

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ  
“БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”  
КАФЕДРА ИИТ

ОТЧЁТ  
по лабораторной работе №4

Выполнил:  
студент 3 курса  
группы ПО-9  
Оводок В.В.

Проверил:  
Крощенко А.А.

**Цель:**приобрести практические навыки в области объектно-ориентированного проектирования

## Вариант 5

### Задание 1

Реализовать указанный класс, включив в него вспомогательный внутренний класс или классы. Реализовать 2-3 метода (на выбор). Продемонстрировать использование реализованных классов.

Создать класс Department (отдел фирмы) с внутренним классом, с помощью объектов которого можно хранить информацию обо всех должностях отдела и обо всех сотрудниках, когда-либо занимавших конкретную должность.

Выходные данные:

```
Employees in the IT department:  
Position: Software Developer, Employees: [Vadim, Vlad]  
Position: System Analyst, Employees: [Dima]  
Total employees in the IT department: 3
```

Краткий текст программы:

```
public static void main(String[] args) {  
    Department itDepartment = new Department("IT");  
    itDepartment.addPost("Software Developer");  
    itDepartment.addPost("System Analyst");  
  
    itDepartment.posts.get(0).addEmployee("Vadim");  
    itDepartment.posts.get(0).addEmployee("Vlad");  
  
    itDepartment.posts.get(1).addEmployee("Dima");  
  
    itDepartment.displayEmployees();  
  
}
```

### Задание 2

Реализовать агрегирование. При создании класса агрегируемый класс объявляется как атрибут (локальная переменная, параметр метода). Включить в каждый класс 2-3 метода на выбор. Продемонстрировать использование разработанных классов.

Создать класс Абзац, используя класс Слово.

Выходные данные:

```
Paragraph: my name is vadim  
Number of words in the paragraph: 4  
The midaV is not a palindrome  
Paragraph: my name is midaV
```

Краткий текст программы:

```
public static void main(String[] args) {
    Paragraph paragraph = new Paragraph();
    paragraph.addWord("my");
    paragraph.addWord("name");
    paragraph.addWord("is");
    paragraph.addWord("vadim");

    paragraph.displayParagraph();
    System.out.println("Number of words in the paragraph: " + paragraph.countWords());

    Word word = paragraph.words.get(3);
    word.capitalizeFirstLetter();
    word.reverseWord();
    word.isPalindrome();

    paragraph.displayParagraph();
}
```

### Задание 3

Построить модель программной системы с применением отношений (обобщения, агрегации, ассоциации, реализации) между классами. Задать атрибуты и методы классов. Реализовать (если необходимо) дополнительные классы. Продемонстрировать работу разработанной системы.

Система Библиотека. Читатель оформляет Заказ на Книгу. Система осуществляет поиск в Каталоге. Библиотекарь выдает Читателю Книгу на абонемент или в читальный зал. При невозвращении Книги Читателем он может быть занесен Администратором в «черный список».

Выходные данные:

```
Книга Заводной апельсин выдана читателю Вадим
Книга 1984 выдана читателю Влад
Книга 451 градус выдана читателю Дима
Книга 451 градус находится в прокате Вадим не может взять эту книгу
Книга Заводной апельсин возвращена читателем Вадим
Книга 1984 возвращена читателем Влад
Влад добавлен в черный список
Книга 451 градус возвращена читателем Дима
Читатель Влад в черном списке и не может взять книгу 451 градус
Книга 451 градус выдана читателю Вадим
```

Краткий текст программы:

```
public static void main(String[] args) {  
    Library library = new Library();  
  
    Book book1 = new Book("Заводной апельсин", "Энтони Берджес");  
    Book book2 = new Book("1984", "Джордж Оруэлл");  
    Book book3 = new Book("451 градус", "Рэй Бредбери");  
    Book book4 = new Book("О дивный новый мир", "Олдос Хаксли");  
  
    Catalog catalog = new Catalog();  
  
    catalog.addBook(book1);  
    catalog.addBook(book2);  
    catalog.addBook(book3);  
    catalog.addBook(book4);  
  
    Reader reader1 = new Reader("Вадим");  
    Reader reader2 = new Reader("Влад");  
    Reader reader3 = new Reader("Дима");  
  
    Order order1 = reader1.placeOrderForReadingRoom(book1);  
    Order order2 = reader2.placeOrderForHome(book2, LocalDate.of(2024, 3, 1));  
    Order order3 = reader3.placeOrderForHome(book3, LocalDate.of(2024, 5, 5));  
    Order order4 = reader1.placeOrderForReadingRoom(book3);  
  
    LibraryWorker libraryWorker = new LibraryWorker(library);  
  
    libraryWorker.addOrder(order1);  
    libraryWorker.addOrder(order2);  
    libraryWorker.addOrder(order3);  
    libraryWorker.addOrder(order4);  
  
    libraryWorker.issueBookToReader(order1);  
    libraryWorker.issueBookToReader(order2);  
    libraryWorker.issueBookToReader(order3);  
    libraryWorker.issueBookToReader(order4);  
  
    reader1.returnBook(book1, libraryWorker);  
    reader2.returnBook(book2, libraryWorker);  
    reader3.returnBook(book3, libraryWorker);  
  
    Order order5 = reader2.placeOrderForReadingRoom(book3);  
    libraryWorker.issueBookToReader(order5);  
    libraryWorker.issueBookToReader(order4);  
}
```

**Вывод:** приобрел практические навыки в области объектно-ориентированного проектирования.