

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Б1.В.ДВ.02.01
(индекс дисциплины)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование на Java (Джава) 1

(наименование дисциплины)

по направлению подготовки
09.03.03 Прикладная информатика

направленность (профиль)
Прикладная информатика в информационной безопасности

Форма обучения: очная

Год набора: 2023

Общая трудоемкость: 5 ЗЕ

Распределение часов дисциплины по семестрам

Семестр	4	Итого
Форма контроля	экзамен	
Вид занятий		
Лекции	32	32
Лабораторные		
Практические	32	32
Руководство: курсовые работы (проекты) / РГР		
Промежуточная аттестация	0.35	0.35
Контактная работа	64.35	64.35
Самостоятельная работа	80	80
Контроль	35.65	35.65
Итого	180	180

Рабочую программу составил(и):

Доцент, к.п.н., Ерофеева Елена Александровна

(должность, ученое звание, степень, Фамилия И.О.)

(должность, ученое звание, степень, Фамилия И.О.)

Рецензирование рабочей программы дисциплины:



Отсутствует



Рецензент

(должность, ученое звание, степень, Фамилия И.О.)

Рабочая программа дисциплины составлена на основании ФГОС ВО и учебного плана направления подготовки

09.03.03 Прикладная информатика

Срок действия рабочей программы дисциплины до «31» августа 2027 г.

СОГЛАСОВАНО

Директор

Института инженерной и экологической безопасности

«__» _____ 20__ г.

(подпись)

Л. Н. Горина

(И.О. Фамилия)

УТВЕРЖДЕНО

На заседании кафедры

«Прикладная математика и информатика»

(протокол заседания № 1 от «30» августа 2022 г.).

1. Цель освоения дисциплины

Цель освоения дисциплины – освоение технологии разработки, тестирования и развертывания программного обеспечения на платформе Java SE.

2. Место дисциплины в структуре ОПОП ВО

Дисциплины и практики, на освоении которых базируется данная дисциплина: Основы программирования, Базы данных и управление данными, Объектно-ориентированное программирование.

Дисциплины и практики, для которых освоение данной дисциплины необходимо как предшествующее: – Программирование на Java (Джава) 2, Производственная практика (технологическая (проектно-технологическая) практика), Производственная практика (преддипломная практика).

3. Планируемые результаты обучения

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
ПК-5 Способен осуществлять выбор языка программирования и моделировать решение для реализации программного обеспечения	ПК-5.1 Знает принципы выбора языка программирования и моделирования решений для реализации программного обеспечения	Знать: технологии моделирования программного обеспечения Уметь: применять технологии моделирования программного обеспечения Владеть :навыками моделирования программного обеспечения
	ПК-5.2 Умеет использовать знания по выбору языка программирования и моделированию решений для реализации программного обеспечения	Знать: технологию разработки программного обеспечения на языках программирования Уметь: выбирать и моделировать решения по разработке программного обеспечения на языках программирования Владеть: навыками реализации программного обеспечения на языках программирования
	ПК-5.3 Владеет навыками выбора языка программирования и моделирования решений для реализации программного обеспечения	Знать: технологии моделирования решения для реализации программного обеспечения на языках программирования Уметь: моделировать решения для реализации программного обеспечения на языках программирования Владеть: инструментом моделирования решения для реализации программного обеспечения на языках программирования

4. Структура и содержание дисциплины

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерак- тив, ч.	Формы текущего контроля (наименование оценочного средства)
	Лек 1	Введение в Java	4	2		-	Тестовые задания
	ПрЗ 1	Утилиты JDK	4	2	6	-	Отчет по практической работе
	Лек 2	Основы синтаксиса Java.	4	2		-	Тестовые задания
	ПрЗ 2	Java Syntax 1	4	2	6	-	Отчет по практической работе
	Лек 3	Объектно- ориентированное программирования в Java	4	2		-	Тестовые задания
	ПрЗ 3	Объектно- ориентированное программирования в Java 1	4	2	6	-	Отчет по практической работе
	Лек 4	Интерфейсы	4	2		-	Тестовые задания
	ПрЗ 4	Интерфейсы	4	2	6	-	Отчет по практической работе
	Лек 5	Обработка строк на Java	4	2		-	Тестовые задания
	ПрЗ 5	Обработка строк на Java 1	4	2	6	-	Отчет по практической работе
	Лек 6	Работа с датой и временем на Java	4	2		-	Тестовые задания
	ПрЗ 6	Работа с датой и временем на Java	4	2	6	-	Отчет по практической работе
	Лек 7	Обобщенные типы.	4	2		-	Тестовые задания
	ПрЗ 7	Обобщенные типы	4	2	6	-	Отчет по практической работе
	Лек 8	Java Collection Framework 1	4	2		-	Тестовые задания
	ПрЗ 8	Java Collection Framework 1	4	2	6	-	Отчет по практической работе
	Лек 9	Java I/O и java NIO	4	2		-	Тестовые задания
	ПрЗ 9	Java I/O и java NIO	4	2	6	-	Отчет по практической работе
	Лек 10	Обработка исключений	4	2		-	Тестовые задания
	ПрЗ 10	Обработка исключений	4	2	6	-	Отчет по практической работе
	Лек 11	Лямбда выражения	4	2		-	Тестовые задания
	ПрЗ 11	Лямбда выражения	4	2	6	-	Отчет по практической работе

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерак- тив, ч.	Формы текущего контроля (наименование оценочного средства)
	Лек 12	Регулярные выражения	4	2		-	Тестовые задания
	Пр3 12	Регулярные выражения	4	2	6	-	Отчет по практической работе
	Лек 13	JAVA DATABASES	4	2		-	Тестовые задания
	Пр3 13	JAVA DATABASES	4	2	6	-	Отчет по практической работе
	Лек 14	Работа с XML на Java	4	2		-	Тестовые задания
	Пр3 14	Работа с XML на Java	4	2	6	-	Отчет по практической работе
	Лек 15	Паттерны проектирования Java	4	2		-	Тестовые задания
	Пр3 15	Паттерны проектирования Java	4	2	6	-	Отчет по практической работе
	Лек 16	Многопоточность и синхронизация в Java	4	2		-	Тестовые задания
	Пр3 16	Многопоточность и синхронизация в Java	4	2	6	-	Отчет по практической работе
	Самостоя- тельное изучение		4	80		-	
	Итогемы й тест по курсу		4	2		-	тест
Итого:				68	100		

Схема расчета итогового балла

Текущий рейтинг (все занятия и промежуточные тесты) + Результат итогового теста и все делится на 2

5. Образовательные технологии

В рамках изучения дисциплины предусмотрено использование следующих образовательных технологий:

- технология традиционного обучения;
- интерактивные технологии: учебные дискуссии (применяются во всех модулях по итогам выполнения работ).

Технологии традиционного обучения - организация учебного процесса в вузе, основанная на лекционных и практических формах обучения: объяснительно-иллюстративное обучение. Данная технология применяется во всех модулях курса.

Технология интерактивного обучения - организация учебного процесса, которая предполагает максимальную активность студентов в процессе формирования ключевых компетенций. На учебной дискуссии студенты представляют результат выполнения заданной работы. Проводится дискуссия по применённым решениям, обсуждается эффективность и архитектура кода.

6. Методические указания по освоению дисциплины

6.1. Рекомендации по подготовке к практическим занятиям

Студентам следует:

- при подготовке к занятиям обязательно использовать не только учебную литературу, но и другие источники;
- в начале занятий задать преподавателю вопросы по материалу, вызвавшему затруднения в его понимании и освоении при решении задач, заданных для самостоятельного решения;
- на занятии доводить каждую задачу до окончательного решения, демонстрировать понимание проведенных расчетов (анализов, ситуаций), в случае затруднений обращаться к преподавателю.

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками.

Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

6.2. Рекомендации по подготовке к экзамену

Подготовка к экзамену способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На экзамене студент демонстрирует то, что он приобрел в процессе обучения по конкретной учебной дисциплине.

На консультации перед экзаменом студенты должны быть ознакомлены с основными требованиями и получить ответы на возникающие в процессе подготовки вопросы.

Необходимо ориентировать студентов на систематическую подготовку к занятиям в течение семестра, что позволит использовать время экзаменационной сессии для систематизации знаний.

7. Оценочные средства

7.1. Паспорт оценочных средств

Семестр	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
4	ПК-5	Тестовые задания Вопросы к экзамену Отчеты по практическим занятиям

7.2. Типовые задания или иные материалы, необходимые для текущего контроля

7.2.1. Примеры тестовых заданий (наименование оценочного средства)

Типовые примеры заданий

1. Технология Java представляет собой:

- ☒ высокоуровневый объектно-ориентированный язык программирования
- ☒ программную платформу
- ☐ низкоуровневый язык программирования
- ☐ язык сценариев

2. Что такое JVM (Java Virtual Machine)?

- ☒ виртуальная машина Java
- ☐ компилятор
- ☐ набор специальных библиотек
- ☐ техническое обеспечение

3. Исходные файлы (.java) содержат:

- ☒ исходный код программ на Java
- ☐ скомпилированные Java- програм - байтовые коды, определенные спецификацией Java
- ☐ потоковое представление текущего состояния объектов
- ☐ наборы файлов, которые могут быть представлены в упакованном виде

4. Файлы классов (.class) содержат:

- ☒ скомпилированные Java- програм - байтовые коды, определенные спецификацией Java
- ☐ исходный код программ на Java
- ☐ потоковое представление текущего состояния объектов
- ☐ наборы файлов, которые могут быть представлены в упакованном виде

5. Файлы сериализации (.ser) содержат:

- ☒ потоковое представление текущего состояния
- ☐ исходный код программ на Java
- ☐ скомпилированные Java- програм - байтовые коды, определенные спецификацией Java
- ☐ наборы файлов, которые могут быть представлены в упакованном виде

6. Файлы архивов (.jar) содержат:

- ☐ потоковое представление текущего состояния
- ☐ исходный код программ на Java
- ☐ скомпилированные Java-программы - байтовые коды, определенные спецификацией Java
- ☒ наборы файлов, которые могут быть представлены в упакованном виде

7. Java Development Kit (JDK) -это:

- ☐ виртуальная машина Java
- ☐ исходный код программ на Java
- ☐ не является компонентой Java
- ☒ набор библиотек и утилит для разработки программ на Java

8. Назначение утилиты javac?

- ☐ утилита для запуска в режиме командной строки откомпилированных программ-приложений
- ☐ утилита для запуска на исполнение и отладку апплетов без браузера. При этом не гарантируется работоспособность отлаженного апплета в браузере
- ☐ отладчик программ, написанных на языке Java
- ☒ компилятор в режиме командной строки для программ, написанных на языке Java

9. Что будет, если ввести команду

```
jar -cfv jar.jar Hello.java Hello.class
```

- ☐ запустится класс Hello из jar-архива
- ☐ файл Hello.java скомпилируется и будет создан файл Hello.class
- ☒ будет создан jar архив с файлами Hello.java и Hello.class
- ☐ ошибка (Error)

10. Параметр -cp позволяет?

- ☒ указать путь, по которому можно найти классы для компиляции
- ☒ указать путь, по которому можно найти классы для запуска JVM
- ☐ вывести версию компилятора
- ☐ вывести версию исходного кода

11. Что произойдет при использовании данной команды

```
java HelloWorld
```

- ☐ компиляция jar файла «HelloWorld»
- ☒ запустится байт-код «HelloWorld»
- ☐ компиляция исходного Java-кода «HelloWorld» в байт-код
- ☐ ошибка

12. Что выполняет данная команда javac hellofromfirst.java?

- ☒ компиляция исходного Java-кода с именем «hellofromfirst» в байт-код с таким же названием
- ☐ компиляция исходного Java-кода с именем «hellofromfirst» в jar с таким же названием
- ☐ компиляция исходного Java-кода с именем «hellofromfirst» в манифест с таким же названием
- ☐ компиляция исходного Java-кода с именем «hellofromfirst» в байт-код с другим названием

13. Базовый пакет java.lang?

- ☐ поддержка вычислений с целыми числами произвольной длины а так же с числами с плавающей точкой произвольной точности
- ☐ содержит важные классы для работы со структурами данных (работа с событиями и датами)
- ☒ содержит базовые классы языка java
- ☐ поддержка базовых средств ввода/вывода

14. Базовый пакет java.io?

- ☐ поддержка вычислений с целыми числами произвольной длины а так же с числами с плавающей точкой произвольной точности
- ☐ содержит важные классы для работы со структурами данных (работа с событиями и датами)
- ☐ содержит базовые классы языка java
- ☒ поддержка базовых средств ввода/вывода

15. Пакет (package) в Java это?

- ☒ контейнер, который используется для того, чтобы изолировать имена классов
- ☐ базовый класс языка java
- ☐ базовый метод языка java
- ☐ интерфейс

16. Выберите правильный вариант значений переменных a,b,c,d чтобы результат выражения $a \&\&(b \parallel c)^d$ являлся ложью?

- ☐ a=false, b=false, c=false, d=true
- ☒ все ответы верны
- ☐ a=true, b=true, c=false, d=true
- ☐ a=false, b=true, c=false, d=true

17. Выберите правильный вариант значений переменных a,b,c,d чтобы результат выражения $a \&\&b \parallel c^d$ являлся истиной?

- ☐ a=false, b=false, c=true, d=false
- ☒ все ответы верны
- ☐ a=true, b=true, c=false, d=true
- ☐ a=false, b=false, c=false, d=true

18. Что означает данный оператор >> ?

- ☐ оператор беззнакового правого побитового сдвига.
- ☐ оператор левого побитового сдвига
- ☐ знак больше
- ☒ оператор правого побитового сдвига с сохранением знака отрицательного числа

19. Что означает данный оператор >>> ?

- ☒ оператор беззнакового правого побитового сдвига.
- ☐ знак больше
- ☐ оператор левого побитового сдвига
- ☐ оператор правого побитового сдвига с сохранением знака отрицательного числа

20. Что означает данный оператор << ?

- ☐ оператор беззнакового правого побитового сдвига.
- ☐ оператор беззнакового левого побитового сдвига
- ☐ знак меньше
- ☒ оператор левого побитового сдвига

21. Какое значение примет переменная result после выполнения программы?

```
int[] m = {4, 8, 10, 16, 12};
int result = 0;
for(int i=0; i<4; i++) {
    result += m[i];
}
System.out.println(result);
```

- ☐ 0
- ☐ 4
- ☐ 50
- ☒ 38

22. Укажите правильный синтаксис тернарного оператора в Java

- ☒ логическое_условие ? выражение1 : выражение2;
- ☐ if логическое_условие ? выражение1 : выражение2;
- ☐ if логическое_условие then выражение1 else выражение2;
- ☐ do ... while

23. В каких случаях применяют оператор switch?

- ☒ к известному числу возможных ситуаций выбора
- ☐ применяют, когда число ситуаций выбора не известно
- ☐ когда происходит дублирование значений выбора
- ☐ данный оператор в программировании не используется

24. Дан код.

```
public class Main {
    static int firstNum = 10;
    public static void main(String[] args) {
        String res = Integer.toHexString(firstNum++);
        System.out.println("postfix: " + res);
    }
}
```

Какой результат будет выведен после компиляции?

- ☒ postfix: a
- ☐ postfix: b
- ☐ postfix: 10
- ☐ postfix: 11

25. Дан код.

```
public class Main {  
    static int firstNum = 10;  
    public static void main(String[] args) {  
        String res = Integer.toHexString(++firstNum);  
        System.out.println("postfix: " + res);  
    }  
}
```

Какой результат будет выведен после компиляции?

- ☐ postfix: a
- ☒ postfix: b
- ☐ postfix: 10
- ☐ postfix: 11

26. Дан код.

```
public class Main {  
    static int firstNum = 10;  
    public static void main(String[] args) {  
        String res = Integer.toHexString(firstNum--);  
        System.out.println("postfix: " + res);  
    }  
}
```

Какой результат будет выведен после компиляции?

- ☒ postfix: a
- ☐ postfix: b
- ☐ postfix: 10
- ☐ postfix: 11

27. Дан код.

```
public class Main {  
    static int firstNum = 10;  
    public static void main(String[] args) {  
        String res = Integer.toHexString(--firstNum);  
        System.out.println("postfix: " + res);  
    }  
}
```

Какой результат будет выведен после компиляции?

- ☐ postfix: a
- ☐ postfix: b
- ☐ postfix: 10
- ☐ postfix: 11
- ☐ postfix: 9

28. Дан код.

```
int a= 10;  
int b= 10;  
String res = Integer.toHexString(a&b);  
System.out.println(res);
```

Какой результат будет выведен после компиляции

- ☒ a
- ☐ 1100
- ☐ 1011
- ☐ 10

29. Дан код.

```
int a= 10;  
int b= 11;  
String res = Integer.toHexString(a|b);  
System.out.println(res);
```

Какой результат будет выведен после компиляции

- ☒ b
- ☐ a
- ☐ 1001
- ☐ 1011
- ☐ 9

30. Дан код.

```
int a= 10;  
int b= 11;  
String res = Integer.toHexString(a^b);  
System.out.println(res);
```

Какой результат будет выведен после компиляции

- ☒ 1
- ☐ a
- ☐ 1001
- ☐ 1011
- ☐ b

31. Дан код.

```
int a= 10;  
String res = Integer.toHexString(a<<1);  
System.out.println(res);
```

Какой результат будет выведен после компиляции

- ☐ b
- ☐ a
- ☐ 10
- ☐ 20
- ☒ 14

32. К стилям (парадигмам) программирования относят:

- ☒ императивный

- ☒ функциональный
- ☐ итерационный
- ☐ линейный

33. К стилям (парадигмам) программирования относят:

- ☒ объектно-ориентированный
- ☒ процедурный (императивный)
- ☐ ветвящийся
- ☐ многопоточный

34. Основным преимуществом объектно-ориентированного программирования перед императивным является

- ☒ сокращение количества межмодульных вызовов и уменьшение объемов информации, передаваемой между модулями
- ☐ отказ от указателей
- ☐ использование специальных библиотек
- ☐ повышенные требования к аппаратному обеспечению

35. Статический аспект объектно-ориентированной декомпозиции программной системы предполагает наличие следующих отношений между классами

- ☒ ассоциация
- ☒ зависимость
- ☐ сцепление
- ☐ делегирование

36. К принципам ООП относят

- ☒ иерархия
- ☒ инкапсуляция
- ☐ динамическое распределение памяти
- ☐ переопределение методов

37. К принципам ООП относят

- ☒ полиморфизм
- ☒ инкапсуляция
- ☐ статическое распределение памяти
- ☐ перегрузка методов

38. К принципам ООП относят

- ☒ иерархия
- ☒ абстрагирование
- ☐ применение интерфейсов
- ☐ применение статических членов класса

39. К принципам ООП относят

- ☒ модульность
- ☒ иерархия

- ☐ динамическое распределение памяти
- ☐ приведение типов

40. К принципам ООП относят

- ☒ типизация
- ☒ параллелизм
- ☐ статическое распределение памяти
- ☐ использование указателей

Краткое описание и регламент выполнения

К тестам допускаются все студенты.

По результатам итогового теста студент может набрать максимально 100 баллов.

7.2.2. Комплект отчетов по практическим работам (примеры)

(наименование оценочного средства)

Комплект отчетов по практическим работам

Практическая работа №1 «Основные утилиты Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №2 «Основы синтаксиса Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №3 «Объектно-ориентированное программирование в Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №4 «Интерфейсы»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №5 «Обработка строк на Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №6 «Работа с датой и временем на Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №7 «Обобщенные типы»

Форма отчета по практическому занятию №7

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №8 «Java Collection Framework»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №9 «Java I/O и java NIO»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №10 «Обработка исключений»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №11 «Лямбда выражения»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №12 «Регулярные выражения»

Форма отчета:

- титульный лист;
- задание;

- результат выполнения задания;
- выводы по работе.

Практическая работа №13 «JAVA DATABASES»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №14 «Работа с XML на Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №15 «Паттерны проектирования Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Практическая работа №16 «Многопоточность и синхронизация в Java»

Форма отчета:

- титульный лист;
- задание;
- результат выполнения задания;
- выводы по работе.

Требования к оформлению

Отчет должен содержать подробное описание (включая иллюстративный материал) последовательности действий, сделанных студентом для выполнения заданий.

Процедура оценивания

Оценка выполненной работы проводится по критериям:

1. Наличие всей существенной информации по работе
2. Точность и полнота предоставляемых сведений
3. Непротиворечивость приводимой информации
4. Правильность интерпретаций и выводов, которые сделаны по результатам работы
5. Степень достижения студентом поставленной цели
6. Обоснованность применяемого решения
7. Грамотность (содержательная) используемых формулировок

Критерии оценки за отчеты по практическим работам:

Полностью выполненное и вовремя защищенный отчет – максимальный балл. За каждое невыполненное задание снимаются баллы в соответствии с заданием на практическое занятие. Просрочка на 1 неделю - коэффициент 0,75, за две - 0,5, за три - 0,25, за четыре и более - 0 (учитывается факт сдачи).

7.2.3. Комплект заданий для оценки сформированности компетенций (примеры)

(наименование оценочного средства)

ПК-5 Способен осуществлять выбор языка программирования и моделировать решение для реализации программного обеспечения

код и наименование компетенции

ОМ закрытого типа

Задание 1

Выберите один правильный вариант ответа.

Что такое JDBC в Java?

- a) Java Database Code
- b) Java Database Control
- c) Java Data Control
- d) Java Database Connectivity

Правильный ответ:

- d) Java Database Connectivity

Задание 2

Выберите один правильный вариант ответа.

Какие основные шаги необходимо выполнить для установки соединения с базой данных с использованием JDBC?

- a) Подключить драйвер JDBC, создать объект Connection, выполнить SQL-запрос, закрыть соединение.
- b) Подключить драйвер JDBC, создать объект Statement, выполнить SQL-запрос, закрыть соединение.
- c) Подключить драйвер JDBC, создать объект ResultSet, выполнить SQL-запрос, закрыть соединение.
- d) Подключить драйвер JDBC, создать объект Connection, создать объект ResultSet, закрыть соединение.

Правильный ответ:

- a) Подключить драйвер JDBC, создать объект Connection, выполнить SQL-запрос, закрыть соединение.

Задание 3

Выберите один правильный вариант ответа.

Какой метод используется для выполнения SQL-запросов в JDBC?

- a) executeQuery()
- b) executeUpdate()
- c) execute()
- d) executeStatement()

Правильный ответ:

- b) executeUpdate()

Задание 4

Выберите один правильный вариант ответа.

Каким образом в JDBC можно предотвратить атаки SQL-инъекций?

- a) Используя параметризованные запросы (Prepared Statements) с подстановкой значений.
- b) Используя метод setSQLFilter() на объекте Connection.
- c) Применяя механизм шифрования для SQL-запросов.
- d) Отключая автоматический коммит транзакций.

Правильный ответ:

- а) Используя параметризованные запросы (Prepared Statements) с подстановкой значений.

Задание 5

Выберите один правильный вариант ответа.

Каким образом освобождаются ресурсы (Connection, Statement, ResultSet) после завершения работы с базой данных в JDBC?

- а) Закрытием каждого объекта вручную с помощью метода close().
- б) Они освобождаются автоматически после завершения блока кода, где были созданы.
- в) Освобождаются только при завершении работы самой программы.
- г) Путем вызова метода releaseResources() на объекте Connection.

Правильный ответ:

- а) Закрытием каждого объекта вручную с помощью метода close().

ОМ открытого типа

Задание 6

Дайте развернутый ответ.

Что такое JDBC в Java?

Правильный ответ: JDBC (Java Database Connectivity) - это API (Application Programming Interface) в Java, которое предоставляет доступ к базам данных и позволяет выполнять SQL-запросы и манипулировать данными.

Задание 7

Дайте развернутый ответ.

Какие основные шаги необходимо выполнить для установки соединения с базой данных с использованием JDBC?

Правильный ответ:

- а) Загрузить драйвер JDBC с помощью метода Class.forName().
- б) Создать объект Connection, используя DriverManager.getConnection().
- в) Создать объект Statement или PreparedStatement, используя Connection.createStatement() или Connection.prepareStatement().
- г) Выполнить SQL-запрос с помощью методов executeQuery() (для SELECT) или executeUpdate() (для INSERT, UPDATE, DELETE).
- д) Закрыть соединение и освободить ресурсы с помощью метода close().

Задание 8

Дайте развернутый ответ.

Какой метод используется для выполнения SQL-запросов в JDBC?

Правильный ответ: Для выполнения SQL-запросов в JDBC используется метод executeQuery() для SELECT-запросов, который возвращает объект ResultSet, и метод executeUpdate() для INSERT, UPDATE или DELETE-запросов, который возвращает количество измененных строк.

Задание 9

Дайте развернутый ответ.

Каким образом в JDBC можно предотвратить атаки SQL-инъекций?

Правильный ответ: Чтобы предотвратить атаки SQL-инъекций в JDBC, рекомендуется использовать параметризованные запросы (Prepared Statements) вместо обычных SQL-запросов с конкатенацией значений. Параметризованные запросы обрабатывают данные как

параметры, а не как часть SQL-запроса, что делает невозможным внедрение вредоносного кода.

Задание 10

Дайте развернутый ответ.

Каким образом освобождаются ресурсы (Connection, Statement, ResultSet) после завершения работы с базой данных в JDBC?

Правильный ответ: Чтобы освободить ресурсы после завершения работы с базой данных в JDBC, необходимо следовать правильной последовательности закрытия объектов:

- а) Закрывать объекты ResultSet с помощью метода close() в порядке от последнего созданного к первому.
- б) Закрывать объекты Statement или PreparedStatement с помощью метода close().
- в) Закрывать объект Connection с помощью метода close().

7.3. Оценочные средства для промежуточной аттестации по итогам освоения дисциплины

7.3.1. Вопросы к промежуточной аттестации

Семестр 4

№ п/п	Вопросы к экзамену
1	Понятие технологии Java: общие сведения, состав (язык программирования и платформа Java).
2	Процесс разработки Java SE приложения. Принципы обеспечения кроссплатформенности Java приложений.
3	JVM: место в платформе Java, области памяти.
4	JVM: управление памятью, острова изоляции.
5	Загрузка классов в Java.
6	JDK: назначение, установка, основные утилиты JDK.
7	JDK: Работа с jar файлами. Файлы манифеста. Entry Point.
8	Структура программы на Java.
9	Лексика языка Java: Unicode, комментарии, Java doc комментарии
10	Лексика языка Java: идентификаторы, целочисленные литералы.
11	Лексика языка Java: вещественные и логические литералы.
12	Лексика языка Java: строковые и символьные литералы, escape последовательности.
13	Java Code conventions: основные положения.
14	Характеристика языка Java с точки зрения типизации. Классификация типов в Java.
15	Переменные и константы в Java. Виды переменных Java и соответствующие области памяти JVM.
16	Простые и квалифицированные имена.
17	Операции: понятие и классификация по числу операндов.
18	Операция присваивание и арифметические операции.
19	Логические операции.
20	Приведение типов в операциях (явное и неявное). Проблемы

№ п/п	Вопросы к экзамену
21	Выражения: понятие, правила вычисления.
22	Операторы Java: понятие, классификация.
23	Операторы ветвлений и циклов.
24	Аннотации Java.
25	Массивы: понятие, объявление, инициализация, способы создания.
26	Работа с массивами Java: доступ к элементам. Члены типа Array.
27	Клонирование одномерных и многомерных массивов.
28	Класс Arrays.
29	Перечисления в Java.
30	Область видимости имен.
31	Принципы ООП.
32	Основные характеристика языка Java как ОО языка программирования. Объектная модель Java.
33	Объявление классов. Модификаторы доступа. Члены класса.
34	Конструкторы. Создание экземпляров класса
35	Методы класса.
36	Абстрактные методы и классы.
37	Окончательные члены и классы.
38	Статические члены класса.
39	Методы с переменным числом параметров.
40	Инкапсуляция и наследования в Java. Отношения «HAS-A» и «IS-A».
41	Сигнатура метода. Полиморфизм. Перегрузка и переопределение методов. Аннотация Override.
42	Вложенные классы.
43	Анонимные классы.
44	Пакеты: понятие, назначение, объявление, импорт.
45	Интерфейсы Java: понятие, назначение, объявление, модификаторы, реализация.
46	Интерфейсы Java 8.
47	Проектирование классов и интерфейсов в UML (изображения классов и интерфейсов, отношения между классами/интерфейсами) и отображение диаграмм UML в код Java.
48	Class Object: назначение, обзор методов. Контракт и реализация метода equals().
49	Class Object: обзор методов. контракт и реализация метода hashCode().
50	Mutable и Immutable классы.
51	Class Class.
52	Wrapper Classes.
53	Автоматическая упаковка и распаковка.
54	Класс BigInteger.
55	Класс BigDecimal.

№ п/п	Вопросы к экзамену
56	Generics: типы, методы, ограниченные (bounded) типы.
57	Классы для работы со строками. Интерфейс CharSequence.
58	Класс String: назначение, особенности, конструкторы, создание строк.
59	Форматирование строк. Поиск символов и подстрок в строке.
60	Замена и сравнение строк.
61	Класс StringBuilder.
62	Класс StringTokenizer.
63	Основы регулярных выражений.
64	Интернационализация и локализация. Класс ResourceBundle.
65	Интернационализация и локализация. Класс Locale.
66	Работа с датой и временем в Java 7: класс Date.
67	Работа с датой и временем в Java 7: класс Calendar.
68	Работа с датой и временем в Java 7: форматирование даты и времени.
69	Принципы работы с датой-временем в Java 8.
70	Работа с датой-временем средствами java.time (Java 8).
71	Понятие коллекции. Преимущества. Элементы Java Collections Framework.
72	Java Collections Framework: основные интерфейсы.
73	Java Collections Framework: основные реализации.
74	Интерфейс Collection: назначение, особенности, основные методы, обход коллекции.
75	Интерфейс Set: назначение, особенности, основные методы.
76	Интерфейс List: назначение, особенности, основные методы.
77	Интерфейс Queue: назначение, особенности, основные методы.
78	Интерфейс Map: назначение, особенности, основные методы.
79	Сортировка элементов в коллекциях.
80	Класс Collections.
81	Класс Arrays.
82	Виды реализаций Java Collection Framework. Обзор General-purpose реализаций.
83	Состав Java I/O: возможности пакетов java.io и java.nio.
84	Понятие потока. Байтовые и символьные потоки.
85	Понятие потока. Входные и выходные потоки.
86	Буферизированные потоки. Общий алгоритм работы с потоками.
87	Иерархия символьных потоков: обзор.
88	Иерархия байтовых потоков: обзор.
89	Классы, создающие потоки.
90	Классы, управляющие потоками.
91	Цепочки потоков и их использование в Java I/O.
92	Потоки преобразования байтов в примитивные типы.

№ п/п	Вопросы к экзамену
93	Вывод на строчные устройства.
94	Потоки класса System
95	Класс StreamTokenizer.
96	Класс SequenceInputStream.
97	Класс LineNumberReader.
98	Класс RandomAccessFile.
99	Класс Console.
100	Управление жизненным циклом потока с использованием try-with- recourse.
101	Класс FileSystems.
102	Интерфейс Path и класс Paths.
103	Класс Files.
104	Работа с Random Access Files
105	Понятие исключения. Механизм перехвата и обработки исключения. Exception object.
106	The Catch or Specify Requirement.
107	Классификация и иерархия исключений на Java.
108	Класс Throwable.
109	Обработка исключений. Try-with-resources.
110	Выбрасывание исключений. Цепочки исключений.
111	Logging API.
112	Потоки и процессы. Стандарт Posix и области памяти JVM.
113	Класс Thread и интерфейс Runnable. Способы создания потоков.
114	Приставка и прерывание спящего потока.
115	Методы wait() и notify(). Общий алгоритм работы на мониторе в Java.
116	Потоки-пользователи и потоки-демоны.
117	Правило <i>happens-before</i> .
118	Синхронизация. Критические секции.
119	Atomic.
120	volatile.
121	Проблемы многопоточности и рекомендации по их обходу в Java.
122	Блокировки.
123	Executors: основные понятия.
124	Thread Pools
125	Fork/join framework.
126	Потокобезопасные коллекции.
127	Понятие XML. Структура и правила XML документов.
128	Элементы XML.
129	Понятие схемы документа XML. DTD.

№ п/п	Вопросы к экзамену
130	Понятие схемы документа XML. XSD.
131	Программные интерфейсы для работы с XML.
132	Алгоритм и виды работы парсеров.
133	DOM парсер: понятие, узлы.
134	Методы DOM парсера.
135	SAX парсер: понятие, события.
136	Построение приложения с SAX парсером.
137	JAXP: понятие, обзор технологий.
138	JAXB: понятие, обзор технологий.
139	JDBC: понятие, структура пакетов, набор компонентов.
140	Использование JDBC в двухуровневых и многоуровневых приложениях.
141	Категории JDBC драйверов.
142	java.sql.DriverManager.
143	java.sql.Connection.
144	java.sql.Statement.
145	java.sql.ResultSet.
146	Алгоритм работы с JDBC.
147	PreparedStatement и CallableStatement.
148	Обработка SQLExceptions.
149	Обзор технологии RMI.
150	Создание удаленного RMI объекта.
151	Создание RMI клиента.
152	Какие порождающие паттерны проектирования бывают?
153	Какие структурные паттерны проектирования бывают?
154	Какие поведенческие паттерны проектирования бывают?
155	Какой паттерн проектирования лучше использовать для создания объектов?

7.3.2. Критерии и нормы оценки

Семестр	Форма проведения промежуточной аттестации	Критерии и нормы оценки	
6	Экзамен (по накопительному рейтингу)	отлично	от 85 до 100 баллов
		хорошо	от 70 до 84 баллов
		удовлетворительно	от 55 до 69 баллов
		неудовлетворительно	от 0 до 54 баллов

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Обязательная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1	Монахов, В. В.	Язык программирования Java и среда NetBeans	Учебное пособие	2021	ЭБС "IPRbooks"
2	Мухаметзянов, Р. Р.	Основы программирования на Java	Учебное пособие	2017	ЭБС "IPRbooks"
3	Вязовик, Н. А.	Программирование на Java	Учебное пособие	2021	ЭБС «IPRBooks»

8.2. Дополнительная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1	Гаряева В. В	Решение задач с использованием пакетов прикладных программ	Учебное пособие	2017	ЭБС "IPRbooks"
2	Лоскутов В. И., Коробова И.Л.	Разработка информационных систем для Windows Store		2016	ЭБС "IPRbooks"
3	Стешин А. И.	Информационные системы в организации	Учебное пособие	2019	ЭБС "IPRbooks"
4	Голицына О. Л., Максимов Н.В., Попов И.И.	Информационные системы	Учебное пособие	2018	ЭБС "Znaniy.com"

8.3. Перечень профессиональных баз данных и информационных справочных систем

- About SWEBOOK. Режим доступа: <https://www.computer.org/web/swebok>, 2016-01-01.
- Java и вы. Режим доступа: <http://www.java.com/ru/>, 2016-01-01.
- Oracle Technology Network - Java. Режим доступа: <http://www.oracle.com/technetwork/java/index.html>, 2016-01-01.
- Project Management Institute. Режим доступа: <http://www.pmi.org/>, 2016-01-01.

8.4. Перечень программного обеспечения

№ п/п	Наименование ПО	Реквизиты договора (дата, номер, срок действия)
1	Eclipse Foundation Eclipse версия 4	Лицензия Eclipse Public License
2	NetBeans Community NetBeans IDE версия 8	Лицензия LGPLv2.1, GPLv2 with Classpatch exception
3	The CodeBlocks team CodeBlocks версия 16-	Лицензия GNU GPLv3

8.5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

№ п/п	Наименование оборудованных учебных кабинетов, лабораторий, мастерских и др. объектов для проведения практических и лабораторных занятий, помещений для самостоятельной работы обучающихся (номер аудитории)	Перечень основного оборудования
1	Компьютерный класс. Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для проведения лабораторных работ. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций. Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации (УЛК-408).	Компьютер (монитор 17", системный блок Intel (R) Celeron (R) 2,66 GHz / 1 Gb / 80 Gb), маршрутизатор 2801 Router, коммутатор Catalyst, экран/интерактивная доска Smart Board TV, проектор Acer P1303W., стол преподавательский, стол ученический, стол компьютерный, стул, доска аудиторная (маркерная).
2	Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций. Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации (УЛК-413).	Стол ученический двухместный (моноблок), стол преподавательский, стул, доска аудиторная (меловая).

№ п/п	Наименование оборудованных учебных кабинетов, лабораторий, мастерских и др. объектов для проведения практических и лабораторных занятий, помещений для самостоятельной работы обучающихся (номер аудитории)	Перечень основного оборудования
3	Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций. Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации (УЛК-418).	Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций. Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации.
4	Помещение для самостоятельной работы Студентов (Г-401)	Стол ученический, стул, компьютер с выходом в сеть интернет.