

Министерство науки и высшего образования РФ
ФГБОУ ВО «Ульяновский государственный университет»
Факультет математики, информационных и авиационных технологий

Кафедра телекоммуникационных технологий и сетей

Булаев Алексей Александрович

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

для семинарских (практических) занятий, лабораторного практикума и самостоятельной работы

по дисциплине

«Быстрая разработка приложений»

для студентов направлений

11.04.02 «Инфокоммуникационные технологии и системы» (магистратура)



УЛЬЯНОВСК
2022

Методические рекомендации для семинарских (практических) занятий и самостоятельной работы по дисциплине «Быстрая разработка приложений» / составитель: А.А. Булаев - Ульяновск: УлГУ, 2022 – 16 с.

Настоящие методические рекомендации предназначены для студентов направлений обучения 11.04.02 «Инфокоммуникационные технологии и системы» (магистратура). В работе приведены литература по дисциплине, темы дисциплины и вопросы в рамках каждой темы, рекомендации по изучению теоретического материала, контрольные вопросы для самоконтроля, задания для самостоятельной работы, задачи и упражнения для самостоятельной подготовки к семинарам или полностью самостоятельного освоения практических навыков, рекомендации по их выполнению.

Студентам всех форм обучения следует использовать данные методические рекомендации при подготовке к семинарам, самостоятельной подготовке, а также промежуточной аттестации по дисциплине «Быстрая разработка приложений».

Рекомендованы к введению в образовательный процесс

Учёным советом факультета математики, информационных и авиационных технологий
УлГУ

протокол № 3/19 от «19» апреля 2022 г.

Оглавление

| | |
|--|----|
| ОБЩИЕ ВОПРОСЫ..... | 4 |
| РЕКОМЕНДАЦИИ ПО ОТДЕЛЬНЫМ ТЕМАМ ДИСЦИПЛИНЫ..... | 5 |
| Тема 1. Основы быстрой разработки приложений..... | 5 |
| Тема 2. Основы разработки low-code приложений..... | 5 |
| Тема 3. Разработка мобильных приложений с использованием low-code..... | 5 |
| Тема 4. Разработка low-code приложений для Web..... | 6 |
| Тема 5. Разработка графических элементов приложений..... | 6 |
| Тема 6. Облачные технологии при быстрой разработке приложений..... | 6 |
| КОНТРОЛЬНЫЕ ТЕСТЫ..... | 7 |
| ВИДЫ И ФОРМЫ САМОСТОЯТЕЛЬНОЙ РАБОТЫ..... | 11 |
| КОНТРОЛЬ САМОСТОЯТЕЛЬНОЙ РАБОТЫ..... | 11 |
| КРИТЕРИИ ОЦЕНКИ РЕФЕРАТА..... | 12 |
| КРИТЕРИИ ОЦЕНКИ ЭКЗАМЕНА..... | 12 |
| МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПОДГОТОВКЕ К ТЕСТИРОВАНИЮ..... | 13 |
| МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПОДГОТОВКЕ К УСТНОМУ ОПРОСУ..... | 13 |
| МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО НАПИСАНИЮ РЕФЕРАТА..... | 14 |
| МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПОДГОТОВКЕ К ЗАЧЕТУ, ЭКЗАМЕНУ..... | 14 |
| РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ..... | 15 |
| Список рекомендуемой литературы..... | 15 |
| Электронно-библиотечные системы..... | 15 |
| Программное обеспечение..... | 16 |

ОБЩИЕ ВОПРОСЫ

В результате изучения дисциплины «Быстрая разработка приложений» студенты должны:

- 1) знать:
 - особенности low-code и no-code подходов;
 - инструментальные средства разработки low-code приложений;
- 2) уметь:
 - применять средства разработки low-code приложений;
 - взаимодействовать с технологиями мобильных устройств;
- 3) владеть:
 - средствами управления доступом мобильных ОС;
 - навыками написания low-code приложений.

Методические рекомендации для семинарских (практических) занятий и самостоятельной работы по дисциплине «Быстрая разработка приложений» направлены на повышение эффективности освоения знаний, умений, навыков и компетенций, связанных с получением студентами теоретических знаний и практических навыков в области изучения технологий разработки современных мобильных приложений.

Методические рекомендации содержат указания по всем темам дисциплины «Быстрая разработка приложений». Методические рекомендации разбиты по темам и содержат набор вопросов для систематизации теоретического материала, полученного на лекционных занятиях, и самостоятельного изучения теории, вопросы (тесты) для текущего контроля на практических занятиях (семинарах), задачи для усвоения практических навыков.

Список литературы и информационного обеспечения, приведённый в конце методических указаний, может служить основой для изучения всех рассматриваемых тем. Дополнительная и учебно-методическая литература могут быть использованы обучающимися для закрепления изучаемого материала.

РЕКОМЕНДАЦИИ ПО ОТДЕЛЬНЫМ ТЕМАМ ДИСЦИПЛИНЫ

Тема 1. Основы быстрой разработки приложений

Основные вопросы темы

- 1) Основные понятия.
- 2) История развития low-code и no-code подходов.
- 3) Технологии разработки low-code приложений.

Рекомендации по изучению темы

Вопрос 1 изложен в учебно-методическом пособии [3] на с. 4-8.
Вопрос 2 изложен в учебно-методическом пособии [3] на с. 8-9.
Вопрос 3 изложен в учебно-методическом пособии [3] на с. 10-12.

Вопросы для самоподготовки

- 1) Раскройте основные понятия быстрой разработки приложений.
- 2) В чем особенности и отличия low-code и no-code приложений?

Тема 2. Основы разработки low-code приложений

Основные вопросы темы

- 1) Среды разработки low-code приложений.
- 2) Особенности разработки low-code приложений.

Рекомендации по изучению темы

Вопрос 1 изложен в учебно-методическом пособии [4] на с. 12-13.
Вопрос 2 изложен в учебно-методическом пособии [4] на с. 14-15.
Вопрос 3 изложен в учебно-методическом пособии [4] на с. 15-18.
Вопрос 4 изложен в учебно-методическом пособии [4] на с. 19-20.

Вопросы для самоподготовки

- 1) В чём заключаются особенности разработки low-code приложений?

Тема 3. Разработка мобильных приложений с использованием low-code

Основные вопросы темы

- 1) Архитектура ОС Android.
- 2) Архитектура ОС iOS.
- 3) IDE разработки мобильных приложений.

Рекомендации по изучению темы

Вопрос 1 изложен в учебно-методическом пособии [2] на с. 21-23.
Вопрос 2 изложен в учебно-методическом пособии [2] на с. 24.

Вопросы для самоподготовки

- 1) Какие элементы входят в архитектуру ОС Android?
- 2) Какова структура приложений под ОС Android?

- 3) Какие особенности IDE для разработки low-code мобильных приложений?

Тема 4. Разработка low-code приложений для Web

Основные вопросы темы

- 1) Архитектура Web-приложений.
- 2) Средства разработки Web-приложений.

Рекомендации по изучению темы

Вопрос 1 изложен в учебно-методическом пособии [1] на с. 30-31.

Вопрос 2 изложен в учебно-методическом пособии [1] на с. 31-32.

Вопросы для самоподготовки

- 1) Какова архитектура web-приложений?
- 2) Какие средства используются для быстрой разработки Web-приложений?

Тема 5. Разработка графических элементов приложений

Основные вопросы темы

- 1) Средства разработки дизайна и элементов интерфейса.
- 2) Подходы к созданию графических элементов.

Рекомендации по изучению темы

Вопрос 1 изложен в учебно-методическом пособии [1] на с. 35-36.

Вопрос 2 изложен в учебно-методическом пособии [1] на с. 37-38.

Вопрос 3 изложен в учебно-методическом пособии [1] на с. 39-40.

Вопрос 4 изложен в учебно-методическом пособии [1] на с. 40-41.

Вопросы для самоподготовки

- 1) Какие существуют средства разработки дизайна и элементов интерфейса?

Тема 6. Облачные технологии при быстрой разработке приложений

Основные вопросы темы

- 1) Основы облачных технологий.
- 2) Вид сервисов.
- 3) Способы подключения облачных сервисов при быстрой разработке приложений

Рекомендации по изучению темы

Вопрос 1 изложен в учебно-методическом пособии [3] на с. 43-45.

Вопрос 2 изложен в учебно-методическом пособии [3] на с. 46-48.

Вопросы для самоподготовки

- 1) В чём заключаются основные особенности облачных технологий?

КОНТРОЛЬНЫЕ ТЕСТЫ

| | |
|---|--|
| 1 | <p>В чем ключевое отличие <code>Iterable<T></code> и <code>Sequence<T></code> в Kotlin?</p> <ol style="list-style-type: none">1) <code>Iterable<T></code> работает только с <code>immutable</code> коллекциями, <code>Sequence<T></code> применим к <code>mutable</code>2) Нет никакой разницы, т. к. <code>Sequence<T></code> аналог <code>Iterable<T></code>3) Последовательности обрабатываются лениво, итераторы жадно4) Последовательности обрабатываются по очереди, итераторы параллельно (многопоточно) |
| 2 | <p>В чем разница между <code>a</code> и <code>b</code>?</p> <pre>var a: String? = "KotlinQuiz" var b: String = "KotlinQuiz"</pre> <ol style="list-style-type: none">1) <code>a</code> является <code>volatile</code>, как в Java2) <code>b</code> является <code>final</code> и не может быть изменено3) <code>a</code> является <code>final</code> и не может быть изменено4) <code>b</code> никогда не сможет стать <code>null</code> |
| 3 | <p>В чем разница между <code>val</code> и <code>var</code> в Kotlin?</p> <ol style="list-style-type: none">1) Переменные, объявленные с помощью <code>val</code>, являются <code>final</code>, а переменные <code>var</code> – нет.2) Переменные, объявленные с помощью <code>val</code>, имеют доступ только к <code>const</code> членам.3) Переменные, объявленные с помощью <code>var</code>, являются <code>final</code>, а переменные <code>val</code> – нет.4) <code>var</code> ограничен видимостью ближайшего функционального блока, а у <code>val</code> видимость заканчивается на <code>({ }</code>). |
| 4 | <p>Для чего нужен оператор <code>!!</code>?</p> <ol style="list-style-type: none">1) Он возвращает левый операнд, если он не равен <code>null</code>, иначе возвращает правый операнд2) Это оператор модуля, аналог <code>%</code> в Java3) Он сравнивает два значения на тождественность4) Он преобразует любое значение в ненулевой тип и выбрасывает исключение, если значение равно <code>null</code> |
| 5 | <p>Есть ли у Kotlin примитивные типы данных, такие как <code>int</code>, <code>long</code>, <code>float</code>?</p> |

| | |
|----|--|
| | <ol style="list-style-type: none"> 1) Нет, Kotlin не имеет и не использует примитивные типы данных. 2) Нет, не на уровне языка. Но компилятор Kotlin использует примитивы JVM для лучшей производительности. 3) Да, но Kotlin всегда конвертирует их в не примитивные аналоги. 4) Да, Kotlin в этом отношении похож на Java. |
| 6 | <p>Как в Kotlin правильно объявить переменную целочисленного типа?</p> <ol style="list-style-type: none"> 1) <code>var i : int = 42</code> 2) <code>let i = 42</code> 3) <code>int i = 42</code> 4) <code>var i : Int = 42</code> |
| 7 | <p>Какое выражение Kotlin эквивалентно данному из Java?</p> <ol style="list-style-type: none"> 1) <code>int x = a ? b : c</code> 2) <code>val x = a ?: b, c</code> 3) <code>val x = if (a) b : c</code> 4) <code>val x = a ? b : c</code> 5) <code>val x = if (a) b else c</code> |
| 8 | <p>Какое из объявлений функций является валидным?</p> <ol style="list-style-type: none"> 1) <code>int sum(int a, int b)</code> 2) <code>int sum(a: Int, b: Int)</code> 3) <code>function sum(a: Int, b: Int): Int</code> 4) <code>fun sum(a: Int, b: Int): Int</code> |
| 9 | <p>Какой тип у arr?</p> <pre>val arr = arrayOf(1, 2, 3)</pre> <ol style="list-style-type: none"> 1) <code>Array<Int></code> 2) <code>Int[]</code> 3) <code>int[]</code> 4) <code>IntArray</code> |
| 10 | <p>По какой лицензии доступен Kotlin?</p> <ol style="list-style-type: none"> 1) Apache 2 2) GPL 3) Kotlin-это коммерческое программное обеспечение с закрытым исходным кодом 4) MIT |

| | |
|----|--|
| 11 | <p>Совместим ли Kotlin с Java?</p> <ol style="list-style-type: none"> 1) Kotlin может легко вызвать код Java, в то время как Java не может получить доступ к коду на Kotlin 2) Kotlin предоставляет уровень совместимости для взаимодействия с Java, который становится доступен в рантайме 3) Пока Kotlin запущен в JVM, он не может взаимодействовать с Java 4) Kotlin может легко вызвать Java код и наоборот |
| 12 | <p>Укажите правильный синтаксис для преобразования строки "42" в long</p> <ol style="list-style-type: none"> 1) <code>val l: Long = <Long>"42"</code> 2) <code>val l: Long = "42".toLong()</code> 3) <code>val l: Long = (Long)"42"</code> 4) <code>val l: Long = Long.parseLong("42")</code> |
| 13 | <p>Что не предлагает dataclass?</p> <ol style="list-style-type: none"> 1) Авто-генерируемый метод toString() 2) Метод copy(...), для создания копии экземпляров. 3) Автоматическое преобразование из/в JSON 4) Авто-генерируемые методы hashCode() и equals() |
| 14 | <p>Что выведет следующий код?</p> <pre>val list : List<Int> = listOf(1, 2, 3) list.add(4) print(list)</pre> <ol style="list-style-type: none"> 1) Он не компилируется, так как List не имеет метода add 2) [1, 2, 3, 4] 3) [5, 6, 7] 4) Он не компилируется, из-за listOf |
| 15 | <p>Что выведет следующий код?</p> <pre>val listA = mutableListOf(1, 2, 3) val listB = listA.add(4) print(listB)</pre> <ol style="list-style-type: none"> 1) [1, 2, 3, 4] |

| | |
|----|--|
| | <p>2) True</p> <p>3) Ничего, тут ошибка компиляции</p> <p>4) Unit</p> |
| 16 | <p>Что выведет этот код?</p> <pre>val a: String? = null val b: String = "Hello World" println(a==b)</pre> <p>1) Ничего, т. к. вылетит исключение NullPointerException</p> <p>2) Не скомпилируется</p> <p>3) false</p> <p>4) True</p> |
| 17 | <p>Что делает этот код?</p> <pre>foo()</pre> <p>1) Создает двумерный массив</p> <p>2) Не скомпилируется</p> <p>3) Вызывает асинхронно foo</p> <p>4) Вызывает функцию, которая вернется после вызова foo</p> |
| 18 | <p>Что из приведенного в Kotlin, эквивалентно статическому методу из Java?</p> <p>1) class Foo { @static fun bar() : String = "Kotlin" }</p> <p>2) class Foo { companion object { fun bar() : String = "Kotlin" } }</p> <p>3) class Foo { static fun bar() : String = "Kotlin" }</p> |

| | |
|----|--|
| 19 | <p>Что применимо для следующего объявления класса?</p> <pre>class Person (val name: String)</pre> <ol style="list-style-type: none"> 1) Он package-private 2) Он может быть расширен другими классами 3) Он public 4) У него приватное свойство "name" |
| 20 | <p>Что такое to в приведенном ниже примере:</p> <pre>val test = 33 to 42</pre> <ol style="list-style-type: none"> 1) Инфиксная функция, создающая пару (33, 42) 2) Ключевое слово Kotlin для создания пары (33, 42) 3) Ключевое слово для создания диапазона от 33 до 42 4) Опечатка |

ВИДЫ И ФОРМЫ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Самостоятельная работа студентов включает текущую и творческую проблемно-ориентированную самостоятельную работу (ТСР).

Текущая СРС направлена на углубление и закрепление знаний студента, развитие практических умений и включает следующие виды работ:

- работа с лекционным материалом;
- подготовка к лабораторным работам; опережающая самостоятельная работа;
- изучение тем, вынесенных на самостоятельную проработку
- подготовка к контрольным работам и промежуточной аттестации.

Творческая СРС включает:

- поиск, анализ, структурирование и презентация информации по теме раздела дисциплины, применительно к индивидуальному заданию;
- исследовательская работа и участие в научных студенческих конференциях, семинарах и олимпиадах;
- анализ научных публикаций по заранее определенной преподавателем теме;
- анализ статистических и фактических материалов по заданной теме, проведение расчетов, составление схем и моделей на основе статистических материалов

КОНТРОЛЬ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Оценка результатов самостоятельной работы организуется как единство двух форм: самоконтроль и контроль со стороны преподавателя.

Самоконтроль в обучающей программе, контроль знаний, полученных с помощью обучающей программы.

Защита индивидуального проекта.

По результатам текущего и рубежного контроля формируется допуск студента к промежуточной аттестации, которая проводится в письменной форме и оценивается преподавателем.

КРИТЕРИИ ОЦЕНКИ РЕФЕРАТА

Критериями оценки реферата являются: новизна текста, обоснованность выбора источников литературы, степень раскрытия сущности вопроса, соблюдения требований к оформлению.

Оценка «отлично» — выполнены все требования к написанию реферата: обозначена проблема и обоснована её актуальность; сделан анализ различных точек зрения на рассматриваемую проблему и логично изложена собственная позиция; сформулированы выводы, тема раскрыта полностью, выдержан объём; соблюдены требования к внешнему оформлению.

Оценка «хорошо» — основные требования к реферату выполнены, но при этом допущены недочёты. В частности, имеются неточности в изложении материала; отсутствует логическая последовательность в суждениях; не выдержан объём реферата; имеются упущения в оформлении.

Оценка «удовлетворительно» — имеются существенные отступления от требований к реферированию. В частности: тема освещена лишь частично; допущены фактические ошибки в содержании реферата; отсутствуют выводы.

Оценка «неудовлетворительно» — тема реферата не раскрыта, обнаруживается существенное непонимание проблемы или реферат не представлен вовсе.

КРИТЕРИИ ОЦЕНКИ ЭКЗАМЕНА

Экзамен - итоговая форма оценки знаний. Проводится в заданный срок, согласно графику учебного процесса.

Критерии оценки при проведении экзамена:

- Оценка "отлично" ставится, если студент обнаружил полное знание учебно-программного материала, успешно выполняет предусмотренные в программе задания, усвоил основную литературу, рекомендованную в программе. Ответ полный и правильный на основании изученного материала. Выдвинутые положения аргументированы и иллюстрированы примерами. Материал изложен в определенной логической последовательности, осознанно, литературным языком, с использованием современных научных терминов; ответ самостоятельный. Студент уверенно отвечает на дополнительные вопросы
- Оценка «хорошо» ставится в том случае, когда студент обнаруживает полное знание учебного материала, демонстрирует систематический характер знаний по дисциплине. Ответ полный и правильный, подтвержден примерами; но их обоснование не аргументировано, отсутствует собственная точка зрения. Материал изложен в определенной логической последовательности, при этом допущены 2-3 несущественные погрешности, исправленные по требованию экзаменатора. Студент испытывает незначительные трудности в ответах на дополнительные вопросы. Материал изложен осознанно, самостоятельно, с использованием современных научных терминов, литературным языком. при этом могут допускаться некоторые погрешности в ответе на зачете, если студент обладает необходимыми знаниями для их устранения под руководством преподавателя.
- Оценка «удовлетворительно» ставится в том случае, когда студент обнаруживает знание основного программного материала по дисциплине, но допускает погрешности в ответе. Ответ недостаточно логически выстроен, самостоятелен. Основные понятия употреблены правильно, но обнаруживается недостаточное раскрытие теоретического материала. Выдвигаемые положения недостаточно аргументированы и не подтверждены примерами; ответ носит преимущественно описательный характер. Студент

испытывает достаточные трудности в ответах на вопросы. Научная терминология используется недостаточно.

- Оценка «неудовлетворительно» выставляется студенту, обнаружившему проблемы в знаниях основного учебного материала по дисциплине. При ответе обнаружено непонимание студентом основного содержания теоретического материала по дисциплине. При ответе обнаружено непонимание студентом основного содержания теоретического материала или допущен ряд существенных ошибок, которые студент не может исправить при наводящих вопросах экзаменатора. Студент подменил научное обоснование проблем рассуждением бытового плана. Ответ носит поверхностный характер; наблюдаются неточности в использовании научной терминологии.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПОДГОТОВКЕ К ТЕСТИРОВАНИЮ

Цель тестирования в ходе учебного процесса студентов состоит не только в систематическом контроле за знанием изученного материала, но и в развитии умения студентов выделять, анализировать и обобщать наиболее существенные этапы технологических процессов. Как и любая другая форма подготовки к контролю знаний, тестирование имеет ряд особенностей, знание которых помогает успешно выполнить тест. Можно дать следующие методические рекомендации:

1. Прежде всего, следует внимательно изучить структуру теста, оценить объем времени, выделяемого на данный тест, увидеть, какого типа задания в нем содержатся. Это поможет настроиться на работу.

2. Очень важно всегда внимательно читать задания до конца, не пытайтесь понять условия «по первым словам» или выполнив подобные задания в предыдущих тестированиях. Такая спешка нередко приводит к досадным ошибкам в самых легких вопросах.

3. Если Вы не знаете ответа на вопрос или не уверены в правильности, следует пропустить его и отметить, чтобы потом к нему вернуться.

4. Психологи также советуют думать только о текущем задании. Как правило, задания в тестах не связаны друг с другом непосредственно, поэтому необходимо концентрироваться на данном вопросе и находить решения, подходящие именно к нему. Кроме того, выполнение этой рекомендации даст еще один психологический эффект - позволит забыть о неудаче в ответе на предыдущий вопрос, если таковая имела место.

5. Многие задания можно быстрее решить, если не искать сразу правильный вариант ответа, а последовательно исключать те, которые явно не подходят. Метод исключения позволяет в итоге сконцентрировать внимание на одном-двух вероятных вариантах.

6. Процесс угадывания правильных ответов желательно свести к минимуму, так как это чревато тем, что студент забудет о главном: умении использовать имеющиеся накопленные в учебном процессе знания, и будет надеяться на удачу. Если уверенности в правильности ответа нет, но интуитивно появляется предпочтение, то психологи рекомендуют доверять интуиции, которая считается проявлением глубинных знаний и опыта, находящихся на уровне подсознания.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПОДГОТОВКЕ К УСТНОМУ ОПРОСУ

Самостоятельная работа студентов включает подготовку к устному опросу на семинарских занятиях. Для этого студент изучает лекции, основную и дополнительную литературу, публикации, информацию из Интернет-ресурсов.

Тема и вопросы к семинарским занятиям, вопросы для самоконтроля содержатся в рабочей учебной программе и доводятся до студентов заранее. Эффективность подготовки

студентов к устному опросу зависит от качества ознакомления с рекомендованной литературой. Для подготовки к устному опросу, блиц-опросу студенту необходимо ознакомиться с материалом, посвященным теме семинара, в учебнике или другой рекомендованной литературе, записях с лекционного занятия, обратить внимание на усвоение основных понятий дисциплины, выявить неясные вопросы и подобрать дополнительную литературу для их освещения, составить тезисы выступления по отдельным проблемным аспектам. В среднем, подготовка к устному опросу по одному семинарскому занятию занимает от 2 до 4 часов в зависимости от сложности темы и особенностей организации студентом своей самостоятельной работы. За участие в устном опросе студент может получить 1 -2 балла в зависимости от полноты ответа.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО НАПИСАНИЮ РЕФЕРАТА

Выполнение реферата является одной из форм контроля в высшем учебном заведении.

Структура реферата:

Титульный лист.

1. После титульного листа на отдельной странице следует оглавление (план, содержание), в котором указаны названия всех разделов (пунктов плана) реферата и номера страниц, указывающие начало этих разделов в тексте реферата.

2. После оглавления следует введение. Объем введения составляет 1,5-2 страницы.

3. Основная часть реферата может иметь одну или несколько глав, состоящих из 23 параграфов (подпунктов, разделов) и предполагает осмысленное и логичное изложение главных положений и идей, содержащихся в изученной литературе. В тексте обязательны ссылки на первоисточники. В том случае если цитируется или используется чья-либо неординарная мысль, идея, вывод, приводится какой-либо цифрой материал, таблицу - обязательно сделайте ссылку на того автора у кого вы взяли данный материал.

4. Заключение содержит главные выводы, и итоги из текста основной части, в нем отмечается, как выполнены задачи и достигнуты ли цели, сформулированные во введении.

5. Приложение может включать графики, таблицы, расчеты. 6. Библиография (список литературы) здесь указывается реально использованная для написания реферата литература.

Этапы работы над рефератом.

Работу над рефератом можно условно подразделить на три этапа:

1. Подготовительный этап, включающий изучение предмета исследования;
2. Изложение результатов изучения в виде связного текста;
3. Устное сообщение по теме реферата.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПОДГОТОВКЕ К ЗАЧЕТУ, ЭКЗАМЕНУ

Успешно работавшим на занятиях студентам зачет, экзамен выставляется без применения специальных форм контроля знаний. Для этого студент должен присутствовать на всех занятиях, готовиться к занятиям и активно на них работать (отвечать на вопросы, дополнять ответы других студентов). Студент, который не получил зачет, экзамен в указанном порядке, должен готовиться в сдаче зачета, экзамена. Для подготовки необходимо использовать литературу, изучаемую по дисциплине, лекционный и практический материал. При этом студент должен отработать все пропущенные темы, а также темы, по которым он получил неудовлетворительные оценки. Студент на зачете, экзамене должен быть готовым ответить устно и письменно на предложенные преподавателем контрольные вопросы и правильно решить предложенные преподавателем ситуации (устно и письменно) по соответствующей теме, правильно дать ответы на вопросы по тестированию, если проводится

тестирование.

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

Список рекомендуемой литературы

основная

1. Соколова, В. В. Разработка мобильных приложений : учебное пособие / В. В. Соколова. — Томск : Томский политехнический университет, 2014. — 176 с. — ISBN 978-5-4387-0369-3. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/34706.html> (дата обращения: 05.11.2019). — Режим доступа: для авторизир. Пользователей
2. Введение в разработку приложений для ОС Android / Ю. В. Березовская, О. А. Юфрякова, В. Г. Вологодина [и др.]. — 2-е изд. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 433 с. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/73669.html> (дата обращения: 05.11.2019). — Режим доступа: для авторизир. пользователей

дополнительная

3. Вычислительная техника и информационные технологии. Разработка мобильных приложений : учеб. пособие для прикладного бакалавриата и для магистров по направл. "Информатика и вычислительная техника" / Соколова Вероника Валерьевна; Нац. исслед. Томск. политехн. ун-т. - Москва : Юрайт, 2018. - 175 с. - (Университеты России). - Библиогр.: с.174-175. - ISBN 978-5-9916-6525-4 (в пер.) : 150.00

учебно-методическая

4. Семакова, А. Введение в разработку приложений для смартфонов на ОС Android / А. Семакова. — 2-е изд. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 102 с. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/73670.html> (дата обращения: 05.11.2019). — Режим доступа: для авторизир. Пользователей.

Электронно-библиотечные системы

1.1. IPRbooks : электронно-библиотечная система : сайт / группа компаний Ай Пи Ар Медиа. - Саратов, [2020]. — URL: <http://www.iprbookshop.ru>. — Режим доступа: для зарегистрир. пользователей. - Текст : электронный.

1.2. ЮРАЙТ : электронно-библиотечная система : сайт / ООО Электронное издательство ЮРАЙТ. — Москва, [2020]. - URL: <https://www.biblio-online.ru>. — Режим доступа: для зарегистрир. пользователей. - Текст : электронный.

1.3. Консультант студента: электронно-библиотечная система : сайт / ООО Политехресурс. — Москва, [2020]. — URL: http://www.studentlibrary.ru/catalogue/switch_kit/x2019-128.html. — Режим доступа: для зарегистрир. пользователей. — Текст : электронный.

1.4. Лань : электронно-библиотечная система : сайт / ООО ЭБС Лань. — Санкт-Петербург, [2020]. — URL: <http://www.studentlibrary.ru/pages/catalogue.html> <https://e.lanbook.com>. — Режим доступа: для зарегистрир. пользователей. — Текст : электронный.

1.5. **Znanium.com** : электронно-библиотечная система : сайт / ООО Знаниум. - Москва, [2020]. - URL: <http://www.studentlibrary.ru/pages/catalogue.html> <http://znanium.com>. – Режим доступа : для зарегистрир. пользователей. - Текст : электронный.

2. КонсультантПлюс [Электронный ресурс]: справочная правовая система. /ООО «Консультант Плюс» - Электрон. дан. - Москва : КонсультантПлюс, [2020].

3. Базы данных периодических изданий:

3.1. База данных периодических изданий : электронные журналы / ООО ИВИС. - Москва, [2020]. – URL: <https://dlib.eastview.com/browse/udb/12>. – Режим доступа : для авториз. пользователей. – Текст : электронный.

3.2. eLIBRARY.RU: научная электронная библиотека : сайт / ООО Научная Электронная Библиотека. – Москва, [2020]. – URL: <http://elibrary.ru>. – Режим доступа : для авториз. пользователей. – Текст : электронный

3.3. «Grebennikon» : электронная библиотека / ИД Гребенников. – Москва, [2020]. – URL: <https://id2.action-media.ru/Personal/Products>. – Режим доступа : для авториз. пользователей. – Текст : электронный.

4. Национальная электронная библиотека : электронная библиотека : федеральная государственная информационная система : сайт / Министерство культуры РФ ; РГБ. – Москва, [2020]. – URL:<http://www.studentlibrary.ru/pages/catalogue.html> <https://нэб.рф>. – Режим доступа : для пользователей научной библиотеки. – Текст : электронный.

5. SMART Imagebase // EBSCOhost : [портал]. – URL: <https://ebSCO.smartimagebase.com/?TOKEN=EBSCO-1a2ff8c55aa76d8229047223a7d6dc9c&custid=s6895741>. – Режим доступа : для авториз. пользователей. – Изображение : электронные.

6. Федеральные информационно-образовательные порталы:

6.1. [Единое окно доступа к образовательным ресурсам](http://window.edu.ru/) : федеральный портал / учредитель ФГАОУ ДПО ЦРГОП и ИТ. – URL: <http://window.edu.ru/>. – Текст : электронный.

6.2. [Российское образование](http://www.edu.ru/) : федеральный портал / учредитель ФГАОУ ДПО ЦРГОП и ИТ. – URL: [http://www.edu.ru.](http://www.edu.ru/) – Текст : электронный.

7. Образовательные ресурсы УлГУ:

7.1. Электронная библиотека УлГУ : модуль АБИС Мега-ПРО / ООО «Дата Экспресс». – URL: <http://lib.ulsu.ru/MegaPro/Web>. – Режим доступа : для пользователей научной библиотеки. – Текст : электронный.

7.2. Образовательный портал УлГУ. – URL: <http://edu.ulsu.ru>. – Режим доступа : для зарегистрир. пользователей. – Текст : электронный

Программное обеспечение

1. Стандартные приложения Windows: Блокнот, WordPad.
2. Программы Microsoft Office.
3. Figma.
4. WordPress.
5. Marvel.
6. uKit
7. glide.