операций
 3. «Логическое отрицание». Оператор !
 4. «Логическое И». Оператор &&
 5. «Логическое ИЛИ». Оператор ||
3. Таблица приоритетов операторов
4. Конструкции логического выбора. Операторы ветвлений
 1. Оператор ветвления if
 2. Оператор ветвления if – else
 3. Лестница if - else if
 4. Обозначение условий в блок-схемах. Блок условия
 5. Обозначение объединения ветвей в блок-схемах
 6. Примеры построения программ с использованием операторов ветвлений на языке блок-схем
 7. Понятие составного оператора
 8. Тернарный оператор
 9. Оператор множественного выбора – switch
5. Побитовые операторы
 1. Системы исчисления двоичная, восьмеричная, шестнадцатеричная
 2. Цели и задачи битовых операций
 3. Битовое "И"
 4. Битовое "ИЛИ"
 5. Битовое "ИСКЛЮЧАЮЩЕЕ ИЛИ"
 6. Битовое отрицание
 7. Битовые сдвиги

Циклы (8 пар)

1. Циклы
 1. Необходимость использования циклов. Примеры использования
 2. Цикл while
 3. Цикл for
 4. Цикл do-while

5. Обозначение циклов в блок-схемах. Блок цикла
6. Операторы break и continue
7. Примеры построения программ с использованием циклов на языке блок-схем
8. Вложенные циклы. Примеры использования
2. Работа с интегрированным отладчиком в Eclipse
 1. Что такое отладчик. Цели и задачи отладчика
 2. Запуск программы по шагам
 3. Окна для работы с отладчиком. Окна переменных, локальных переменных, памяти
 4. Исполнение одного шага
 5. Установка точки останова (breakpoint)

Строки, массивы одномерные, многомерные (8 пар)

1. Работа со строками
2. Массивы
 1. Что такое массивы?
 2. Необходимость использования массивов
 3. Синтаксис объявления одномерного массива
 4. Схема размещения массивов в памяти.
 5. Индексация элементов массива
 6. Примеры использования массивов на языке блок-схем
3. Алгоритмы суммирования
4. Алгоритмы поиска
 1. Линейный
 2. Бинарный
5. Алгоритмы сортировки
 1. Пузырьковая сортировка
 2. Сортировка выбором
 3. Сортировка вставками
 4. Другие алгоритмы сортировки (быстрая сортировка и т.д.)
6. Понятие сложности алгоритма
7. Многомерные массивы
 1. Многомерные массивы. Цели и задачи их использования
 2. Двумерные массивы, как частный случай многомерных
 3. Синтаксис объявления многомерного массива
 4. Примеры использования многомерных массивов

Методы (на примере статических методов) (6 пар)

1. Методы
 1. Что такое метод?
 2. Необходимость использования методов
 3. Синтаксис объявления методов
 4. Использование ключевого слова void при работе с методами
 5. Вызов метода
 6. Аргументы
 7. Возврат значения из метода (return)
2. Область видимости
 1. Понятие области видимости
 2. Примеры использования областей видимости
3. Рекурсия

Объектно-ориентированное программирование (20 пар)

1. Введение в объектно-ориентированное программирование
 1. Инкапсуляция
 2. Полиморфизм
 3. Наследование
2. Понятие класса
3. Понятие объекта
4. Понятие члена класса, поля класса, метода класса
5. Спецификаторы доступа
6. Конструкторы объекта
 1. Что такое конструктор?
 2. Цели и задачи конструктора
 3. Примеры создания конструкторов
7. Ключевое слово `this`
8. Перегрузка методов и конструкторов
9. Статические методы классов
 1. Что такое статический метод класса?
 2. Отличие статического и обычного метода класса
 3. Примеры использования статических методов
10. Передача объектов в метод
11. Область видимости в методах классов
12. Наследование
 1. Спецификаторы доступа при наследовании
 2. Ключевое слово `super`
 3. Порядок вызова конструкторов
 4. Переопределение методов
 5. Динамическая диспетчеризация методов
 6. Абстрактный класс
13. Понятие интерфейса
 1. Что такое интерфейс?
 2. Реализация интерфейса
 3. Методы интерфейса по умолчанию
 4. Использование реализации интерфейса через ссылки
 5. Вложенные интерфейсы
 6. Переменные и интерфейсы
14. Вложенные классы
15. Ключевое слово `final`. Использование его в классах, методах, переменных
16. Сборка мусора
 1. Что такое сборка мусора?
 2. Принцип работы сборщика мусора
 3. Что такое финализатор?
 4. Принципы создания финализатора
17. Пакеты
18. Шаблоны (Generics)
 1. Что такое шаблоны?
 2. Цели и задачи шаблонов
 3. Шаблонные классы
 4. Шаблонные методы
 5. Шаблонные конструкторы
 6. Шаблонные интерфейсы
 7. Шаблоны и наследование

Исключения (4 пары)

1. Что такое исключительная ситуация?

2. Принципы обработки исключительных ситуаций
3. Понятие checked и unchecked исключений
 1. Что такое checked и unchecked исключения?
 2. Отличия и принципы использования
4. Ключевое слово try
5. Ключевое слово catch
6. Ключевое слово throw
7. Ключевое слово finally
8. Подробности использования исключительных ситуаций
9. Раскрутка стека вызовов

JavaCollectionFramework (8 пар)

1. Классы-обертки
2. Введение в JCF:
 1. Причины создания
 2. Обзор
3. Интерфейсы JCF:
 1. Collection
 2. Comparator
 3. Enumeration
 4. EventListener
 5. Iterator
 6. List
 7. ListIterator
 8. Map
 9. Map.Entry
 10. Observer
 11. RandomAccess
 12. Set
 13. SortedMap
 14. SortedSet
4. Классы JCF:
 1. AbstractCollection
 2. AbstractList
 3. AbstractMap
 4. AbstractSequentialList
 5. AbstractSet
 6. ArrayList
 7. Arrays
 8. BitSet
 9. Collections
 10. Dictionary
 11. HashMap
 12. HashSet
 13. Hashtable
 14. IdentityHashMap
 15. LinkedHashMap
 16. LinkedHashSet
 17. LinkedList
 18. Stack
 19. TreeMap
 20. TreeSet
 21. Vector

Работа с файлами (4 пары)

1. Знакомство с пакетом java.io
2. Потоки ввода/вывода
 1. Потоки ввода/вывода
 2. Фильтрованные потоки
 3. Канальные потоки
 4. Буферизированные потоки
 5. Файловые потоки
 6. Потоки для работы с файлами
 7. Потоки, размещаемые в оперативной памяти
3. Сериализация объектов
 1. Понятие сериализации
 2. Граф сериализации
 3. Использование сериализации

Многопоточность (4 пары)

1. Многопоточность в Java
 - 1.1. что такое многопоточность?
 - 1.2 Класс Thread и интерфейс Runnable
 - 1.3 Приоритеты и состояния потоков
2. Синхронизация потоков
 - 2.1 synchronized
 - 2.2 wait, notify, notifyAll
 - 2.3 Проблемы, возникающие при синхронизации потоков и их решение
3. Concurrency Framework
 - 3.1 - Executors, Executor Service и ThreadPool
 - 3.2 Concurrent Collections
 - 3.3 - Atomic типы данных и операции
 - 3.4 Immutable объекты, назначение и применение
4. Практические примеры

Паттерны проектирования (12 пар)

1. Паттерны создания объектов
 1. Singleton
 2. Factory Method
 3. Prototype
 4. Builder
 5. Abstract Factory
2. Структурные паттерны
 1. Adapter
 2. Facade
 3. Decorator
3. Поведенческие паттерны проектирования
 1. Strategy
 2. Iterator
4. Антипаттерны проектирования

Экзамен (2 пары)

МОДУЛЬ2: Front-end Technologies

Разработка веб-страниц на языке разметки XHTML с использованием каскадных таблиц стилей CSS

Модуль 1. Введение в Web-технологии. Структура HTML.

- Введение в предмет.
- Введение в язык разметки. Язык разметки гипертекста HTML.
 - Развитие HTML, версии. Текущие используемые версии: HTML и XHTML.
 - Вопросы межбраузерной совместимости. Война браузеров.
 - W3C.
- Теги – основной элемент структуры HTML. Правила записи тегов и их атрибутов в стандарте XHTML на примере тегов ``, `<i>`, `<u>`, ``, `<sup>`, `<sub>`, `
`. Синтаксические отличия HTML4 от XHTML.
- Основные ошибки в записях тегов.
 - Спецификации `<!DOCTYPE HTML>`.
 - Валидация документа при помощи FireFox – дополнение HTML Validator.
 - Понятие well-formed.
 - Прародители HTML4/5 и XHTML: SGML и XML.
- Основная структура XHTML документа. Основные элементы и их назначение.
- Кодировки страницы и теги `<meta>`:
 - Применение тега `<meta>` – задание информации о странице (expires, refresh, autor, copyright, keywords, description).
 - Задание кодировки страницы при помощи тега `<meta>`.
 - Символьные подстановки и кодировки.

Необходимое время – 2 пары

Модуль 2. Форматирование текста при помощи HTML. Списки

- Классификация тегов: линейные и блочные.
 - Линейные: ``, `<i>`, `<u>`, ``.
 - Блочные: `<p>`, `<h1>..<h6>`.
- Модель форматирования текста: заголовки и абзацы. Элементы `<p>`, `<h1>..<h6>`. Выравнивание текста в блочных элементах: атрибут align.
- Классификация тегов: логическое и физическое форматирования.
 - Теги физического форматирования: ``, `<i>`.
 - Теги логического форматирования: ``, ``. Их отличие.
 - Краткий обзор основных тегов логического форматирования: `<abbr>`, `<acronym>`, `<cite>`, `<code>`, ``, `<dfn>`, `<ins>`.
- Цвета в Web.
 - Web-палитра.
 - Photoshop/GIMP – дополнительные инструменты верстальщика. Рассмотрение основных возможностей: открытие изображений, подбор цветов при помощи палитры Photoshop|GIMP, инструмент пипетка.
 - Использование выбранного цвета в атрибуте color тега ``.
- Практика: создание простейшей web-страницы.
- Создание списков
 - Неупорядоченные списки: элементы ``, ``.
 - Упорядоченные списки: элементы ``, ``.

- о Атрибуты type, value, start.
- Создание вложенных списков.
- Списки определений: элементы <dl>, <dd>, <dt>.
- Практика: создание списков.

Необходимое время – 2 пары

Модуль 3. Графика в web-дизайне. Оптимизация графики. Гиперссылки. Принципы навигации web-сайта.

- Форматы графических файлов в Web.
- Тег и его атрибуты (src, alt, width, height, border).
 - о Свойство border – аналог атрибута border.
 - о Задание свойств margin, padding, border для изображения.
 - о Выравнивание изображений на странице при помощи атрибута align. Аналог атрибута align – свойство float.
- Общие сведения о гиперссылках.
 - о Тег <a> и его атрибуты (href, target).
 - о Эргономика, удобство навигации.
- Абсолютная и относительная адресация.
 - о Организация внешних ссылок.
 - о Организация внутренних ссылок с помощью элемента <a>. Атрибуты id и name.
 - о Организация "смешанного" перехода (на указанный элемент во внешнем HTML-документе).
 - о Графические ссылки. Отмена границ у ссылок.
- Создание меню при помощи структуры списков (,), его форматирование.
- Практика: работа по разработке галереи изображений.

Необходимое время – 2 пары

Модуль 4. Таблицы.

- Создание простейшей таблицы. Теги <table>, <tr> и <td>.
 - о Атрибуты border, cellpadding, cellspacing. Их возможные аналоги CSS: border, padding.
 - о Указание ширины и высоты ячейки: атрибуты width, height. Правила задания ширины и высоты. Аналоги CSS: свойства width, height.
 - о Выравнивание данных в таблице: атрибуты align и valign. Аналоги CSS: свойства text-align, vertical-align.
 - о Управление цветом фона и цветом рамок таблицы (отдельной строки, отдельной ячейки).
 - о Использование изображений в качестве фона таблицы (отдельной строки, отдельной ячейки).
- Объединение ячеек: атрибуты colspan, rowspan.
- Теги логического структурирования таблиц: <thead>, <tbody>, <tfoot>. Теги логического группирования столбцов: <colgroup>, <col>.
- Управление рамками таблицы: атрибуты frame, rules.
- Практика: создание сложных таблиц.
- Основы табличной верстки. Пример табличной верстки: ее минусы.

Необходимое время – 2 пары

Модуль 5. Формы

- Введение в формы.
- Управляющие элементы форм.
 - Кнопки (отправки, сброса, пр.).
 - Флажки.
 - Кнопки с зависимой фиксацией (радиокнопки).
 - Всплывающие списки.
 - Текстовый ввод.
 - Выбор файлов.
 - Скрытые управляющие элементы.
- Создание форм при помощи HTML:
 - Элемент `<form>`.
 - Элемент `<input>`.
 - Элемент `<button>`.
 - Элементы `<select>`, `<optgroup>` и `<option>`.
 - Элемент `<textarea>`.
 - Метки `<label>`.
 - Структура форм: `<fieldset>` и `<legend>`.

Необходимое время – 2 пары

Модуль 6. Форматирование при помощи CSS.

- CSS – каскадные таблицы стилей.
 - Введение. Обзор версий. Назначение: HTML служит для задания структуры, CSS – для форматирования.
 - Встраивание CSS в HTML при помощи атрибута `style`. Правила записи CSS свойств.
- Теги без форматирования `<div>` – блочный, `` – линейный.
- Аналогия HTML и CSS на примере линейных и блочных тегов:
 - Тег `` – свойства `color`, `font-size`, `font-family`.
 - Тег `` – свойства `font-weight`.
 - Тег `<i>` – свойства `font-style`.
 - Тег `<u>` – свойства `text-decoration`.
 - Тег `<sup>`, `<sub>` – свойства `vertical-align`.
 - Атрибут `align` – свойства `text-align`.
 - Сокращенная запись свойства `font`.
 - Дополнительные свойства CSS для форматирования текста: `letter-spacing`, `line-height`, `text-indent`, `text-transform`, `white-space`, `word-spacing`.
- Использование атрибутов `class` и `id` для задания стилей.
 - Создание стилей для тегов, классов, идентификаторов внутри тега `<style>`. Понятие селекторов. Правило записи селекторов: селектор тегов, селектор классов, селектор идентификаторов, универсальный селектор `*`.
 - Приоритет использования стилей (`tag / class / id / style`). Повышение приоритета правилом `!important`.
 - Наследуемость стилей. Стандартные значения свойств.
 - Отслеживание стилей при помощи средства разработки `firebug` (дополнение для Firefox)
- Использование внешних CSS файлов стилей.
 - Подключение CSS файлов при помощи тега `<link>` и инструкции `@import`.
 - CSS файлы и кэш браузера.
- Практика: форматирование текста при помощи CSS.

Необходимое время – 2 пары

Модуль 7. Позиционирование. Верстка web-страниц блоками.

- Свойство position.
 - Рассмотрение позиционирования: relative и absolute.
 - Свойства top, left, bottom, right.
- Свойства visibility, overflow.
- Практика.
- Основы верстки блоками. Правила верстки.
 - Вложение блоков.
 - Задание ширины и высоты блокам при помощи свойства width и height.
 - Обтекание блоков. Отмена обтекания блоков. Свойства float и clear.
 - Правила задания отступов и полей.
 - Задание минимальной высоты и ширины блока: свойства min-height, min-width. Задание этих свойств в браузере IE6.
 - Выравнивание внутри блоков (margin, text-align, line-height, position). Кроссбраузерность выравниваний.
- Рассмотрение простейших структур страниц
- Управление отступами и полями.
 - Свойство margin и его потомки margin-left, margin-top, margin-right, margin-bottom.
 - Свойство padding и его потомки padding-left, padding-top, padding-right, padding-bottom.
 - Отличие padding от margin и их назначения.
 - Отмена отступов по умолчанию у некоторых тегов: <body>, <h1>..<h6>, <p>.. Структура фиксированного размера.
- Резиновая структура. Блоки с отрицательными margin.

Необходимое время – 2 пары

Модуль 8. Изображения в CSS. Списки. Псевдоклассы. Псевдоэлементы

- Форматирование списков при помощи CSS.
 - Свойства list-style-type, list-style-image, list-style-position.
 - Сокращенная запись свойства list-style.
 - Оформление многоуровневых списков. Вложенные селекторы.
- Свойство display. Преобразование ссылки в блочный элемент.
- Свойство background.
- Псевдоэлементы
 - Псевдоэлементы: before, after.
- Псевдоклассы.
 - Псевдоклассы ссылок: active, hover, link, visited.
 - Псевдоклассы для обычных элементов: first-child, first-line, first-letter, last-child, first-child
- CSS свойство cursor.

Модуль 9. Возможности CSS3. Новые возможности HTML 5

- Структура HTML5 документа.
 - Новые теги задания структуры: <header>, <nav>, <section>, <article>, <aside>, <footer>. Доступность новых тегов в современных браузерах. Отображение новых тегов в устаревших браузерах.
- HTML5 – конкурент Flash.

- о Вставка видео на странице посредством тега <video>.
 - о Вставка аудио на странице посредством тега <audio>.
 - о Создание изображений и анимации посредством тега <canvas>.
 - о Использование SVG формата.
- Новые элементы форм.
- Новые свойства: CSS3.
 - о Работа с фоном: создание градиентов, изменение размеров фона – свойства background и background-size.
 - о Работа с границами: скругленные края у блоков – свойства border-radius.
 - о Задание полупрозрачности элементам страниц – свойство opacity.
 - о Полная поддержка селекторов CSS 2.1.

Необходимое время – 2 пары

Модуль 10. Экзамен.

- Создание web-сайта с последующим размещением в Internet – (Основные требования: блочная верстка, валидный код).

Необходимое время – 2 пары

Разработка клиентских сценариев с использованием JavaScript

Модуль 1 (2 пары)

Введение в JavaScript

1. Сценарии, выполняемые на стороне клиента
2. Что такое JavaScript?
3. История создания JavaScript
4. Различия между JavaScript и Java, JScript, ECMAScript
5. Версии JavaScript
6. Понятие Document Object Model
7. Понятие Browser Object Model
8. Внедрение в HTML документы. Редакторы кода JavaScript
9. Тег <noscript>
10. Основы синтаксиса
 - a. Регистрозависимость
 - b. Комментарии
 - c. Ключевые и зарезервированные слова
11. Переменные. Правила именования переменных
12. Типы данных
13. Операторы
 - a. Арифметические операторы
 - b. Операторы отношений
 - c. Логические операторы
 - d. Оператор присваивания
 - e. Битовые операторы
 - f. Приоритет операторов
 - g. Оператор typeof
14. Ввод/вывод данных. Диалоговые окна
 1. Условия

- a. Что такое условие?
 - b. if
 - c. if else
 - d. Тернарный оператор ?:
 - e. switch
2. Циклы
- a. Что такое цикл?
 - b. while
 - c. do while
 - d. for
 - e. break
 - f. continue
 - g. Понятие метки
3. Что такое функция?
- a. Синтаксис объявления функции
 - b. Параметры функции
 - c. Возвращаемое значение функции. Ключевое слово return
4. Объект arguments
- a. Цель и задачи объекта
 - b. Свойство length
5. Область видимости переменной. Ключевое this
6. Рекурсия

Модуль 2 (2 пары)

Объект. Массивы. Объект Array. Строки. Объект String. Объект Date. Объект Math. Введение в объектно-ориентированное программирование

1. Объекты
- a. Что такое объект?
 - b. Введение в объектный тип данных
 - c. Объект Object
 - d. Ключевое слово new
 - e. Понятие свойства
 - f. Добавление свойств. Синтаксис добавления свойств.
 - g. Синтаксис обращения к свойствам
2. Массивы
- a. Что такое массив?
 - b. Объект Array
 - c. Создание массива
 - d. Обращение к элементам массива
 - e. Свойства и методы Array
3. Строки
- a. Объект String
 - b. Свойства и методы String
4. Задержки и интервалы. Периодический вызов функций
5. Объект Date. Обработка даты и времени
6. Объект Math. Свойства и методы. Случайные числа
7. Что такое ООП?
8. Три фундаментальных принципа ООП
- a. Инкапсуляция
 - b. Наследование

- с. Полиморфизм
- 9. Понятие класса и объекта в терминах JavaScript
- 10. Свойства
- 11. Методы
- 12. Свойства-аксессоры
 - а. get – свойства (геттеры)
 - б. set – свойства (сеттеры)
- 13. Конструктор
- 14. Понятие prototype
 - а. Что такое prototype
 - б. Цели и задачи prototype
- 15. Наследование

Модуль 3 (2 пары)

Обработка событий

1. Что такое событие?
2. Что такое обработчик события?
3. Обработка событий в сценариях
4. Управление стилями элементов web-страницы
5. Объект event и его свойства
6. Обработчики событий по умолчанию (стандартные обработчики), запрет вызова стандартного обработчика
7. Объект Image. Управление рисунками и ролловерами

Модуль 4 (4 пары)

Browser Object Model. Document Object Model.

1. Что такое Browser Object Model?
2. Объекты Browser Object Model
 - а. Объект Window. Открытие, перемещение и изменение размера окон
 - б. Объект Navigator. Управление браузером
 - с. Объект Screen. Свойства экрана
 - д. Объекты Location и History. Перемещение по страницам.
3. Что такое Document Object Model?
4. Отличия DOM от BOM
5. Представление HTML-документа в виде дерева
6. Объекты модели DOM. Иерархия узлов
7. Свойства и методы модели DOM. Модель событий DOM
8. Изменение дерева DOM
9. Знакомство с объектами Document и Link
10. Управление выделением и текстовым диапазоном: объекты Selection и TextRange
11. Особенности DOM в HTML5

Модуль 5 (2 пары)

Формы

1. Применение форм. Размещение элементов формы в HTML.
2. Коллекция Forms. Создание и программирование элементов форм:
 - а. Кнопки: элементы Button, Submit, Reset

- b. Текстовые поля: элементы Text, Password, File Upload, Textarea
- c. Скрытое поле формы: общее понятие об элементе Hidden
- d. Флажок: элемент Checkbox
- e. Переключатель: элемент Radio
- f. Список: элементы Select, Option

Модуль 6 (2 пары)

Проверка достоверности форм. Использование Cookie

- 1. Объект RegExp. Правила записи регулярных выражений
- 2. Методы объектов String и RegExp для работы с регулярными выражениями
- 3. Проверка достоверности данных формы
- 4. Что такое cookie?
- 5. Преимущества и недостатки cookie
- 6. Создание, использование и удаление cookie

Модуль 7 (2 пары)

JSON

- 1. Что такое JSON?
- 2. Цели и задачи JSON
- 3. Синтаксис JSON
 - a. Переменные
 - b. Объекты
 - c. Массивы
- 4. Объект JSON
 - a. Что такое сериализация?
 - b. Что такое парсинг?
 - c. Методы stringify и parse
- 5. Настройка пользовательской сериализации в JSON. Метод toJSON

Модуль 8 (2 пары)

Ajax

- 1. Синхронные и асинхронные запросы
- 2. Что такое Ajax?
- 3. Объект XMLHttpRequest
 - a. Создание через ActiveX объект
 - b. Создание через объект XMLHttpRequest
- 4. Методы и свойства XMLHttpRequest
- 5. Понятие HTTP заголовка
- 6. Использование метода GET. URL кодирование
- 7. Использование метода POST

МОДУЛЬ 3: Theory of Database

Введение в теорию баз данных (2 пары)

- 1. Введение в теорию баз данных.

- a. История и этапы развития.
 - b. Понятия база данных и система управления базами данных.
 - c. Сравнение существующих моделей баз данных.
 - Файловая модель.
 - Сетевая модель.
 - Иерархическая модель.
 - Реляционная модель.
 - Объектно-ориентированная модель.
 - d. Понятие реляционной модели баз данных.
 - e. Двенадцать правил Кодда.
2. История СУБД MySQL
3. Архитектура СУБД MySQL
4. Версии MySQL Oracle
5. Утилиты:
- a. Admin Utilities (Clone, Copy, Compare, Diff, Export, Import, User Management)
 - b. Replication Utilities (Setup, Configuration, Verification)
 - c. General Utilities (Disk Usage, Redundant Indexes, Manage Meta & Audit Data)
6. Демонстрация: Инсталляция СУБД MySQL.
7. Архитектура БД под управлением MySQL. Сравнение с другими СУБД
- a. Создание базы данных с помощью Database Configuration Assistant
 - b. Создание базы данных с помощью файла конфигурации
8. Демонстрация:
- a. Создание базы данных и управления базами данных с помощью Database Configuration Assistant.
 - b. Создание базы данных с помощью файла конфигурации
9. Практическая работа: Создание базы данных с помощью файла конфигурации
10. Основы взаимодействия с MySQL
- a. Объекты MySQL
 - b. Таблицы.
 - Первичный ключ.
 - Значение по умолчанию.
 - Уникальность.
 - c. Типы данных.
 - Целочисленные типы.
 - Типы данных для хранения текста.
 - Вещественные типы данных.
 - Типы для хранения даты и времени.
 - Типы данных с фиксированной точкой.
 - Другие типы данных
 - d. Понятие индекса.
 - Цели и задачи индексов.
 - Внутреннее устройство индексов.
11. Запросы.
12. Введение в язык структурированных запросов SQL.
- a. Язык SQL. Стандарты языка SQL.
 - b. Понятия DDL,DML,DCL.

Запросы SELECT, INSERT, UPDATE, DELETE. (2 пары)

1. Оператор SELECT.
 - a. Предложение SELECT.
 - b. Предложение FROM.
 - c. Предложение WHERE.
 - d. Предложение ORDER BY.
2. Ключевые слова IN, BETWEEN, LIKE.
3. Оператор INSERT.
4. Оператор UPDATE.
5. Оператор DELETE.

Необходимое время – 2 пары.

Многотабличные базы данных. (2 пары)

1. Аномалии взаимодействия с однотабличной базой данных.
 - a. Аномалии обновления.
 - b. Аномалии вставки.
 - c. Аномалии обновления.
 - d. Аномалии удаления.
2. Принципы создания многотабличной базы данных.
 - a. Причины создания многотабличной базы данных.
 - b. Внешний ключ.
 - c. Связи. Типы связей.
 - d. Целостность данных.
 - e. Нормализация.
 - Необходимость нормализации.
 - Понятие нормальной формы.
 - Первая нормальная форма.
 - Вторая нормальная форма.
 - Третья нормальная форма.
 - Нормальная Форма Бойса-Кодда.
3. Многотабличные запросы.
 - a. Принципы создания многотабличного запроса.
 - b. Декартовое произведение.

Функции агрегирования. (2 пары)

1. Функции агрегирования
 - a. Функция COUNT.
 - b. Функция AVG.
 - c. Функция SUM.
 - d. Функция MIN.
 - e. Функция MAX.
2. Понятие группировки. Ключевое слово GROUP BY.
3. Ключевое слово HAVING. Сравнительный анализ HAVING и WHERE.
4. Подзапросы
 - a. Необходимость создания и использования подзапросов.
 - b. Сравнение подзапросов и многотабличных запросов.
 - c. Принцип работы подзапросов.

Объединения. (2 пары)

1. Операторы для использования в подзапросах.
 - a. Оператор EXISTS.
 - b. Операторы ANY/SOME.
 - c. Оператор ALL.
2. Объединение результатов запроса.
 - a. Принципы объединения.
 - b. Ключевое слово UNION.
 - c. Ключевое слово UNION ALL.

Объединения JOIN. (2 пары)

- d. Понятие inner join.
- e. Понятие left join.
- f. Понятие right join.
- g. Понятие full join.

Представления, хранимые процедуры, триггеры (4 пары)

1. Понятие транзакции. Использование транзакций.
2. Представления.
 - a. Создание представлений
 - b. Модификация представлений
 - c. Удаление представлений
 - d. Изменения данных через представления
3. Хранимые процедуры.
4. Триггеры.

Экзамен (2 пары)

МОДУЛЬ 4: Java Technologies for Web Applications

Введение в разработку серверных решений с использованием Java (6 пар)

1. Введение в серверное программирование
 1. Что такое серверное решение?
 2. Что такое веб-приложение?
 3. Чем отличается клиентская и серверная часть приложения?
 4. Какие механизмы предоставляет Java для создания веб-приложений?
 5. Какие утилиты полезны для создания веб-приложений на Java?
2. Краткий обзор полезных утилит и библиотек
 1. Что такое Maven?
 2. Что такое TomCat?
 3. Что такое JBoss?
 4. Что такое Spring?
 5. Что такое Hibernate?
3. Понятие сервлета
 1. Что такое сервлет?
 2. Цели и задачи сервлета
 3. Каркас сервлета
 4. Базовые интерфейсы сервлета
 5. Базовые классы сервлета
 6. Пример создания простого сервлета
 7. Настройка сервлета

8. Взаимодействие сервлета и клиента (http request / response)
9. Примеры создания сложных сервлетов

Взаимодействие с источниками данных (4 пары)

1. Источники данных
 1. Что такое источник данных?
 2. Какие бывают источники данных?
 3. База данных, как источник данных
2. JDBC
 1. Что такое JDBC?
 2. История возникновения JDBC
 3. Версии JDBC
 4. Использование JDBC для доступа к различным СУБД
3. Работа с JDBC
 1. Соединение с СУБД
 2. Получение данных из базы данных
 3. Сохранение данных в базу данных
 4. Обновление данных в базе данных
 5. Примеры использования JDBC в сервлетах

JavaServer Pages (8 пар)

1. Что такое JSP?
2. Цели и задачи JSP
3. История возникновения JSP
4. Понятие директивы
5. Обработка ошибок в JSP
6. Model View Controller
 1. Что такое Model View Controller?
 2. Цели и задачи Model View Controller
 3. Примеры создания серверных решений с помощью MVC
7. Expression Language в JSP
 1. Что такое Expression Language?
 2. Цели и задачи Expression Language
 3. Примеры использования
8. JavaBean
 1. Что такое JavaBean
 2. Цели и задачи JavaBean
 3. Примеры использования

Tags в JSP (4 пары)

1. Java Standard Tag Library
 1. Что такое Java Standard Tag Library?
 2. Цели и задачи Java Standard Tag Library
 3. Понятие Tag
2. Различные виды Tags
 1. Core Tags
 2. Formatting Tags
 3. SQL Tags
 4. XML Tags
 5. JSTL functions
3. Использование Conditional Tags

4. Использование Iteration Tags
5. Примеры использования других Tags
6. Что такое Custom Tags?
7. Что такое Tag Files?
8. Что JSP Fragment?
9. Примеры использования

Cookies, session и другое в JSP (4 пары)

1. Cookies
 1. Что такое cookies?
 2. Цели и задачи cookies
 3. Cookies и JSP
 4. Примеры использования
2. Сессии
 1. Что такое сессии?
 2. Цели и задачи сессий
 3. Сессии и JSP
 4. Примеры использования
3. Работа с файлами в JSP
4. Почтовые возможности JSP

Введение в Spring (8 пар)

1. Что такое Spring?
2. Цели и задачи Spring
3. История возникновения
4. Архитектура Spring
5. Spring MVC
6. Архитектура Spring MVC
7. Примеры использования

Введение в Hibernate (4 пары)

1. Что такое Hibernate?
2. Цели и задачи Hibernate
3. История возникновения
4. Архитектура Hibernate
5. Примеры использования

Экзамен (2 пары)