

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ЭКОНОМИКИ И УПРАВЛЕНИЯ



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б1.2.26.1 Современные средства разработки программного обеспечения
(индекс дисциплины по учебному плану, наименование дисциплины (модуля))

Направление подготовки 38.03.05 "Бизнес-информатика"
(код, наименование направления подготовки)

Квалификация выпускника – *Бакалавр*

Форма обучения очная
(очная, заочная, очно-заочная)

Пенза, 2017

1. Цели освоения учебной дисциплины

Целями освоения учебной дисциплины «Б1.2.26.1 Современные средства разработки программного обеспечения» являются:

- получение студентами теоретических знаний и практических навыков в программировании, позволяющих решать задачи обработки числовой и символьной информации в рамках прикладных и системных задач;
- формирование у студентов профессиональных компетенций, связанных с использованием теоретических знаний и практических навыков в области современных средств разработки программного обеспечения, позволяющих творчески применять их для решения задач создания программного обеспечения и обработки информации как в своей профессиональной деятельности, так и при выполнении курсовых и практических работ при последующем обучении на старших курсах.

2. Место учебной дисциплины в структуре ОПОП бакалавриата

Дисциплина «Современные средства разработки программного обеспечения» относится к дисциплинам по выбору вариативной части блока Б1 дисциплин подготовки студентов по направлению 38.03.05 «Бизнес-информатика».

Изучение дисциплины «Современные средства разработки программного обеспечения» базируется на положениях следующих дисциплин: «Теоретические основы информатики», «Программирование», «Объектно-ориентированный анализ и программирование», «Распределенные системы», «Управление информационными системами и интернет-ресурсами».

Основные положения данной дисциплины могут быть использованы в дальнейшем при работе над выпускной квалификационной работой.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) Б1.2.26.1 «Современные средства разработки программного обеспечения»

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО по данному направлению:

Коды компетенции	Наименование компетенции	Структурные элементы компетенции (в результате освоения дисциплины обучающийся должен знать, уметь, владеть)
1	2	3
ОПК-3	Способность выбрать инструментальные средства для обработки экономических данных в соответствии с поставленной задачей, проанализировать результаты расчетов и обосновать полученные выводы	Знать: - <i>инструментальные средства для обработки экономических данных</i>
		Уметь: - <i>выбрать инструментальные средства для обработки экономических данных в соответствии с поставленной задачей;</i> - <i>анализировать результаты расчетов и обосновывать полученные выводы</i>
		Владеть: - <i>инструментальными средствами для обработки экономических данных и разработки современных программных систем в соответствии с поставленной задачей</i>

4. Структура и содержание дисциплины (модуля) Б1.2.26.1 «Современные средства разработки программного обеспечения»

4.1. Структура дисциплины (модуля)

Общая трудоемкость освоения дисциплины составляет 2 зачетных единицы, 72 часа. Программой дисциплины предусмотрены: лабораторные работы (22 часа), самостоятельная работа студента (50 часов) в восьмом семестре.

Очная форма обучения

№ п/п	Наименование разделов и тем дисциплины (модуля)	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)								Формы текущего контроля успеваемости (по неделям семестра)									
				Аудиторная работа				Самостоятельная работа				Собеседование	Коллоквиум	Проверка тестов	Проверка контр. работ	Проверка реферата	Проверка эссе и иных творческих работ	курсовая работа (проект)	Тестирование		
				Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Курсовая работа (проект)									Подготовка к зачету	
1.	Введение в программирование на языке Java	8	1-2	4			4	6	6					1-2							
2.	Абстрактные классы и интерфейсы	8	3-4	4			4	10	10					3-4							
3.	Основы оконной графики	8	5-6	4			4	8	8					5-6							
4.	Создание апплетов	8	7-8	4			4	8	8					7-8							
5.	Реализация многопоточности в Java	8	9-11	6			6	18	10			8	9-11	10-11							10-11
	Общая трудоемкость, в часах	72		22			22	50	42			8	Промежуточная аттестация								
Форма													Семестр								
Зачет													8								

4.2. Содержание дисциплины (модуля)

Тема 1. Введение в программирование на языке Java.

JAVA - объектно-ориентированный язык программирования. Базовые типы данных. Операции (operators) в языке Java (Операции сравнения. Операции инкремента, декремента). Литералы (константы). Операторы. Массивы в Java. Комментарии. Первая программа на языке Java.

Практическое задание.

Тема 2. Абстрактные классы и интерфейсы.

Описание абстрактных классов. Описание интерфейсов.

Практическое задание.

Тема 3. Основы оконной графики.

Пакет AWT. Пакет Swing. . Модель событий в Swing.

Практическое задание.

Тема 4. Создание апплетов.

Понятие апплетов. Проблема безопасности. Создание апплетов. Апплеты и приложения. Запуск апплетов.

Практическое задание.

Тема 5. Реализация многопоточности в Java.

Множественные нити выполнения (Multiple threads). Методы класса Thread. Реализация интерфейса Runnable. Завершение и останов нити. Приоритеты нитей. Средства синхронизации нитей в Java. Применение многопоточности для анимации.

Практическое задание.

5. Образовательные технологии

Технологии получения студентами теоретических знаний связаны с лекционными занятиями и основаны на принципах самостоятельной внеаудиторной работы с рекомендуемым учебно-методическим и информационным обеспечением учебной дисциплины. Освоение теоретического и методологического материала предполагается также за счет проведения Интернет-тестирования и Интернет-тренингов с использованием баз ФЕПО и специализированных сайтов.

Технология формирования и закрепления практических навыков и умений у студентов базируется на выполнении и защите лабораторных и курсовых работ, основанных на компьютерных симуляциях, связанных с созданием и разработкой программ на языке программирования высокого уровня.

Удельный вес занятий, проводимых в интерактивных формах, определяется главной целью программы, особенностью контингента обучающихся и содержанием конкретных дисциплин, и в целом в учебном процессе они составляют не менее 40% аудиторных занятий (определяется требованиями ФГОС с учетом специфики ОПОП).

6. Учебно-методическое обеспечение самостоятельной работы студентов. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Контрольные вопросы:

1. Язык JAVA как объектно-ориентированный язык программирования.
2. Сущность и основные принципы языка JAVA.
3. Понятие платформы программирования.

4. Платформа J2SE. Основные достоинства, недостатки и особенности языка JAVA.
5. Основные области применения языка Java.
6. Основные понятия языка Java. Классы, объекты, интерфейсы, пакеты.
7. Структура файла исходного кода.
8. Виды приложений на языке Java.
9. Структура приложения в платформе J2SE.
10. Пример приложения на языке Java. Компиляция и отладка приложений.
11. Основные элементы языка. Лексемы. Идентификаторы. Ключевые слова.
12. Основные элементы языка. Константы.
13. Основные элементы языка. Примитивные типы данных—целые,символьный, вещественные, логические.
14. Основные элементы языка. Объявление и инициализация переменных. Время жизни переменных.
15. Операторы языка Java. Приоритет и правила ассоциативности. Порядок вычисления операндов.
16. Оператор присваивания =. Присваивание примитивных значений, присваивание ссылок, многократные присваивания.
17. Арифметические операторы. Приоритет и ассоциативность арифметических операторов.
18. Арифметические операторы. Порядок вычислений в арифметических выражениях.
19. Арифметические операторы. Диапазон числовых значений.
20. Арифметические операторы. Унарные и бинарные арифметические операторы.
21. Арифметические операторы. Составные операторы присваивания. Операторы декремента и инкремента.
22. Операторы отношения. Равенство. Равенство значений примитивных типов данных.
23. Равенство ссылок на объекты. Равенство значений объектов.
24. Булевы логические операторы.
25. Логические составные операторы присваивания.
26. Условные операторы.
27. Целочисленные поразрядные операторы.
28. Операторы сдвига.
29. Унарный оператор приведения (тип).
30. Преобразования сужения и расширения.
31. Числовые расширения.
32. Примеры преобразования типов. Неявное преобразование типов. Правила приведения.
33. Массивы. Объявление переменных -массивов.
34. Создание массива. Инициализация массива.
35. Использование массивов. Анонимные массивы. Многомерные массивы.
36. Операторы ветвления – сокращенный оператор if.
37. Оператор if –else.
38. Оператор switch.
39. Операторы цикла. Оператор while. Оператор do-while.
40. Оператор цикла for.
41. Команды перехода. Маркированные операторы. Операторы break, continue, return.
42. Определение классов. Модификаторы объявления классов.

43. Структура класса – поля, методы, конструкторы, блоки инициализации.
44. Определение полей. Модификаторы объявления полей. Инициализация значений.
45. Блоки инициализации.
46. Неизменяемые поля.
47. Методы класса. Модификаторы объявления методов классов. 45. Методы класса.

Передача параметров.

48. Статические методы. 47. Неизменяемые методы. 48. Абстрактные методы.
49. Модификаторы и правила видимости. 50. Объектная ссылка this.
50. Перегрузка методов. 52. Конструкторы.
51. Конструктор по умолчанию. Перегруженные конструкторы.
52. Блоки инициализации.
53. Объекты. Создание объектов. Время жизни объекта и сборка мусора.
54. Одиночное наследование. Переопределение и скрытие методов.
55. Соккрытие (инкапсуляция) полей;
56. Соккрытие статического метода.
57. Объектная ссылка super. Организация цепочки конструкторов.
58. Интерфейсы.
59. Вложенные классы и интерфейсы.
60. Назначение исключительных ситуаций.
61. Типы исключительных ситуаций.
62. Исключительная ситуация как объект.
63. Основные классы исключительных ситуаций. Классы Exception, RuntimeException,

Error.

64. Определение новых классов исключительных ситуаций.
65. Обработка исключений – try, catch, finally. Оператор throw.
66. Генерация исключительной ситуации.
67. Диагностические утверждения.
68. Основные классы пакета java.lang. Класс Object.
69. Классы оболочки над примитивными типами.
70. Неизменяемые строки - класс String.
71. Изменяемые строки – класс StringBuffer.
72. Класс Math – реализация основных математических функций.
73. Инструментальный набор коллекций. Классы и интерфейсы для реализации коллекций и карт.
74. Коллекции – основные действия, массовые операции, операции с массивами, итераторы.
75. Множества – классы HashSet и LinkedHashSet.
76. Списки ArrayList, LinkedList
77. Карты. Классы HashMap, LinkedHashMap и Hashtable.
78. Отсортированные множества и отсортированные карты.
79. Интерфейсы Comparator, Comparable, SortedSet, SortedMap.
80. Работа с коллекциями.
81. Организация чтения/записи байтовых потоков. 83. Классы InputSteamReader и OutputStreamReader.
82. Буферизованные потоки чтения и записи – класс BufferedReader. 85. Чтение и запись из файлов.
83. Организация взаимодействия с консолью.

84. Многозадачность. Обзор потоков.
85. Главный поток.
86. Создание потока, реализация интерфейса Runnable.

Задания для самостоятельной работы:

См. «Методические указания по проведению самостоятельной работы по дисциплине».

6.1 План самостоятельной работы студентов очной формы обучения

№ нед.	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Количество часов
1-2	Введение в программирование на языке Java.	1. Подготовка к аудиторным занятиям	1. Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	6
3-4	Абстрактные классы и интерфейсы.	1. Подготовка к аудиторным занятиям	1. Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	10
5-8	Основы оконной графики.	1. Подготовка к аудиторным занятиям	1. Изучение рекомендуемой литературы и необходимых сайтов	См. список основной и дополнительной литературы	8
7-8	Создание апплетов.	1. Подготовка к аудиторным занятиям	1. Изучение рекомендуемой литературы и сайтов	См. список основной и дополнительной литературы	8
9-11	Реализация многопоточности в Java.	1. Подготовка к аудиторным занятиям	1. Изучение рекомендуемой литературы и сайтов	См. список основной и дополнительной литературы	10

6.2. Методические указания по организации самостоятельной работы студентов

1. Подготовка к аудиторным занятиям представляет собой изучение рекомендуемой литературы и сайтов, решение контрольных вопросов по лабораторным работам.

2. Написание курсовой работы предполагает обобщение основных теоретических, методических и прикладных положений по выбранной теме в разрезе следующей структуры: актуальность, цель и задачи, основное содержание, ожидаемые результаты и возможность их использования на практике, выводы и предложения, список литературы (согласно методическим указаниям).

6.3. Материалы для проведения текущего и промежуточного контроля знаний студентов

№ п\п	Вид контроля	Контролируемые темы (разделы)	Компетенции, компоненты которых контролируются
1.	Собеседование 1	Введение в программирование на языке Java.	ОПК-3
2.	Собеседование 2	Абстрактные классы и интерфейсы.	ОПК-3
3.	Собеседование 3	Основы оконной графики.	ОПК-3
4.	Собеседование 4	Создание апплетов.	ОПК-3

5.	Собеседование 5	Реализация многопоточности в Java.	ОПК-3
----	-----------------	------------------------------------	-------

Текущий контроль успеваемости в виде контрольных точек проводится по результатам защиты 5-и лабораторных работ в 8-ом семестре, каждая из которых оценивается по 7-12 баллов. Количество контрольных точек и сроки их проведения в семестре устанавливается по решению деканата.

Темы лабораторных работ

1. Введение в программирование на языке Java.
2. Абстрактные классы и интерфейсы.
3. Основы оконной графики.
4. Создание апплетов.
5. Реализация многопоточности в Java.

По результатам текущего контроля студент может получить от 0 до 60 баллов. Для допуска к зачету студент должен набрать не менее 36 баллов.

Собеседование 1. Введение в программирование на языке Java..

- Что такое расширяющее преобразование типов?
- Что такое примитивный тип в Java?
- Что такое явное и неявное приведение типов?
- Какие существуют способы инициализации массивов?
- В чем состоит принципиальное отличие между классами String и StringBuffer?
- Какие типы исключительных ситуаций вы знаете? К исключительным ситуациям какого типа относится java.lang.Error?;
- Допустимо ли преобразование объекта класса Integer к типу String?
- Какие методы предусмотрены в классе Integer для представления числовых значений в различных системах счисления?

Собеседование 2. Абстрактные классы и интерфейсы.

- Дайте определения понятиям статический метод и статическое поле класса.
- Какие методы классов-оболочек над примитивными типами используются для получения значения примитивного типа из его строкового представления.
- Для какого примитивного типа не существует класса-оболочки?
- Каким образом реализуется наследования в Java?
- Каким образом класс-потомок может обратиться полям и методам суперкласса?
- В какой последовательности осуществляется вызов конструкторов классов, являющихся суперклассами для данного класса?

Собеседование 3. Основы оконной графики.

- Основные библиотеки для разработки графического пользовательского интерфейса.
- Какие существуют различные графические компоненты в Java?
- Достоинства и недостатки основных графических пакетов Java.
- Как подключить библиотеку swing?
- Где осуществляется работа с графикой в Java?

Собеседование 4. Создание апплетов.

- Чем выполнение апплета отличается от выполнения простого Java-приложения?
- Чем отличаются первичные классы приложения и апплета?

- Какие методы должен переопределять первичный класс апплета?
- Каковы принципы функционирования апплета?
- Как передаются параметры апплету?
- Чем отличаются Java-приложения и Java-апплеты?
- Какие основные составные части должны присутствовать в каждой Java-программе, их функции (назначение)?
- Что такое первичный класс приложения? Какой обязательный метод он должен содержать?

Собеседование 5. Реализация многопоточности в Java.

- Каким образом ведут себя главный поток приложения и порожденные им потоки, после выполнения всех операций?
- В чем заключаются отличия между классами TreeSet и HashSet?
- Объясните понятие и назначение финализации объектов.
- Приведите примеры использования класса ArrayList.
- На основе какой структуры данных реализован класс HashMap?
- Какие классы платформы J2SE существуют для поддержки многопоточности?

Промежуточная аттестация студентов по итогам освоения дисциплины осуществляется в виде зачета в восьмом семестре. Набранные во время аттестации баллы заключены в интервале от 0 до 40.

Результатом зачета является сумма баллов текущего контроля и баллов, набранных на зачете/экзамене. Количество баллов по итогам зачет/экзамена варьируется от 60 до 100.

7. Учебно-методическое и информационное обеспечение дисциплины (модуля) Б1.2.26.1 «Современные средства разработки программного обеспечения»

а) основная литература:

1. Гаврилов А.В., Клименков С.В., Цопа Е.А. Программирование на Java: Конспект лекций. - СПб.: СПбГУ ИТМО, 2010. - 130 с. - Режим доступа: <http://window.edu.ru/resource/761/72761>
2. Сухов, С. А. Программирования сетевых взаимодействий в Java: методические указания / С. А. Сухов. - Ульяновск: УлГТУ, 2010. - 52 с. - Режим доступа: <http://window.edu.ru/resource/458/74458>

б) дополнительная литература:

1. Мухамедзянов, Р.Р. JAVA. Серверные приложения. [Электронный ресурс] - Электрон. дан. - М.: СОЛОН-Пресс, 2007. - 336 с. - Режим доступа: <http://e.lanbook.com/book/13633>
2. Машнин Т. С. Современные Java-технологии на практике. - СПб.: БХВ-Петербург, 2010. - 560 с. - (Профессиональное программирование). - ISBN 978-5-9775-0561-1. - Режим доступа: <http://znaniium.com/catalog.php?bookinfo=351236>

в) программное обеспечение и Интернет-ресурсы

1. Операционная система Windows версии XP и выше
2. Система программирования Microsoft Visual Studio 2010 (лицензионное программное обеспечение);
3. MS Office 2003 и выше
4. Файловая оболочка типа Total Commander
5. Электронно-библиотечная система: e.lanbook.com


6. Федеральный образовательный портал «Единое окно доступа к образовательным ресурсам» window.edu.ru
7. Электронно-библиотечная система: znanium.com

8. Материально-техническое обеспечение учебной дисциплины (модуля)

1. Лаборатории по информационно-коммуникационным технологиям;
2. Компьютерные учебные классы;
3. Пакеты прикладных программ;
4. Слайды, учебные видеофильмы, плакаты.

Рабочая программа дисциплины «Объектно-ориентированный анализ и программирование» составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 38.03.05 «Бизнес-информатика».

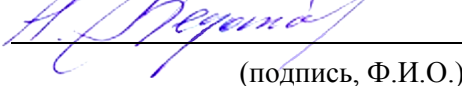
Программу составили:

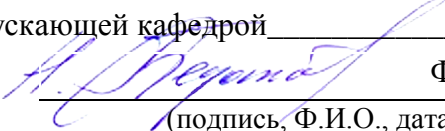
1. Абрамова Т.А., ст. преподаватель 
(Ф.И.О., должность, подпись)
2. _____
(Ф.И.О., должность, подпись)

Настоящая программа не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.

Программа одобрена на заседании кафедры Экономическая кибернетика


Протокол № 1 от « 1 » сентября 2017 года

Зав. кафедрой ЭК  Федотов Н.Г.
(подпись, Ф.И.О.)

Программа согласована с заведующим выпускающей кафедрой _____
Экономическая кибернетика  Федотов Н.Г.
(название кафедры) (подпись, Ф.И.О., дата)

Программа одобрена методической комиссией факультета экономики и управления

Протокол № 1 от « 28 » сентября 2017 года

Председатель методической комиссии факультета экономики и управления  Еремина Е.В.
(подпись) (Ф.И.О.)

